

BETA VERSION

Når man kommer med kommentarer eller rettelser til en given beta version bedes man angive det tilhørende id-nummer

ID: 2011-09-13-20-10

LARS MADSEN

INTRODUKTION TIL L^AT_EX

– noget for alle

MMXI

Copyright © 2002–2011 Lars Madsen

REGLER FOR ANVENDELSE: Ingen ud over de almindelige copyrightregler. Man bedes dog undlade at placere kopier af bogen til download fra egen hjemmeside, link i stedet direkte til <http://www.imf.au.dk/system/latex/bog/> hvorved man sikrer sig at alle altid har direkte adgang til seneste version af bogen.

Layout og typografi valgt og implementeret af forfatteren

Skrifttype: 11 pt / 13.6 pt Kp-fonts

Sat med \LaTeX og *memoir*-klassen

Alle figurer er lavet med METAPOST eller TiKz

Anvendte pakker (pakker som kun er anvendt i eksempler er ikke inkluderet ej heller pakker som er anvendt af andre pakker):

afterpage	fix-cm	mathtools	svn-multi
amsmath	fixltx2e	mdframed	textcomp
babel	fixme	mflogo	threeparttablex
berasans	fontenc	mhchem	tikz
beramono	fourier-orns	microtype	upquote
bm	graphicx	multicol	url
calc	hyperref	natbib	varioref
caption	ifsym	ntheorem	verbatimcopy
comment	inputenc	paralist	xcolor
csquotes	keyval	placeins	xkeyval
dlfltxbmarkup	kpfonts	ragged2e	xparse
dlfltxbmisc	lipsum	siunitx	xspace
dlfltxbcodetips	listings	soul	
dlfltxbtocconfig	longtable	stmaryrd	
enumitem	mathdots	subfig	

Samt desuden nogle (endnu) ikke-offentlige pakker:

felinebooksetup	felineSD	felinepreamble	felinetablenotes
-----------------	----------	----------------	------------------

Alle registrerede varemærker nævnt i denne bog tilhører deres respektive ejere.

*til de danske L^AT_EX-brugere med håbet om
at dette vil hjælpe på vejen mod pænere
dokumenter*

*samt til alle de som fortsat bruger tid på at
forbedre L^AT_EX og venner – nye som gamle*

Kort oversigt

	<i>i · Kort oversigt</i>
	<i>iii · Indholdsfortegnelse</i>
	<i>xiii · Tabeloversigt</i>
	<i>xiv · Figuroversigt</i>
	<i>xv · Forord</i>
Grundlæggende L^AT_EX	<i>3 · Den forunderlige L^AT_EXniske verden</i>
	<i>21 · Almindelig tekst</i>
	<i>45 · Introduktion til matematik</i>
Overbyggende L^AT_EX	<i>83 · Matematik på et højere niveau</i>
	<i>115 · Om sætningskonstruktioner</i>
	<i>135 · Om grafik, flydende objekter og billedtekst</i>
	<i>165 · Om tabeller</i>
	<i>197 · Konstruktion af litteraturliste</i>
	<i>223 · Mere om tekst</i>
	<i>263 · Diverse</i>
Vedr. større projekter	<i>291 · Håndtering af større projekter</i>
	<i>301 · Dokumentopsætning med memoir-klassen</i>
	<i>331 · Generelle gode råd vedr. forskellige projekter</i>
	<i>337 · Om justering af standardklaesserne</i>
Yderligere materiale	<i>351 · Om generering af nye makroer</i>
	<i>363 · Nyttige metoder og konstruktioner</i>
	<i>377 · L^AT_EX-distributioner samt andre redskaber</i>
	<i>399 · Om fejlmeddelelser</i>
	<i>407 · Kommentarer</i>
	<i>419 · Skabeloner</i>
	<i>421 · Litteratur og pakkeinformation</i>
	<i>433 · Stikordsregister</i>

Indholdsfortegnelse

Kort oversigt	i
Indholdsfortegnelse	iii
Tabeloversigt	xiii
Figuroversigt	xiv

Forord

Forord	xv
Emner som ikke gennemgås i denne bog	xv
Indholdet af bogen	xvi
Om bogens syntaks	xvii
Kommentarer til bogen	xviii
Takkeskrivelse	xix

Grundlæggende L^AT_EX

1 Den forunderlige L^AT_EXniske verden **3**


Kort introduktion til hvad L^AT_EX er for en fisk og hvordan man anvender systemet.

1.1	Introduktion	3
1.1.1	Lidt historie	3
1.1.2	Om udtalen af L ^A T _E X	4
1.1.3	Hvad er det så for noget?	4
1.2	Kært barn mange navne	5
1.3	Hvor får man L ^A T _E X fra?	6
1.4	Om at udføre L ^A T _E X	6
1.5	Show us the code	6
1.6	L ^A T _E X-syntaks	8
1.6.1	Lidt L ^A T _E X-terminologi	9
	<i>Mange makroer kan godt lide (at spise) mellemrum</i> 10	
1.6.2	Dokumentklassen	10
	<i>Nogle dokumentklasser</i> 12	
1.6.3	Preamblen	12
1.6.4	Selve dokumentet	13
1.6.5	Encoding	13
1.7	Når noget er gået galt	15
1.8	Hvordan finder man hjælp?	16
1.8.1	Hvordan finder man manualen til pakken X?	16
1.8.2	Når man skal spørge om hjælp	17
1.8.3	Minimalt eksempel	18
1.9	Dansk opsætning	19

Indholdsfortegnelse

2	Almindelig tekst	21
	<i>Hvordan man skriver almindelige tekstuelle konstruktioner i L^AT_EX.</i>	
2.1	Tekstopdeling	21
	<i>Nyt afsnit 21 · Afsnitsindrykning 22 · Tvungen sideskift 22 · Overskriftsopdeling af teksten 23 · Afsnitoverskrifter uden nummer 24</i>	
2.2	Tegnsætning	24
	<i>Orddeling 24 · Anførelstegn 25 · Om streger 26 · Streger og orddeling 27 · Ligaturer 27 · Tvungen mellemrum uden lineombrydning 28 · Engelsk punktum og mellemrum 28 · Ellipsis 29 · Specialbogstaver og accenter 29 · Tekstsymboler 29</i>	
2.3	Om fontvalg og -størrelser	30
	<i>Fontstørrelser 30 · Skift af fontvariant 31 · Fremhævning af tekst 33</i>	
2.4	Krydsreferencer	34
2.5	Fodnoter	35
2.6	Indholdsfortegnelse	36
2.7	Justering af tekst	37
	<i>Centrering 37 · Venstre-/højrejustering af tekstaafsnit 37 · Justering med indbygget orddeling 37 · Citater 38</i>	
2.8	Lister	38
	<i>Punktopstilling 38 · Nummereret liste 39 · Beskrivende liste 41</i>	
2.9	Simpel indsættelse af titel, forfatternavn og dato	41
2.10	Hvordan håndterer man URLer, stier og filnavne	42
3	Introduktion til matematik	45
	<i>Introducerende kapitel om grundlæggende matematiske konstruktioner i L^AT_EX. Basisviden som alle L^AT_EX-brugere bør kende til.</i>	
3.1	Matematik i L ^A T _E X	45
3.1.1	Lidt syntaksregler i matematik mode	45
	<i>Mellemrum tæller ikke 45 · Ingen danske vokaler i matematik 46 · Begrænset automatisk lineombrydning 46</i>	
3.1.2	Tegnsætning i matematik	47
3.1.3	Fremhævet matematik	47
	<i>One-liners 47 · Flerliniede formler 48</i>	
3.1.4	Formelnumre og krydsreferencer	50
3.2	Symboler og operatører	51
3.2.1	Forskellige symboler	51
	<i>Pile 54 · Skrive over eller under en pil 54</i>	
3.2.2	Operatorsymboler	55
3.2.3	Matematiske alfabetfonte	57
	<i>Skrive almindelig tekst indeni i et matematisk udtryk 57 · Danske vokaler i matematik 59 · Fed skrift i matematik 59</i>	
3.2.4	Navngivne funktioner og operatører	59
3.2.5	Navngivne mængder	61
3.2.6	... – prikker er ikke bare prikker	61
3.2.7	Matematiske accenter	62
3.3	Hegn	65
	<i>Om \left og \right 67 · Varianten \middle 68 · Om m-varianterne af \big... \Bigg 68</i>	
3.4	Almindelige matematiske konstruktioner	68
	<i>Sub- og superscript 68 · Brøker 69 · Binomialkoefficienter 70 · Rodfunktionen 70 · Modululus 70 · Afledt funktion 71 · Om matricer 71 · Gaffelfunktioner 75 · Over/underbrace 76</i>	

Indholdsfortegnelse

	· Grænser i flere lag 76 · Restingering/evaluering af funktioner 77	
3.5	Nogle almindelige hjemmelavede makroer	77
	Diverse 78	
	Overbyggende L^AT_EX	
4	Matematik på et højere niveau	83
	<i>Udvidet kapitel omkring matematiske konstruktioner og matematisk typografi.</i>	
4.1	Matematisk typografi	83
4.1.1	Hvor må man dele en formel?	83
4.1.2	Formelombrydning af matematik i teksten	83
	<i>Hvad gør man når man ikke kan få det til at passe? 85 · Hvor vil L^AT_EX automatisk dele matematiske udtryk i teksten? 85 · Automatisk indsættelse af gangetegn ved linieombrydning 86</i>	
4.1.3	Formelombrydning i fremhævede formler	86
	<i>Opstilling af fremhævede (brudte)formler 87</i>	
4.1.4	Tegnsætning i matematik	89
4.2	Environments til fremhævet matematik	90
4.2.1	Alignment	90
	<i>Align 91 · Aligned 92 · Flalign 93 · Kommentarer i alignment matematik 93</i>	
4.2.2	Centreret flerliniet matematik	94
4.2.3	Multline	94
4.2.4	Varianter	95
	<i>Split 95 · Aligned 96 · Alignedat 96 · Gathered 97</i>	
4.2.5	Eksemplerne 4.5 til 4.18 på side 87–89	97
4.2.6	Sideskift i fremhævet matematik	98
4.3	Ser du spøgelse?	99
4.4	Konfigurering af og leg med formelnummeret	99
	<i>Fjernelse af formelnummer 99 · Skrive noget andet i stedet for formelnummeret 99 · Konfigurering af hvordan formler nummereres 100 · Ét formelnummer til udregning 101 · Undernummerering af formelnummer 101 · Flytte lidt på formelnummeret 101 · Vis kun refererede formler 102</i>	
4.5	$\mathcal{A}\mathcal{M}\mathcal{S}$ udvidelsen mathtools	102
	<i>Justering vedrørende grænser på operatorer 102 · Intelligente afgrænserfunktioner 103 · Hvordan viser man kun refererede formler? 104 · lgathered 104 · Overlappe alignment 104 · Ensbetyderpile mellem linierne i en beregning 105 · Justering af indryk ved brudte formler 105 · Andre 106</i>	
4.5.1	Pakken empheq	106
4.6	Diagrammer i L ^A T _E X	107
4.6.1	Diagrammer med \Xypic og \xymatrix	107
4.6.2	\Xypic pile syntaks	108
	<i>Placering 108 · Pilens udformning 109 · Labels på pile 109</i>	
4.6.3	Prækompilering af diagrammer	110
4.6.4	\Xypic og Babel	110
4.6.5	\Xypic diagrammer og formelnumre	111
4.6.6	Andre \Xypic tips	111
4.6.7	Alternativ til \Xypic	112

Indholdsfortegnelse

5	Om sætningskonstruktioner	115
	<i>Brug af samt konfigurerings af sætningskonstruktioner. Inkluderer blandt andet hvordan man laver ordentlige bevis-environments.</i>	
5.1	Opret en ny sætningskonstruktion	116
5.2	Opsætning af sætningskonstruktioner med amsthm-pakken	117
5.2.1	Almindelig brug af amsthm	118
5.2.2	Beviser med amsthm	119
5.3	Opsætning af sætningskonfigurationer med ntheorem-pakken	120
5.3.1	Konfigurerings	121
5.3.2	Theoremstyles	122
	<i>Citering i sætningsoverskrift</i> 124	
5.3.3	Slutmarkører	124
5.3.4	Beviser med ntheorem	125
5.4	Andet godt med sætningskonstruktioner	127
5.4.1	Sætningskonstruktioner med ramme eller farvet baggrund	128
5.4.2	Konstruktioner som starter med en liste	131
5.4.3	Mere om optimeringseksemplet	132
6	Om grafik, flydende objekter og billedtekst	135
	<i>Om inklusion af ekstern grafik samt forklaring af konceptet bag begrebet floats. Desuden præsenteres konstruktioner til at anvende flere figurer samtidigt samt et kig på billedtekst (captions) og subfigurer.</i>	
6.1	Inklusion af ekstern grafik	135
6.1.1	Syntaks for inkludering af ekstern grafik	137
6.1.2	Yderligere konfiguration vedrørende grafik	139
	<i>En kildefil – flere outputformater</i> 139 · <i>METAPOST inklusion med pdflatex</i> 140 · <i>At hente figurer fra andre biblioteker</i> 140	
6.1.3	Inkludere dele af (pdf-) dokumenter	141
6.2	Flydende objekter	142
6.2.1	Syntaks	142
6.2.2	Algoritmen bag placeringen af flydende objekter	144
6.2.3	Her til og ikke længere	145
6.2.4	Ikke-flydende floats	146
6.3	Flere figurer på én gang	147
	<i>Flere figurer i samme float</i> 147 · <i>To figurer – en billedtekst</i> 147 · <i>To figurer ved siden af hinanden med hver deres billedtekst</i> 148 · <i>To figurer ved siden af hinanden – justering af billedtekst placering</i> 148 · <i>Tekst ved siden af figur – justering i toppen</i> 149	
6.4	Konfiguration af billedteksten	150
6.4.1	Brug af pakken caption	150
	<i>Ændring af billedtekstens font og fontstørrelse</i> 150 · <i>Ændring af separatortegn</i> 151 · <i>Bredden af caption</i> 151 · <i>Mellemrum før og efter billedtekst</i> 152 · <i>En figur og en tabel i samme float</i> 153 · <i>Flere floats med samme nummer</i> 153	
6.4.2	Undernummerering af figurer – subfig	154
	<i>Flere floats under samme nummer</i> 154	
6.5	Andre grafikrelaterede ting	155
	<i>At skrive tekst rundt om figurer</i> 155 · <i>Rette på labels i EPS figurer</i> 155 · <i>Baggrundsbilleder</i> 156 · <i>Figurer i landscape</i> 156	
6.6	Tips og tricks	157
6.6.1	At skrive ovenpå inkluderede billeder	157
6.6.2	Genstridige EPS-filer	157

Indholdsfortegnelse

6.6.3	Genstridige PDF-billeder	158
6.6.4	Generering af EPS-filer eller croppede PDF-filer via \LaTeX	159
6.6.5	Billed/figur-konvertering	159
	<i>EPS til PDF</i> 159 · <i>JPG til EPS</i> 160 · <i>PNG til EPS</i> 160 · <i>MPS til EPS</i> 160 · <i>PDF til EPS</i> 160 · <i>SVG til EPS</i> 161	
6.6.6	Generering af grafik til brug i \LaTeX	161
	<i>METAPOST</i> 161 · <i>Asymptote</i> 162 · <i>Gnuplot</i> 162 · <i>PSTRICKS</i> 162 · <i>TikZ/PGF</i> 162	
7	Om tabeller	165
	<i>Opsætning af tabeller, specielt med henblik på pæne datatabeller.</i>	
7.1	Guidelines vedr. opbygning af en (data)tabel	165
7.2	En læserundersøgelse	167
7.3	Tabelsyntaks	168
7.3.1	Flydende tabel	169
7.3.2	Angivelse af søjlerne	169
7.3.3	Overskriftsceller	173
7.3.4	Linier i tabeller	174
	<i>Booktabs makrosyntaks</i> 175 · <i>Brug af fuldbredde-linier</i> 175 · <i>Afbrudte linier</i> 176	
7.4	Dataformatering	177
7.4.1	Automatisk decimaljustering med dcolumn-pakken	177
7.4.2	Kildekoden til tabel 7.2	179
7.4.3	Dataformatering via pakken siunitx	179
7.5	Andre tabel-environments	182
7.5.1	Tabularx – tabel med fast bredde	182
7.5.2	Longtable – tabel over flere sider	183
7.6	Ændre ved luft i tabeller	185
	<i>Vertikal luft</i> 185 · <i>Horisontal luft</i> 186	
7.7	Noter i tabeller	188
7.7.1	Threeparttable(x)	188
7.8	Ekstra bemærkninger	191
7.8.1	Lade tabellen rage lidt ud i marginen	191
7.8.2	Tabeller eller figurer i landskabsorientering	191
7.8.3	Brug af farver i tabeller	193
7.8.4	Data fra eksterne kilder	194
	<i>Excel til \LaTeX</i> 194 · <i>Data fra CSV</i> 194	
8	Konstruktion af litteraturliste	197
	<i>Om citering af litteratur og hvordan man laver litteraturlister, både i hånden samt autogenereret via BibTeX.</i>	
8.1	Citeringskonventioner	197
8.2	Bibliografienvironmentet	197
8.2.1	Citerings syntaks	197
8.2.2	Forfatter-år citeringer lavet uden BibTeX	199
8.3	BibTeX	199
8.3.1	Hvad er BibTeX?	199
8.3.2	Forfatter-år citeringer med pakken natbib	202
	<i>Pakkeoptions til natbib</i> 203	

Indholdsfortegnelse

8.4	Generering af BibTeX-database	204
	<i>Beskrivelse af format</i> 204 · <i>Syntaks for author-feltet</i> 206 · <i>Specielt om titelfeltet</i> 206 · <i>URL i bibliografier</i> 207	
8.5	Bibliografistilarter	207
8.5.1	Eksempler på BibTeX konfigureringsfiler	207
	<i>Standardstile</i> 208 · <i>Danske oversættelser</i> 209 · <i>BibTeX-stile kompatible med natbib-pakken</i> 211 · <i>Eksempler fra jurabib pakken</i> 214	
8.6	Andet	215
	<i>Sortering af danske navne</i> 215 · <i>Sprogproblemer ved brug af BibTeX</i> 216 · <i>Online reference databaser</i> 216	
8.7	Emneopdelt litteraturliste	217
	<i>Brug af pakken multibib</i> 217 · <i>Kort om bibtopic</i> 219	
8.8	Håndtering af referencer databaser	221
	<i>Jabref</i> 221 · <i>RefWorks</i> 221	
8.9	Biblatex	222
9	Mere om tekst	223
	<i>Gennemgang af flere forskellige tekst relaterede emner, så som fontvalg, fontstørrelser, justering af tekst, lister, farver samt kodefremvisning.</i>	
9.1	Lidt mere om afsnitsoverskrifter samt indholdsfortegnelsen	223
9.1.1	Om appendiks/bilag	223
	<i>Appendices til et kapitel</i> 224 · <i>Bilag og appendiks</i> 224	
9.2	Fodnoter og endnotes/slutnoter	225
9.2.1	Mere om fodnoter	225
9.2.2	Endnotes/slutnoter	226
9.3	Mere om orddeling og sprog	226
	<i>Babel og orddeling</i> 227 · <i>Orddeling ved bindestreg og skråstreg</i> 228 · <i>Mærkelige orddelelinger</i> 228 · <i>Undgå orddeling</i> 229 · <i>Teste efter sprog</i> 229 · <i>Lokalt skift af sprog</i> 229	
9.4	Ændring af standardnavne kontrolleret af babel	229
9.5	Tekstfyld – Lorem ipsum	230
9.6	Fontrelaterede emner	231
9.6.1	Fontvalg med matematiksupport	231
	<i>Computer Modern</i> 232 · <i>Latin Modern</i> 232 · <i>Times</i> 232 · <i>Palatino</i> 233 · <i>Utopia og fourier</i> 233 · <i>New Century Schoolbook og fourier</i> 234 · <i>Garamond</i> 234 · <i>Kp-fonts</i> 234 · <i>Concrete</i> 235	
9.6.2	Et par sans-serif skrifttyper	235
	<i>Computer Modern sans serif</i> 235 · <i>Helvetica klon</i> 235 · <i>Bera sans</i> 236	
9.6.3	Fast-bredde skrifttype (skrivemaskinefont)	236
	<i>Computer Modern Typewriter</i> 236 · <i>TX Typewriter</i> 236 · <i>LuxiMono</i> 236 · <i>BeraMono</i> 236	
9.6.4	Vilkårlig fontstørrelse	237
9.6.5	Større linieafstand	237
9.6.6	At pille ved afstanden mellem bogstaver (letterspacing)	238
9.7	Lister	239
9.7.1	Flere lister med paralist-pakken	240
	<i>Enumerate og itemize</i> 240 · <i>Kompakte lister</i> 241 · <i>En løbende liste</i> 241	
9.7.2	Mere kontrol over lister med enumitem-pakken	241
	<i>Kort oversigt over enumitem parametre</i> 245	
9.7.3	Egne listekonstruktioner	245

Indholdsfortegnelse

9.8	Brug af farver	245
9.9	Udsende mange breve	247
9.10	Fremvisning af tekst uden L ^A T _E X-fortolkning	249
9.10.1	Kildekode fremvisning	250
9.10.2	Lidt om listings	251
	<i>Opsætning til SAS-udskrifter via listings</i> 252	
9.11	Stikordsregister	253
9.11.1	Generere indeksindgangene	253
9.11.2	Sortere selve indekset	255
	<i>Sortering via makeindex</i> 256 · <i>Sortering via texindy</i> 257	
9.12	Terminologiforklaring	257
9.12.1	Meget simpel symbolliste	258
9.12.2	Ikke helt så simpelt	258
9.13	Terminologi- eller tegnforklaring (glossaries-pakken)	258
9.13.1	Data til ordlisten	259
9.13.2	Brug i teksten	261
9.13.3	Indsættelse af den formaterede terminologiliste	262
9.13.4	Eksempel	262
10	Diverse	263
	<i>Diverse emner som ikke helt falder indenfor nogen specifik kategori, men vi kigger bl.a. på håndtering af enheder, noget kemi, slides og andre ting.</i>	
10.1	Kode til at tjekke for L ^A T _E X eller pdfL ^A T _E X	263
10.2	Formatering af tal, enheder og navngivne variable	263
10.2.1	Tal og enheder via siunitx	263
	<i>Formatering af tal</i> 264 · <i>Formatering af enheder alene</i> 265 · <i>Værdier med enhed</i> 265	
10.3	Komma som separator i decimaltal	268
10.4	Navngivne variable og enheder	268
	<i>Navngivne variable</i> 268	
10.5	Versionskontrol	269
10.6	Normalsider og anslag	271
10.7	Elektroniske slides	272
	<i>Layers</i> 274 · <i>Sætninger</i> 275 · <i>Opdeling af præsentationen</i> 276 · <i>Animationer</i> 277 · <i>Handouts</i> 277 · <i>Omvej</i> 278 · <i>Transparenter</i> 278	
10.8	Statistik	278
	<i>Statistikprogrammer</i> 280	
10.9	Lidt fysik og kemi	281
	<i>Kemiske forbindelser</i> 281 · <i>Grafik til fysik, kemi og elektronik</i> 283	
10.10	Beskrivelse af algoritmer	283
10.10.1	Brug af algpseudocode-pakken	283
10.10.2	Struktur konstruktioner	284
10.10.3	Flydende algoritmer	285
10.11	Interne hyperlinks	285
	Vedrørende større projekter	
11	Håndtering af større projekter	291
	<i>Hvordan man håndterer projekter i flere filer, samt andre tips til behandlingen af store projekter.</i>	

Indholdsfortegnelse

11.1	Opdeling af kildekoden	291
	<i>\input 292 · \include 292 · Master/hoved-filen 293</i>	
11.2	Ekskludering af dele af teksten	294
11.3	Arbejds kommentarer	295
	<i>Pakken fixme 296 · Pakken todonotes 297</i>	
11.4	At holde styr på labels og lignende	297
	<i>Vis indsatte og anvendte label- og citeringsnøgler 297 · Lidt smartere krydsreferencer 298</i>	
11.5	Referere til label fra et andet projekt	299
12	Dokumentopsætning med memoir-klassen	301
	<i>Kort om hvilke dele en bog normalt er bygget op af og hvad de består af. Dernæst en gennemgang af hvordan man kan justere layoutet til et speciale med memoir-klassen.</i>	
12.1	Pakker allerede inkluderet i memoir-klassen	301
12.2	Opbygningen af en bog	302
	<i>Titelmateriale 302 · IMF krav til forsiden 303 · Frontmatter 303 · Mainmatter 304 · Backmatter 304</i>	
12.3	Specialelayout med memoir – casestudy	305
12.3.1	Layoutformulering	305
12.3.2	Dokumentets dimensioner	306
12.3.3	Sidehoved og -fod	308
	<i>Opsætning af marks 311 · Håndtering af kapitaler i marks 312</i>	
12.3.4	Justering vedr. overskrifter	313
12.3.5	Afsnitsnummerering og justering af indholdsfortegnelsen	314
12.3.6	Forside med memoir	315
	<i>Louises forside 316 · Strits forside 318 · Centrering af forside i forhold til papiret 320 · En kolofon bag på forsiden 320</i>	
12.3.7	Frontmatter	320
	<i>En dedikationsside 321 · Resumé på flere sprog 321</i>	
12.3.8	Mainmatter	322
12.3.9	Backmatter	322
12.3.10	Eksempelfil for specialeopsætning med memoir	322
12.4	Omjustering af layoutet i forbindelse med andre typer større opgaver	323
12.4.1	Et par ideer vedr. ph.d.-afhandlinger	323
	<i>Afhandling: Artikler samlet i samme dokument 323 · Afhandling: Artikler inkluderet som PDF 324 · Inklusion af PDF med en master som ikke tåler pdf_latex 324 · Pagestyle til artikler i afhandling 325</i>	
12.4.2	Tanker omkring opsætning af bachelorprojekter	326
12.4.3	Større opgaver i gymnasiet	326
12.5	Andre tricks man kan lave med memoir-klassen	327
	<i>Automatisk fjernelse af sidetal 327 · Sidetal af formen side / total 328 · Få ordet Kapitel eller Appendix med i indholdsfortegnelsen 328 · Fjerne afstand i figur/tabelliste 328 · Appendix OG bilag 328 · Referere til en overskrift 330 · Markere noget nyt via en blank linie 330</i>	
13	Generelle gode råd vedr. forskellige projekter	331
	<i>Forskellige gode råd når man skriver opgaver i L^AT_EX.</i>	
13.1	Dødssynder	331
13.2	Værd at huske mens man skriver	332
13.3	Værd at huske når man er ved at være færdig med at skrive	333

Indholdsfortegnelse

13.4	Til noteskribenter	334
13.5	Til artikelskribenter	334
14	Om justering af standardklasserne	337
	<i>Kort om et par generelle metoder til at kunne styre noget af opsætningen af standardklasserne i L^AT_EX</i>	
14.1	Opsætning af marginer	337
	<i>B5 på A4</i> 339	
14.2	Sidehoved og -fod	340
	<i>Begrebet pagestyle</i> 340 · <i>Justering af pagestyle med pakken fancyhdr</i> 341 · <i>Standardklassernes problemer vedr. sidehoved og -fod</i> 344 · <i>For lang headertekst</i> 345	
14.3	Indholdsfortegnelsen	345
	<i>At få bibliografi og indeks med i ToC</i> 347 · <i>Konfigurering af ToC</i> 347	
14.4	Overskrifter	347
	<i>Afsnitstitler uden nummer men med i ToC</i> 348	
	Yderligere materiale	
A	Om generering af nye makroer	351
	<i>Generelt om hvordan man laver sine egne simple makroer, både med og uden argumenter.</i>	
A.1	At lave sin egen makro	351
	<i>Navne på makroer</i> 351 · <i>Makroer uden argumenter</i> 352 · <i>Makroer med argumenter</i> 353 · <i>Lave om på en allerede eksisterende makro</i> 354 · <i>Lave en makro hvis den ikke findes i forvejen</i> 354 · <i>Lave om på en makro mens man genbruger den gamle definition</i> 355 · <i>Makroer specielt til teksten</i> 355	
A.2	Hvor gælder ændringer af makroer, længder etc.?	356
A.3	Lave sit eget environment	357
A.4	Makronavne med @	358
A.5	Om at lave sin egen pakke	358
A.6	Makroer med avancerede argumenter	359
A.7	Andre ting værd at kende vedr. makroprogrammering	360
B	Nyttige metoder og konstruktioner	363
	<i>Makroer og andre værktøjer man kan anvende i konstruktionen af nye makroer.</i>	
B.1	Om længder	363
	B.1.1 Nyttige længder	365
B.2	Om tællere	365
	B.2.1 Den grafiske repræsentation af en tæller	366
	B.2.2 Om at ramme tællerværdier i errata eller præsentationer	367
B.3	Betingedekonstruktioner (if-then)	367
	B.3.1 Sammenligninger	367
	B.3.2 Boolske udtryk	368
	B.3.3 Andet fra etoolbox	369
B.4	Luftkasteller	369
	B.4.1 Lodret	369
	B.4.2 Vandret	369
	B.4.3 Elastiske konstruktioner	370
B.5	Kassetækning	371

Indholdsfortegnelse

B.5.1	Kasser med linieombrydning	371
B.5.2	Inline kasser (uden linie ombrydning)	372
B.6	Et par makroer fra graphicx-pakken	373
B.7	Gemme/undersøge indhold	374
C	L^AT_EX-distributioner samt andre redskaber	377
	<i>Kort om L^AT_EX-distributioner, editorer samt andre nyttige redskaber.</i>	
C.1	L ^A T _E X-distributioner	377
C.1.1	T _E X Live	377
	<i>T_EX Live på Mac 377 · T_EX Live på Windows 378 · T_EX Live på Linux 380 · Andet vedr. T_EX Live 383</i>	
C.1.2	MikT _E X	384
	<i>ProTeXt 384 · Ren MikT_EX 384 · Opdatering af MikT_EX 385</i>	
C.2	Editorer	386
	<i>TeXworks 386 · TeXMaker 388 · TeXstudio 389 · TeXnicCenter 389 · TeXShop 389 · Gnu Emacs+AUCT_EX+refT_EX 390 · Aquamacs 391 · WinEdt 391 · Notepad++ 391 · TeXmacs 391 · Led 392 · Kile 392 · Lyx 392 · Scientific Workplace 392</i>	
C.3	Fremvisere	393
	<i>Adobe Reader 394 · Sumatra PDF 394 · Skim 395 · Foxit 395 · Evince 395 · Okular 396 · Xpdf 396 · PDF-Xchange Viewer 396 · GV 396 · Gsview32 397 · Xdvi 397 · Dviout 397 · Yap 397</i>	
C.4	Andet	397
	<i>Et par nyttige redskaber på Linux 397 · Printerdrivere 398</i>	
D	Om fejlmeddelelser	399
	<i>En oversigt over relevante meddelelser, hvad de betyder og hvad man skal gøre.</i>	
D.1	Fejlmeddelelser	399
D.2	Advarsler	403
D.2.1	Ikke helt så alvorlige advarsler	404
	<i>Font advarsler 404 · Underfull advarsler 404</i>	
D.2.2	Advarsler som man bør rette	405
	<i>Labels og citeringer 405 · Den notoriske ‘overfull \hbox’ 405</i>	
E	Kommentarer	407
	<i>Nogle kommentarer som de interesserede kan tage et kig på, men som ikke helt passer i den almindelige tekst.</i>	
E.1	The Intent to indent	407
E.2	Hjælp min vejleder vil have min afhandling i Word!	408
E.3	Matematik bemærkninger	409
	<i>Lidt mere om align til højre for lighedstegn 409 · Matematisk mellemrum mellem symboler 409 · Forkert mellemrum efter \left(eller \big(410</i>	
E.4	Makrobemærkninger	411
E.5	Lidt om filtyperne anvendt af L ^A T _E X	411
E.6	Lidt om de pakker som anvendes til at lave denne bog	412
E.7	Ordbog	415
E.8	Pakker som <i>ikke</i> anbefales	416
F	Skabeloner	419
	<i>Skabeloner til lidt af hvert</i>	
	Litteratur og pakkeinformation	421
	Stikordsregister	433

Tabeloversigt

Tabel 1.1	Taster på tastaturet med speciel fortolkning.	9
Tabel 2.1	Makroer til at lave afsnitsoverskrifter	23
Tabel 2.2	Anførelstegn i \LaTeX	26
Tabel 2.3	Typografiske streger i \LaTeX .	26
Tabel 2.4	Accenter og specialbogstaver i \LaTeX	30
Tabel 2.5	Nogle tekstsymboler fra textcomp-pakken	30
Tabel 2.6	Fontstørrelser i \LaTeX	31
Tabel 2.7	\LaTeX s metoder til skift af skrifttype variant.	32
Tabel 3.1	Makroer til indsættelse af vandret mellemrum	46
Tabel 3.2	Græske bogstaver	52
Tabel 3.3	Nogle operatorsymboler	53
Tabel 3.4	Nogle relationssymboler	53
Tabel 3.5	Nogle inklusioner	53
Tabel 3.6	Diverse symboler	53
Tabel 3.7	Nogle forskellige pile-symboler	55
Tabel 3.8	Nogle store symboler	56
Tabel 3.9	Matematiske alfabetfonte	58
Tabel 3.10	Prædefinerede navngivne operatoren i \LaTeX	60
Tabel 3.11	Dots i \LaTeX	62
Tabel 3.12	Matematiske accenter	63
Tabel 3.13	Indhegnings- og afgrænsersymboler	65
Tabel 7.1	Almindelig nybegynder datatabel	167
Tabel 7.2	Forsøg på at lave en professionel udgave af Tabel 7.1.	167
Tabel 8.1	Fordele og ulemper ved pakkerne multibib og bibtopic.	218
Tabel 9.1	Standardnavne i \LaTeX (engelske)	230
Tabel 10.1	Nogle enheder	266
Tabel 10.2	Enheds prefixer	267
Tabel 10.3	Nogle sammensatte og forkortede enheder	267
Tabel 10.4	Predefinerede potenser	267
Tabel B.1	Længdeenheder i \LaTeX	363
Tabel B.2	Nyttige længder i \LaTeX	365
Tabel B.3	Hjælpemakroer til at skrive værdien af en tæller.	366
Tabel B.4	S/M for tællere.	366

Figuroversigt

Figur 1.1	Speciel tast på et dansk tastatur	9
Figur 12.1	Oversigt over dækark og forsideark	302
Figur 12.2	Forsiden til Louises bachelorprojekt.	317
Figur 12.3	Forsiden til Strits speciale.	319
Figur 14.1	Oversigt over fortolkningen af options til geometri-pakken.	338

Forord

For de fleste er L^AT_EX bare et redskab i hverdagen, det er ikke noget man gider bruge alt for meget tid på at sætte sig ind i eller følge med i udviklingen af. Dette er selvfølgelig meget forståeligt, man er ikke på universitetet for at lære L^AT_EX, man er der for at arbejde. Dette har desværre en kedelig bivirkning: Da man ikke følger med i udviklingen har man i stedet tendens til, at har man fået fat i en guide/introduktion til L^AT_EX, da holder man sig slavisk til konstruktionerne i denne. Man kommer så at sige aldrig videre. Derfor er det altid sjovt at se når folk finder ud af hvad L^AT_EX faktisk kan, når man lige ved hvordan.

Ideen er derfor at idet læseren for de flestes vedkommende kun kommer til at anvende den/de note(r) de anvendte da de lærte at bruge L^AT_EX, da skal denne note indeholde alt det skribenten har brug for (indenfor rimelighedens grænser). Det er derfor at denne bog er ca. 495 sider lang og at der er 633 nummererede eksempler. Men faktisk er det meningen at man skal supplere bogen med [Wilson \(2010\)](#) (manualen til dokumentklassen *memoir*, som anbefales til større værker) for så at kunne skrive (og konfigurere), f.eks. et speciale, uden ret megen brug af ekstra bøger eller pakkedokumentation.



Den nuværende udgave af bogen er tredje udgave.¹ Der er kommet en del ekstra ting med, og en hel del stave- og formuleringsfejl skulle nu være rettet. Desuden er der blevet gjort en hel del for at forsøge at gøre bogen nemmere at forstå, samt forklare tingene på en mere gennemskuelig måde. Skulle der være ting som burde forklares bedre, er man velkommen til at sende en mail.

Emner som ikke gennemgås i denne bog

Da hovedformålet med »*Introduktion til L^AT_EX*« er rettet mod den *trykte* version af specialer og bachelorprojekter (og lignende) er der visse emner vi behandler i mindre detaljer end andre:

- Hyperlinkede PDF-filer. Konstruktionerne er kort beskrevet i afsnit [10.11 på side 285](#).
- Elektroniske slides og præsentationer, vi har dog en beskrivelse af den mest anvendte klasse til præsentationer (*beamer*), se afsnit [10.7 på side 272](#).

Der er også emner som kan være relevante, men som vi slet ikke behandler da der ikke er plads:

- Generering af (ekstern) grafik til brug indenfor L^AT_EX. Der er lavet en kort oversigt over et par muligheder i afsnit [6.6.6 på side 161](#).

¹ De første to udgaver bestod af et alt for langt sæt slides, samt en omskrivning af disse.

Forord

- Avanceret makroprogrammering, det er til tider lidt en kunst i sig selv. Man lærer meget hen ad vejen samt ved at kigge i kildekoden til andre pakker og klasser.

Bogen handler om brugen af L^AT_EX, men ikke om selve anskaffelsen, installationen og vedligeholdelsen af en L^AT_EX-installation. Dette er meget forskelligt fra distribution til distribution, så det ville fylde alt for meget, læs i stedet dokumentationen til den distribution man nu en gang har valgt at anvende. Vi har dog inkluderet nogle tips i Appendix C på side 377.

Indholdet af bogen

Bogen er bygget op i tre dele plus appendix, litteraturliste og indeks.

Første del, »*Grundlæggende L^AT_EX*«, er en introduktion til helt nye L^AT_EX-brugere. Her vil de mest basale ting blive forklaret. Vi vil se lidt på hvordan man skriver almindelig tekst og vi vil kigge på hvordan man skriver matematiske konstruktioner i L^AT_EX. Indholdet af disse kapitler anses for at være standardviden og er derfor noget alle L^AT_EX-skribenter bør vide, hvad enten de er nybegyndere eller øvede.

Der er forskellige ting som ikke er inkluderet i den introducerende del, dette omfatter sætningskonstruktioner (til sætninger, definitioner, lemmaer m.m.), inklusion af grafik, tabeller og litteraturliste. Forståelse for disse konstruktioner bør vente til man lige er blevet vant til syntaksen vi anvender indenfor L^AT_EX.



Anden del, »*Overbyggende L^AT_EX*«, er beregnet for brugere som nu har snuset lidt til L^AT_EX. Hele den introducerende del anses nu for at være noget man godt ved. Vi starter her med at kigge på flere forskellige matematiske konstruktioner, startende med en gennemgang af anbefalinger til hvordan man lineombryder matematik.

Kapitel 5 på side 115 behandler konstruktioner til at lave sætninger, lemmaer, osv. Desuden behandles forskellige måder hvorpå man kan konfigurere disse konstruktioner til ens egne behov. I kapitel 6 på side 135 betragtes inklusion af ekstern grafik samt håndtering af begrebet *floats* samt konfigurering af *caption* (billedtekst). Kapitel 7 på side 165 går så over til tabeller, hvor vi for det første gennemgår nogle af de muligheder man har vedrørende opsætningen af en tabel, men forsøger også at komme med gode råd om hvordan man bør sætte datatabeller op således de bliver pæne at se på. Litteraturliste behandles i kapitel 8 på side 197.

Del II afsluttes med et ekstra kapitel omkring tekst (lister, fonte etc.) samt et 'diverse'-kapitel med relevante ting som ikke rigtigt hører til andre steder.



Tredje del, »*Vedrørende større projekter*«, beskæftiger sig med håndteringen af større projekter. Lige fra et kig på generelle metoder til at justere på layoutet, over hvordan man strukturerer sit projekt på en smart måde, til hvordan man sætter specialer og lignende op med klassen *memoir*.



Bogen afsluttes med nogle appendixer hvor især Appendix A på side 351 og Appendix F på side 419 vil være i læserens interesse. Den første forklarer hvordan man laver sine egne makroer. F.eks. anbefaler vi at når man f.eks. anvender den samme matematiske mængde mange gange, så bør man lave en makro til at skrive den med, så er det nemmere at både skrive teksten samt justere senere. Et andet appendix er skabeloner til forskellige ting (indtil videre ret tomt), disse kan alle downloades hvilket vi forklarer mere om i afsnittet »*Om bogens syntaks*«.

Om bogens syntaks

Appendix D på side 399 bør man også kende til, der finder man en gennemgang af en masse fejlmeddelelser og advarsler man vil komme ud for i L^AT_EX.

Om bogens syntaks

L^AT_EX-syntaksen bliver forklaret senere i bogen, her er kort om de konventioner vi har valgt at anvende i bogen.

Alle kode eksempler er sat med en »fastbredde skrifttype«. I en syntaksomtale er selve makronavnene skrevet med samme skrifttype. Her indgår ofte termen *⟨tekst⟩*, som betyder at her skal man (måske) selv indsætte noget. Tit vil *tekst* være erstattet af en mere sigende tekststump. Et tænkt eksempel kunne være:

```
\foo[⟨valgfrit argument⟩]{⟨obligatorisk argument⟩}
```

Som demonstrerer en makro, »\foo«, og måden hvorpå man i L^AT_EX angiver et obligatorisk og et valgfrit argument.²

Visse ord, termer og især *makronavne* er blevet markeret i teksten for skelne dem fra resten. Disse ord er til tider gentaget i marginen. Eksempelvis formateres pakkenavn som: »pakkenavn«, hvor navnet her er gentaget i den ydre margin. Ofte vil disse termer være at finde i indekset.

pakkenavn

Til fremvisning af (kode)eksempler anvender vi varianter af:

Et eksempel.

Et eksempel.

**Eksempel
0.1**

– hvor man har kildekoden til eksemplet på venstre side og resultatet i kassen til højre.

Dette kan varieres således at vi f.eks. kun har kode delen (ofte anvendt til at fremvise konfigurationsmuligheder) eller kun resultatet (med eller uden kasse). Eventuelt kan koden og kassen stables, hvis situationen kræver det, i dette tilfælde kan eksemplet deles mellem de to dele. Desuden kan kildekoden i visse tilfælde strække sig over flere sider (anvendes primært når vi når til skabeloner).

Næsten alle eksempler (med kode) har et nummer og overskriften »Eksempel« placeret i den ydre margin. Der er yderligere to fremvisningstyper mærket hhv. »Kodetip« og »Skabelon«, disse er *altid* kode.

Eksempelkonstruktionen har en søster som vi ser herunder.

```
\usepackage{mflogo}
```

— slut preamble —

```
\MP
```

METAPOST

**Eksempel
0.2**
📄

Her er tale om eksempler, hvor læseren skal skrive noget i sin preamble – hvilket er det som står ovenover strengen:

— slut preamble —

Det som står under linien (hvis der er noget) skal brugeren så placere i det indre af sit dokument (i.e. efter `\begin{document}`).³ Ved visse eksempler ser man kun det som skal skrives i preamble.

² Valgfrit betyder i denne henseende at `\foo[1]{⟨tekst⟩}` er det samme som `\foo{⟨tekst⟩}`, det er altså valgfrit om man ønsker at tage den med.

³ Rent teknisk bliver disse eksempler faktisk lavet via eksterne filer – det som står i eksemplet gemmes i en ekstern fil, preamble-delen placeres i preamble og teksten under strengen placeres i kroppen af dokumentet. Dette dokument bliver herefter oversat og omdannet til et billede som er det resultat man ser.



Forord

Eksemplerne (og lignende) markeres som det ses i marginen med en overskrift og et nummer. Hvis der under nummeret står symbolet



så betyder det, at man kan downloade kildekoden til dette eksempel via

<http://www.imf.au.dk/system/latex/bog/eksempler/>

Man skal bare lede efter filen »Eksempel- \langle nummer \rangle .tex«, »Kodetip- \langle nummer \rangle .tex« eller »Skabelon- \langle nummer \rangle .tex«. Hvis man læser PDF-udgaven af bogen skal man bare trykke på »«, så skulle man (hvis PDF-fremviseren er konfigureret til det) automatisk blive ført til den rigtige fil. På denne måde er man fri for selv at skulle skrive en større mængde kode af fra bogen. Hvis der er tale om et kodetip og der ved siden af »« står symbolet



betyder det at denne makro kan man også få adgang til ved at installere pakken

`dlftxbcodetips`

`dlftxbcodetips`

så er man fri for at skulle kopiere kode. Pakken kan hentes via CTAN.⁴

Henvisninger til ting på CTAN skrives på formen CTAN: \langle sti \rangle , som så skal opfattes som [http://www.ctan.org/tex-archive \$\langle\$ sti \$\rangle\$](http://www.ctan.org/tex-archive/sti) , dermed skal CTAN: [/macros/latex/contrib/dlftxb/](http://www.ctan.org/tex-archive/macros/latex/contrib/dlftxb/) skal læses som URLen

<http://www.ctan.org/tex-archive/macros/latex/contrib/dlftxb/>

I PDF-udgaven af bogen er disse URLer aktive og peger det rigtige sted hen.

Bemærk desuden at vi fra tid til anden vil fremhæve brugen af mellemrum i et eksempel. Dette markeres traditionelt via symbolet »_«.

Sidebemærkning. Forskellige kommentarer bliver sat i en farvet kasse. Disse kasser kan fortsætte hen over et sideskift. Hvad kommentaren handler om, fremgår oftest af titlen på kassen. Dette kan være bemærkninger, tips, advarsler etc.

Kommentarer til bogen

Kommentarer og forslag modtages gerne på

`daleif+latex@imf.au.dk`

Bemærk at jeg skriver altså »*linie*« med »i« og ikke med »j«, dette er en stille protest og dermed ikke en fejl!

Sidebemærkning. I den version du sidder og læser netop nu, finder man fodnoter på visse underlige steder som her »⁵«, dette er noter til mig som om noget som skal fikses. De bliver stille og roligt fjernet hen ad vejen.⁶

⁴ <http://www.ctan.org>, det centrale register for L^AT_EX-pakker og andre godbidder.

⁶ I den nyeste version af `fixme` fungerer margin noter ikke videre godt, så jeg måtte skifte til at markere med fodnoter.

Takkeskrivelse

Takkeskrivelse

Der er selvfølgelig nogle mennesker der skal takkes i forbindelse med et sådant projekt. Enhver som på en eller anden måde bidrager til bogen (om det så bare er relevante bemærkninger eller rapporter om stavfejl) bør naturligvis takkes. I alfabetisk rækkefølge:

Chadi Abdel-Halim, Hans Harhoff Andersen, Henrik Brix Andersen, Camilla Mondrup Andreassen, Qaim Ali Bakhsh, Jakob Blaavand, Michel Bovani, Christian Bruun-Ghalbia, Jeppe Brønsted, Preben Blæsild, Lennart Borregaard, Jonas Bæklund, David Carlisle, Peter Christensen, Kaja Christiansen, Johan Clausen, Jules Colding, Ole Dalsgaard, Ulrich Diez, Simon Laing Ebbensgaard, Jens Kristian Egsgaard, Brian Elmegaard, Mikkel Kamstrup Erlandsen, Søren Galatius, Per Graa, Jørgen Granfeldt, Enrico Gregorio, Henrik Christian Grove, Rasmus Handberg, Kim Granly Hansen, Sveinung Heggen, Martin Heller, Karl Ove Hufthammer, Mikkel Høgh, Morten Høgholm, Malthe Høj-Sunesen, Jens Kusk Block Jacobsen, Jonatan Jakobsen, Mads Jensen, Marie 'strit' Jensen (nu Janussen), Morten Juhl Johansen, Arne Jørgensen, Anders Kock, Morten Keblovszki, Jesper Kjær, Johan Kraft, Jonas Kölker, Sebastian Lange, Mogens Larsen, Thomas Alexander Lassen, Ole Laursen, Bo Lind, Emil Hedevang Lohse, Enrico Gregorio, Matias Langgaard Madsen, Patrick Kofod Mogensen, Birger Nielsen, Rasmus T. Nielsen, Jonas Nyrup, Heiko Oberdiek, Morten Ovi, Igor Pechtchanski, Helge Møller Pedersen, Jørn Skoven Pedersen, Lars Petersen, Markus Petersen, Mikkel Stouby Petersen, Lan Thuy Pham, Poul Riis, Will Robertson, Jan Rolandsen, Morten Schaumburg, Joachim Schrod, Jens Siegstad, Henrik Stetkær, Mads Sørensen, Jean C. B. Thomsen, Louise Pold Thomsen, Jørgen Vesterstrøm, Jes Vestervang, Lasse Vilhelmsen, Rasmus Villemoes, Thorbjørn Falk Villesen, Peter Wilson, Joseph Wright, Jakob Wrigley, Vegard Øye, Christian Aastrup,

Andre:

Center (og det senere Netværk) for Matematisk Fysik og Stokastik (MaPhySto), T.N. Thiele Centret for Anvendt Matematik i Naturvidenskaberne, Center for Topologien og Kvantisering af Modulirum (CTQM), Institut for Matematik – Aarhus Universitet, brugerne på nyhedsgrupperne comp.text.tex og dk.edb.tekst, brugerne af DK-TUGs mailingliste, brugerne af L^AT_EX-forummerne på <http://www.studieportalen.dk> samt <http://eksperten.dk>.

*Lars »daleif« Madsen
13. september 2011*

Grundlæggende L^AT_EX

Den forunderlige \LaTeX niske verden

1.1 Introduktion

Lad os starte med en gang for alle at forklare hvad \LaTeX er og hvad det især *ikke* er.

1.1.1 Lidt historie

Historien bag \LaTeX går faktisk helt tilbage til midten af halvfjerdserne hvor professor Donald E. Knuth (berømt matematiker og datalogipioner) begyndte arbejdet på \TeX (som er grundsystemet \LaTeX bygger ovenpå). Knuth var svært utilfreds med den typografiske kvalitet forlagene/journalerne leverede når de skulle trykke hans artikler og bøger.

*“The worst of it was the spacing, the way the letters would jam up against each other,” Knuth says. “It was like if you took every letter and you wiggled it and made some of them go up and some of them go down. It wasn’t random—it was systematically bad.” Because the letters in some words got smooshed together, it gave them the illusion of being darker than the others. The eye is naturally drawn toward dark spots, so the reader’s focus would jump all over the page. By 1976, when it was time to print a second edition of Volume 2, Knuth could no longer stand to look at his own work, and he felt that other scientists were getting a similarly raw deal. “We didn’t want our papers just to be there, we wanted them to be beautiful,” he protests. “I wouldn’t have wanted to write *The Art of Computer Programming* if it was going to look ugly.”*

citeret fra <http://www.stanfordalumni.org/news/magazine/2006/mayjun/features/knuth.html>

Knuths løsning på problemet var makrofortolkeren \TeX og skrifttypeformatet METAFONT.¹ Arbejdet tog over 10 år, involverede blandt andet studier i godt typografisk design, se Knuth (1986).

Det kan desværre være meget bøvlet at arbejde i ren \TeX (også kendt som *plain \TeX*) så i midten af firserne samlede Leslie Lamport en hel masse nyttige makroer, han mente folk sikkert ville kunne få brug for. Denne samling blev oversat til et såkaldt *format* (hvilket gør \TeX adgang til dem hurtigere), i dette tilfælde kaldet \LaTeX . Så, når man anvender \LaTeX så anvender man faktisk en \TeX -variant hvor man har forindlæst formatet \LaTeX .

Hovedudviklingen af selve \LaTeX -formatet varetages i dag af det såkaldte *\LaTeX -projekt* (<http://www.latex-project.org>). Dog ændrer man ikke så meget ved selve \LaTeX -formatet længere (pga. kompatibilitet), i stedet kommer de fleste udvidelser i dag via pakker, dvs. samlinger af makroer, som man selv indlæser i sit dokument, så står det brugeren frit for at anvende de ekstra makrosamlinger han/hun har brug for. I dag er computerne så hurtige at man ikke ser den store forskel på om noget er lagt ind i et format, eller om det kommer via en pakke.

Da \LaTeX bare er et format oven på \TeX , vil vi anvende \TeX og \LaTeX i flæng i resten af bogen.

¹ METAFONT ses ikke så meget mere, i dag anvender man andre formater til håndtering af skrifttyper.

1. Den forunderlige L^AT_EXniske verden

1.1.2 Om udtalen af L^AT_EX

De fleste nybegyndere (eller andre uforstående) udtaler L^AT_EX som »lateks«. I vores tilfælde er den sproglige udtale dog noget anderledes. Donald Knuth skriver selv om navnet (Knuth, 1986, side 1):

English words like 'technology' stem from a Greek root beginning with the letters $\tau\epsilon\chi\dots$; and this same Greek word means *art* as well as technology. Hence the name T_EX, which is an uppercase form of $\tau\epsilon\chi$.

Da χ kan udtales som et »k«, udtaler vi L^AT_EX som »la'tek«, ofte med tryk på sidste stavelse. Andre (ofte engelsksprogede) steder anvender man en »ch«-lyd i udtalen (som i »loch«), altså »latech« eller »laytech«. Bare så længe man ikke kalder det »lateks« så er det fint nok.

I L^AT_EX-dokumenter kan man skrive T_EX via `\TeX` og L^AT_EX via `\LaTeX`. Andre steder skriver man almindeligvis »LaTeX« for at indikere en forskel i forhold til gummimaterialet som staves med de samme bogstaver.

1.1.3 Hvad er det så for noget?

Den bedste beskrivelse for L^AT_EX er det engelske begreb »typesetting engine«, dette vil sige at L^AT_EX digitalt vil gøre hvad *typografer/sættere* tidligere gjorde når der skulle trykkes bøger eller aviser.

Den gang blev der afleveret en tekst og en besked om at anvende en specifik tekstbredde og -højde (svarer til marginerne) samt hvilken skrifttype der skal bruges. Sætteren ville så dele teksten op i afsnit (paragraffer). Hver af disse blev splittet op i linier som justeret til den ønskede længde, f.eks. ved at pille lidt ved mellemrummet mellem ordene, eller ved at ombryde et ord. Sådant samles hvert afsnit indtil hver side er fyldt op (evt. deles et afsnit fra en side til en anden) og så fremdeles.

Det er dette L^AT_EX gør for os på elektronisk vis. Vi giver den en tekst, som er forberedt på en speciel måde, L^AT_EX jonglerer rundt med den indtil den spytter en PDF ud i den anden ende (vores pendant til en avis eller bog). Den specielle syntaks vi benytter os af gør os i stand til at fortælle L^AT_EX at noget skal formateres som en overskrift, mens andet er en fremhævet formel eller en tabel.



Der er store forskelle mellem L^AT_EX og f.eks. Word eller OpenOffice Writer. For at blive i symbolikken for før er L^AT_EX *kun* sætteren. Der er ingen redskaber til at skrive teksten eller lave stavekontrol af den. Det er ikke sætterens opgave, den slags har han folk til. Tilsvarende i L^AT_EXens verden, det så en frit for hvilket program man anvender til at forberede teksten til L^AT_EX og det står brugeren frit for at anvende den stavekontrol brugeren bedst kan lide (f.eks. en som er indbygget i den editor man anvender). L^AT_EXs eneste opgave er at »typesette« teksten, intet andet.

Dette er noget man lige skal vænne sig til, især hvis man kommer fra en Word baseret verden. Efter noget tid føler de fleste sig mere frigjorte, da de nu kan koncentrere sig om at skrive indholdet og ikke sidde og slås med f.eks. Word hvis man ikke er enig i det den har besluttet sig for at gøre.

Så kort fortalt er opgaver så som stavekontrol, versionskontrol, databaseopslag og en masse af de ting som f.eks. OpenOffice Writer kan lave, overladt til andre mere specialiserede redskaber. Disse redskaber er gode til det de gør så hvorfor skal L^AT_EX gå ind og genopfinde den dybe tallerken?

Kært barn mange navne



Hvad er der så af ulemper ved L^AT_EX? Det er ikke nogen hemmelighed at L^AT_EX er lidt svært at lære og lære at vænne sig til. L^AT_EX lægger sig meget tæt op af hvad der er god typografi, dette er de fleste ikke vant til.² Så er der visse ting man har været vant til at så på som nu skal *aflæres* igen fordi dette ikke er god typografi. Har man anvendt L^AT_EX i noget tid kan man blive overfølsom overfor dårlig typografi. Det er f.eks. meget nemt at se når en artikel i et (matematisk) tidsskrift ikke er skrevet med L^AT_EX.

Fordelene kan være svære at få til at give mening hvis man ikke er vant til at arbejde med L^AT_EX.

- L^AT_EX er gratis! Man kan bruge det til hvad man har lyst, hvor man har lyst og med hvad man nu har lyst.
- Vi arbejder i ren tekst og L^AT_EX-oversættelsen *læser* kun fra vores dokument, det ændrer ikke ved det.
- Sker der en fejl kan man altid redde sig selv.
- L^AT_EX-brugere er normalt meget venligt og hjælper gerne andre brugere i nød.

Dette med at L^AT_EX-dokumenter er i ren tekst, gør det meget nemt at være flere om at skrive på samme projekt. Hvilket ikke altid er lige nemt med f.eks. Word.

1.2 Kært barn mange navne

I skrivende stund findes der faktisk fire forskellige L^AT_EX programmer. De to primære er L^AT_EX og pdfL^AT_EX, hvor vi anbefaler at de fleste anvender pdfL^AT_EX.³

De to andre er XeL^AT_EX og LuaL^AT_EX. Den første stammer egentlig fra MAC-verdenen og er en L^AT_EX-variant som gør en i stand til at anvende de fonte som er tilrådighed som en del af styresystemet.

LuaL^AT_EX bliver efterfølgeren til pdfL^AT_EX og er et forsøg på at løse nogle programmeringsmæssige besværligheder i L^AT_EX ved at indlejre programmeringssproget Lua i L^AT_EX. Man forventer sig meget af dette projekt, men der kommer nok til at gå nogle år før man er helt færdig. De fleste L^AT_EX distributioner har dog allerede en inkluderet en version af LuaL^AT_EX samt en masse redskaber i T_EX Live-distributionen er skrevet om til at vende dette nye system.

Vi vil ikke kommen nærmere ind på XeL^AT_EX eller LuaL^AT_EX.

Da forskellen mellem L^AT_EX og pdfL^AT_EX egentlig bare er output-formatet vil vi i det resten at bogen anvende termen L^AT_EX for at dække over begge. De få tilfælde hvor der er forskel (inklusion af grafik) skal vi nok forklare forskellen.



For at være helt på det rene er der faktisk også et helt andet system kaldet ConT_EXt, dette er *ikke* som L^AT_EX, men snarere en anden tilgang til det at lave en tilgængelig overbygning til T_EX. Mere information kan findes på <http://contextgarden.net>. De fleste T_EX distributioner inkluderer både L^AT_EX og ConT_EXt.

² For en del af standardvalgene i Word, anses ikke for god typografi.

³ Eneste forskel mellem de to er at L^AT_EX ender med at give en fil i formatet DVI, som så f.eks. skal oversættes til PDF. Ved at anvende pdfL^AT_EX går vi direkte til PDF. Faktisk er L^AT_EX og pdfL^AT_EX samme program, det er bare en simpel konfigurerings som afgør om man får DVI eller PDF.

1. Den forunderlige L^AT_EXniske verden

1.3 Hvor får man L^AT_EX fra?

I denne bog er det *ikke* vores opgave at diktere hvilken L^AT_EX-distribution man skal anvende (til Linux og MAC er der dog ikke det store valg), ej heller hvilken man editor man skal gøre brug af. Det er helt i ånden bag T_EX, op til læseren selv at bestemme.

Frygt dog ikke, vi har lagt nogle nogle råd og forslag i Appendix C på side 377.



Advarsel 1.1. Anvender man MikT_EX, så bør man sørge for at opdatere til mindst version 2.9. De tidligere versioner installerede nemlig ikke de danske orddelingsmønstre som standard, det skulle man selv huske, og de fleste glemte det.

Anvender man i stedet T_EX Live, så installeres alle orddelingsmønstrene som standard.

1.4 Om at udføre L^AT_EX

Vi nævnte kort i afsnit 1.2 på forrige side at der faktisk findes fire L^AT_EX'er, vi vil kun koncentrere os om L^AT_EX og pdfL^AT_EX. For de flestes vedkommende vil man *kun* anvende pdfL^AT_EX.

Sidebemærkning 1.2. Almindelig L^AT_EX spytter en såkaldt DVI-fil ud, denne kan fremvises med dertil indrettede programmer, eller den kan konverteres til PostScript via dvips. Denne kan igen fremvises (f.eks. med gsvieview på windows eller evince på Linux) eller konverteres til PDF via ps2pdf.

1.5 Show us the code

Lad os lægge blidt ud og starte med at se på hvordan man får et læsbart resultat ud af L^AT_EX. Herefter vil vi forklare lidt mere om hvad de enkelte ting er. Tag din favorit teksteditor og gem følgende som eks1.tex (eller hvis du er doven, så download det via linket eller via bogens hjemmeside):

Eksempel

1.1



```
\documentclass[a4paper]{article}
\usepackage[latin1]{inputenc}
\usepackage[danish]{babel}
\usepackage[T1]{fontenc}
\usepackage{amsmath,amssymb,bm,mathtools}
\begin{document}
Hej verden her er jeg og her er lidt matematik  $\sum f(n)$  og noget fremhævet
\begin{equation}
\frac{1}{3}\int_0^b f(x)\,dx
\end{equation}
\end{document}
```

Gå nu til en kommandoprompt (et terminalvindue eller en DOS-prompt hvis man benytter Windows) og oversæt nu denne kildefil med⁴

Show us the code

pdflatex eks1.tex

pdflatex

Programmet vil nu skrive en masse tekst i vinduet, sikkert i stil med:

```
tmp> pdflatex eks1.tex
This is pdfTeX, Version 3.1415926-1.40.9 (Web2C 7.5.7)
%&-line parsing enabled.
entering extended mode
(./eks1.tex
LaTeX2e <2005/12/01>
Babel <v3.8l> and hyphenation patterns for english, usenglishmax, dumylang, noh
yphenation, german-x-2008-06-18, ngerman-x-2008-06-18, ancientgreek, ibycus, ar
abic, basque, bulgarian, catalan, pinyin, coptic, croatian, czech, danish, dutc
h, esperanto, estonian, farsi, finnish, french, galician, german, ngerman, mono
greek, hungarian, icelandic, indonesian, interlingua, irish, italian, la
tin, lithuanian, mongolian, mongolian2a, bokmal, nynorsk, polish, portuguese, r
omanian, russian, sanskrit, serbian, slovak, slovenian, spanish, swedish, turki
sh, ukenglish, ukrainian, uppersorbian, welsh, loaded.
(/vol/packages/TL/2008/texmf-dist/tex/latex/base/article.cls
Document Class: article 2005/09/16 v1.4f Standard LaTeX document class
(/vol/packages/TL/2008/texmf-dist/tex/latex/base/size10.clo)
(/vol/packages/TL/2008/texmf-dist/tex/latex/base/inputenc.sty
(/vol/packages/TL/2008/texmf-dist/tex/latex/base/latin1.def)
(/vol/packages/TL/2008/texmf-dist/tex/generic/babel/babel.sty
(/vol/packages/TL/2008/texmf-dist/tex/generic/babel/danish.ldf
(/vol/packages/TL/2008/texmf-dist/tex/generic/babel/babel.def)))
(/vol/packages/TL/2008/texmf-dist/tex/latex/base/fontenc.sty
(/vol/packages/TL/2008/texmf-dist/tex/latex/base/tlenc.def))
(/vol/packages/TL/2008/texmf-dist/tex/latex/amsmath/amsmath.sty
For additional information on amsmath, use the '?' option.
(/vol/packages/TL/2008/texmf-dist/tex/latex/amsmath/amstext.sty
(/vol/packages/TL/2008/texmf-dist/tex/latex/amsmath/amsgen.sty))
(/vol/packages/TL/2008/texmf-dist/tex/latex/amsmath/amssy.sty)
(/vol/packages/TL/2008/texmf-dist/tex/latex/amsmath/amsopn.sty))
(/vol/packages/TL/2008/texmf-dist/tex/latex/amssymb.sty
(/vol/packages/TL/2008/texmf-dist/tex/latex/amssymb/amsfonts.sty))
No file eks1.aux.
(/vol/packages/TL/2008/texmf-dist/tex/latex/amssymb/umsa.fd)
(/vol/packages/TL/2008/texmf-dist/tex/latex/amssymb/umsb.fd) [1{/vol/packages/
TL/2008/texmf-var/fonts/map/pdftex/updmap/pdftex.map}] (./eks1.aux) {/vol/pack
ages/TL/2008/texmf-dist/fonts/enc/dvips/cm-super/cm-super-t1.enc}</vol/packages
/TL/2008/texmf-dist/fonts/type1/bluesky/cm/cmex10.pfb></vol/packages/TL/2008/te
xmf-dist/fonts/type1/bluesky/cm/cmml10.pfb></vol/packages/TL/2008/texmf-dist/fo
nts/type1/bluesky/cm/cmml7.pfb></vol/packages/TL/2008/texmf-dist/fonts/type1/bl
uesky/cm/cmr10.pfb></vol/packages/TL/2008/texmf-dist/fonts/type1/bluesky/cm/cmr
7.pfb></vol/packages/TL/2008/texmf-dist/fonts/type1/public/cm-super/sfrml000.pf
b>
Output written on eks1.pdf (1 page, 28652 bytes).
Transcript written on eks1.log.
```

Her kan vi til slut se, at L^AT_EX har skrevet sit output til eks1.pdf og den har lavet en .log-fil med yderligere information (ud over det den skrev ovenfor). PDF-filen kan så fremvises med

- Adobe Reader, alle platforme⁵
- Sumatra PDF (til Windows, anbefales på det kraftigste), Foxit
- Okular eller Evince (Linux)

⁴ De fleste editorer har en knap man bare skal trykke på. Det bedes man dog undlade i dette tilfælde. Dette med at kunne oversætte et L^AT_EX-dokument gennem DOS eller en terminal, er vigtigt at kunne, idet man så altid har noget at gå tilbage til, hvis der er gået noget helt galt med ens editor. Efter man har lært at anvende en kommandoprompt, kan man sagtens gå over til at oversætte L^AT_EX dokumentet direkte fra sin editor, hvilket også er det man normalt gør.

⁵ Adobe Reader, er god til at vise ting frem, men ikke god i det daglige arbejde, den bruger rigtigt meget hukommelse og kan ikke genindlæse PDF-filer. Vi anbefaler derfor alternative PDF-fremvisere til det daglige arbejde.

1. Den forunderlige L^AT_EXniske verden

På MAC har den mest anvendte editor TeXShop sin egen indbyggede PDF-fremviser. Det samme har den nye TeXWorks-editor (denne editor findes til alle platforme og er den editor vi anbefaler man starter med).

Havde man i stedet anvendt almindelig L^AT_EX via

```
latex    latex eks1.tex
```

får man en lignende besked, men ender altså op med en DVI-fil. Dette kan vises frem eller konverteres til PostScript via

```
dvips    dvips eks1.dvi
```

```
[27]:~/tmp> dvips eks1.dvi
This is dvips(k) 5.94b Copyright 2004 Radical Eye Software (www.radicleye.com)
' TeX output 2005.02.21:0855' -> eks1.ps
<tex.pro><cm-super-t1.enc><texps.pro>. <cmr7.pfb><cmmi7.pfb><cmr10.pfb>
<cmmi10.pfb><cmex10.pfb><sfrm1000.pfb>[1]
```

Her er det vigtige, at resultatet bliver skrevet til eks1.ps. Denne kan så fremvises eller konverteres til PDF via ps2pdf eller Adobe Distiller (hvis man har adgang til dette program).

ps2pdf

Resultatet skulle gerne ligne følgende plus et sidetal (som her er fjernet af pladshensyn).

Eksempel 1.2

Hej verden her er jeg og her er lidt matematik $\sum f(n)$ og noget fremhævet

$$\frac{1}{3} \int_0^b f(x) dx \quad (1)$$

Af ovenstående fremgår det, at fremgangsmåden ved L^AT_EX er, at man først skriver sin tekst i det specielle format som L^AT_EX nu anvender og herefter får et L^AT_EX-program til at fortolke det. Først da, kan man få lov til at se resultatet af det man har skrevet. Dette er noget som forvirrer de fleste i starten, men har man vænnet sig til det er det faktisk meget rart, for man koncentrerer sig mere om det man skal skrive end om hvordan det skal se ud. Hvis man desuden sørger for at lære de mest almindelige genveje til sin editor, så går det faktisk ret stærkt.

1.6 L^AT_EX-syntaks

Ud fra [Eksempel 1.1 på side 6](#) man kan se, at indenfor L^AT_EX har visse tegn en speciel betydning. Dette inkluderer »\«, »\$« og »{ }«. Der er ti af disse tegn, se [tabel 1.1 på modstående side](#).

Et andet tegn som kan volde nogen problemer er tegnet »`« indenfor computerverdenen ofte kaldet »back ping« (teknisk set er det en accent-grave), hvor »'« (apostrof) kaldes ping. Tegnet »`« findes på et amerikansk tastatur på tasten lige under ESC-tasten. Mens det er mere problematisk på et dansk tastatur.

På et dansk tastatur er findes tegnet »`« som øverste accent på tasten med de to accenter (ligger i nærheden af *Backspace*). Tasten kan se ud som på [figur 1.1 på modstående side](#).

Denne tast er *aktiv* i den danske tastaturopsætning, dvs. trykker man »`« og derefter »a« så får man tegnet »à«. Så hvis man direkte ønsker tegnet »`« skal man på et dansk tastatur trykke »`« efterfulgt af *mellemrum*.

Lidt L^AT_EX-terminologi

Tabel 1.1: Taster på tastaturet med speciel fortolkning samt hvordan man så skriver disse tegn hvis man har brug for selve tegnet.

Tegn	Erstatning ^a	Kommentar
\	<code>\textbackslash</code> , <code>\backslash</code> ^b	anvendes i starten af (næsten) alle makronavne
\$	<code>\\$</code> , <code>\textdollar</code>	anvendes i par omkring matematik i teksten
#	<code>\#</code>	angiver argumenter i makrodefinitioner
%	<code>\%</code>	kommentartegn, resten af linien inklusive % ignoreres
&	<code>\&</code>	adskiller søjler i tabeller og markerer alignment (og søjler) matematik
~	<code>\textasciitilde</code> ^c	tie, mellemrum hvor man ikke må lave linieombrydning
_	<code>_</code> , <code>\textunderscore</code> ^d	angiver subscript i matematik
^	<code>\textasciicircum</code>	angiver superscript i matematik (accent-hat forklares senere)
{ }	<code>\{ \}</code>	anvendes om argumenter til makroer

^a Dvs. det som skal skrives hvis man ønsker at skrive dette tegn direkte.

^b Anvendes i matematikmode, se desuden `\setminus`.

^c Skal man i matematik anvende »~« er det et helt andet tegn (`\sim`), selve tegnet »~« må dog gerne anvendes indeni `\url`, se afsnit 2.10 på side 42.

^d Skal man skrive navnet på en fil, og dette navn indeholder »_« da bør man anvende `\url` eller `\path` fra `url`-pakken.



Figur 1.1: Tast på et dansk tastatur, accent-grave (også kendt som back-ping) er det øverste accenttegn.

Desuden skal det lige bemærkes at accenten fundet *nederst* på ovennævnte tast, *ikke* er apostrof! Apostrof findes ved siden af »Enter/Return«. Det nedeste tegn på figur 1.1 (accent-grave) anvendes ikke direkte i L^AT_EX, men med den rette opsætning kan man fint anvende den til f.eks. at lave »é«-tegnet uden brug af L^AT_EX-koder.

apostrof

Tegnet »`« (back-ping eller accent-grave) vil man støde ind i to steder i denne bog. Første gang er konstruktionen »\`e« som giver »è«, altså en speciel måde at få skrevet et accenttegn på. Det andet sted er i et specielt kodeeksempel i afsnit 4.5 på side 105.

1.6.1 Lidt L^AT_EX-terminologi

Lad os vende tilbage til [Eksempel 1.1 på side 6](#) og forklare lidt om den terminologi vi anvender:

- Sekvenser af typen `\langle navn \rangle` kaldes *makroer* eller *kommandoer* *makroer*
- En undtagelse er parret `\begin{\langle navn \rangle} ... \end{\langle navn \rangle}` som udgør et *environment*. Dette er en markering i teksten inden i hvilken L^AT_EX's opførsel ændrer sig. Dette kunne være en fremhævet formel, en tabel, eller som her en liste. *environment*
- Materiale skrevet inden i `{ }` vil oftest fortolkes som et *argument* til en eller *argument*

1. Den forunderlige L^AT_EXniske verden

anden kommando. Hvis der i en specifikation af brugen af en makro f.eks. står `\foo{<tekst>}` betyder det, at `\foo` tager `<tekst>` som argument.

Hvis en makro i en oversigt er skrevet som `\foo{...}` så betyder det at `\foo` tager et *obligatorisk* argument og de krøllede parenteser er i dette tilfælde påkrævet (skal være der).⁶ Indholdet kan dog være tomt.

Nogle makroer/environments tager argumenter andre gør ikke. Environments kan sagtens tage obligatoriske og valgfri argumenter (se næste punkt) på samme vis som makroer, klassiske eksempler er environmentet *thebibliography* (se afsnit 8.2 på side 197) eller floaten *figure* (se afsnit 6.2 på side 142).

Et lidt mere 'eksotisk' eksempel på brugen af `{ }` som argumenter er til f.eks. superscript/hævet skrift `x^{y+z}` hvilket giver x^{y+z} , her er `{y+z}` argument til makroen bestående af tegnet `»^«` (internt vil L^AT_EX erstatte `»^«` med et mere almindeligt makronavn).

Til tider vil man også se `{ }` anvendt enten stående efter makroer som egentlig ikke tager noget argument (se [Eksempel 1.3](#) for et klassisk eksempel). Man kan dog også se et par af krøllede parenteser stå omkring noget uden at blive fortolket som et argument, eksempelvis `{\bfseries ...}`. Her er `{ }`-parret blevet anvendt som *afskærmning*. Det er dog ikke noget vi vil gøre så meget brug af.

valgfri argumenter

(d) Argumenter kan også til tider angives i et `[]`-par – disse kaldes *valgfri argumenter* eller *options*, se evt. [Eksempel 1.1 på side 6](#), linie 1. Om disse gælder, at det er ekstra argumenter som man ikke nødvendigvis behøver anvende. Desuden er `»\foo[]«` (ofte) det samme, som `»\foo«`. Visse konstruktioner kan endda anvende `()` til at angive argumenter. Se f.eks. [Eksempel 7.13 på side 176](#).

preamble

(e) Området mellem `\documentclass...` og `\begin{document}` kaldes *preamble* (linie 1–6 i [Eksempel 1.1 på side 6](#)).

(f) Området mellem `\begin{document}` og `\end{document}` (altså indenfor environmentet *document*) kaldes selve *dokumentet* eller *body*.

Al tekst efter `»\end{document}«` ignoreres af L^AT_EX.

Mange makroer kan godt lide (at spise) mellemrum

Man bør være opmærksom på følgende interessante feature som ses ved de fleste L^AT_EX-makroer der ikke tager noget obligatorisk argument.

Eksempel 1.3

Når man skriver `\LaTeX` spiser makroen `\verb+\LaTeX+_al` mellemrummet op til det næste ord. Dette kan fikses via `\LaTeX{x}`, eller `\LaTeX\ _x` eller `\LaTeX}_x`.

Når man skriver L^AT_EX spiser makroen `\LaTeX` al mellemrummet op til det næste ord. Dette kan fikses via L^AT_EX `x`, eller L^AT_EX `x` eller L^AT_EX `x`.

Man skal altså i disse sammenhænge huske at tage sig sine forholdsregler. Problemet ses dog kun i den almindelige tekst, i det alle mellemrum i matematik-mode automatisk ignoreres.

1.6.2 Dokumentklassen

`\documentclass`

Første linie i et L^AT_EX-dokument er normalt `\documentclass` og argumentet til denne er så den anvendte *dokumentklasse*. Dokumentklassen sørger for at sætte et standard

⁶ Det er ikke altid korrekt, men det skal vi ikke komme nærmere ind på.

Dokumentklassen

layout op, med hensyn til marginer, fontstørrelser, ekstra makroer og lignende. Det er det man måske ville kalde en basisskabelon.

Dokumentklassen tilbyder ofte diverse konfigurationsmuligheder, kaldet *options*, som vil ændre den måde dokumentet opfører sig på. Disse options afhænger naturligvis af hvilken dokumentklasse man har valgt, men klassen inkluderer som oftest punkterne på nedenstående liste.

a4paper

denne option *skal altid* anvendes! Standardopsætning er det amerikanske *letter*-papirformat hvilket man ikke anvender i Europa.

10pt standardstørrelsen af tekstskrifttypen.

11pt giver størrelse på 11 pt.

12pt tilsvarende for 12 pt.

leqno

formelnumre til venstre.

reqno

formelnumre til højre, dette er normalt standarden.

fleqn vil venstrejustere al fremhævet matematik i stedet for at centrere den.

onecolumn

gæt selv hvad denne gør, dette er standarden.

twocolumn

gæt selv. Ting vedrørende dokumenter i flere søjler er ikke noget vi behandler i denne bog.⁷

oneside

standard i visse klasser. Anvendes når resultatet skal trykkes enkeltsidet.

twoside

til dobbeltsidet tryk, standard i f.eks. *book* og *memoir*. Dette bevirker at højre- og venstremarginer automatisk byttes om på alle venstresider.

Bemærk, at det er printerens ansvar at sørge for at lave dobbeltsidet tryk og det er brugerens eget ansvar at sørge for at fortælle printerens dette.

openright

hvis klassen understøtter `\chapter`, betyder denne, at kapitlerne starter på en højreside. Denne er standard i *book*- og *memoir*-klassen.

openany

igen hvis `\chapter` findes, betyder denne, at kapitler starter på næstfølgende side uanset om det er en højre- eller en venstreside. Denne er standard i *report*-klassen.

Tip 1.3. Almindeligvis vil alle options man giver dokumentklassen også blive givet til alle de pakker man anvender (hvad enten de anvender den eller ej). Dette er faktisk meget smart idet man på denne måde kan give en fælles option til mange pakker på en gang. Et eksempel kan være sprogoptionen `danish` (som `babel` jo reagerer på, men det gør `varioref` også). Så kan man altså nøjes med

⁷ Se også pakken `multicol`.

1. Den forunderlige L^AT_EXniske verden

Eksempel 1.4

```
\documentclass[a4paper,danish,...]{...}
\usepackage{babel}
\usepackage{varioref}
...
```

Nogle dokumentklasser

L^AT_EX kommer med fire standardklasser: *article*, *report*⁸, *book*⁸ og *letter*⁹. Almindeligvis kan man anvende *article* til opgaver og lignende.

Standardklasserne er desværre ikke videre fleksible, så når man ønsker at ændre noget bliver man ofte nødt til at load en pakke (forklares nedenfor) og vide hvordan denne pakke fungerer. En anden mulighed er at anvende nogle mere konfigurerbare klasser. Der er to systemer som kan anbefales *memoir* samt klasserne fra KOMA-bundtet. Jeg gør selv kraftigt brug af den første og det er også den som anbefales gennem bogen.

memoir

Meget stor og veldokumenteret klasse. Umiddelbart en afløser til *book*, man kan også sagtens konfigureres til at ligne *report* og endda *article*. Se [Wilson \(2010\)](#), [Wilson \(2007a\)](#) samt [Madsen \(2008b\)](#) og [Madsen \(2003\)](#).¹⁰

scrartcl

Afløser til *article*. Del af KOMA-bundtet.

scrbook

Afløser til *book*. Del af KOMA-bundtet.

scrlettr2

Afløser til *letter*. Denne anbefales hvis man ønsker at skrive breve i L^AT_EX. Del af KOMA-bundtet.

scrreprt

Afløser til *report*. Del af KOMA-bundtet.

For en brugervejledning i brugen af KOMA klasserne se [Kohm og Morawski \(2006\)](#).

Mange journaler indenfor fysik anbefaler klassen *revtex4*. Fælles for alle klasserne er naturligvis at man bør læse de relevante manualer.

1.6.3 Preamblen

11

Selve preambelen anvendes til at tilføje ekstra makroer eller hele samlinger af disse *pakker* – såkaldte *pakker*. Pakker inkluderes i dokumentet via

```
\usepackage[options]{pakkenavn(e)}
```

Eksempel 1.1 på side 6 viser de fem standardpakker, man altid kommer til at anvende (se desuden afsnit 1.9 på side 19):

⁸ *report* og *book* ligner hinanden en hel del.

⁹ Denne kan anvendes til breve, omtales ikke nærmere. Oftest vil *scrlettr2* være et meget mere konfigurerbart valg.

¹⁰ Læser man resten af bogen vil man se at undertegnede er ret glad for *memoir*, men er også en smule forundtaget, da det er undertegnede som vedligeholder *memoir*.

¹¹ FiXme Note: dette bør opdateres pga. TW default til utf8, og andre editorer som ikke kan utf8

```
\usepackage[latin1]{inputenc}
\usepackage[danish]{babel}
\usepackage[T1]{fontenc}
\usepackage{amsmath,amssymb,bm,mathtools}
```



Afsnit 1.6.5 giver en nærmere forklaring på vigtigheden af pakken `inputenc` og hvorfor det er givtigt at vide hvilken option denne pakke *skal* have.

Den første pakke i oversigten sørger for, at vi må skrive æ, ø og å (og andre specialtegn) direkte i kildekoden uden at skulle anvende specialkonstruktioner (se evt. Tabel 2.4 på side 30). Den anden giver dansk opsætning af specielle ord og lignende, aktiverer desuden de danske orddelingsmønstre (se desuden afsnit 2.2 på side 24 samt afsnit 9.3 på side 226). Pakken `fontenc` hjælper bl.a. med orddelingen ved æ, ø og å, desuden findes visse fonte kun i encodingen »T1«. De sidste pakker giver bedre matematikopsætning.

Vi skal senere se på hvordan man laver sine egne simple makroer (se appendix A på side 351).

1.6.4 Selve dokumentet

Imellem `\begin{document}` og `\end{document}` skriver man altså sit dokument.

Tip 1.4. Der gælder den specielle feature, at alt *efter* `\end{document}` *ignoreres*. Dette kan f.eks. være nyttigt når man forsøger at lokalisere fejl i et dokument. Man indsætter så bare `\end{document}` og oversætter igen.

Hvad man ellers skriver her, beskriver vi fra kapitel 2 og frem.

1.6.5 Encoding

Encoding er nok noget de færreste har hørt om før. Men som dansk L^AT_EX bruger er dette ret vigtigt (skriver man kun på engelsk, man kan bare ignorere dette afsnit).

Vi har allerede nævnt at man skal lave en tekst som skal fodres til L^AT_EX. Men tekst er ikke længere bare tekst. Lige så snart man anvender specialtegn, så som é, æ bliver man nødt til at vide *hvordan* disse tegn er gemt, det vi kalder *character encoding*. Et specialtegn kan gemmes på en måde i en encoding, men på en anden måde i en anden encoding. Hvis en editor ikke selv kan finde ud af at detekttere den encoding en fil er skrevet i, eller tager fejl, så ser man ofte dette ved at der vises noget helt andet end f.eks. æøå.

Kort fortalt gemmes en tekst i en fil som en serie af nuller og ettere. Er filen tildelt en *character encoding* så har man fortalt hvilket tegnsæt (mængde af bogstaver, tal og specialtegn) der er anvendt i denne vil, samt hvilken sekvens af nuller og ettere der svarer til hvert tegn.

L^AT_EX anvender denne information således at man uden videre kan skrive æøå i kildekoden i stedet for f.eks. `\ae\o\aa`. Men L^AT_EX kan ikke selv se hvilken encoding en given fil er skrevet i, dette bliver vi nødt til at fortælle L^AT_EX:

```
\usepackage[encoding]{inputenc}
```

Her kommer så problemet, hvordan ved jeg hvilken encoding min fil er skrevet med? Det er faktisk er meget godt spørgsmål. De fleste er i dag på vej over til at bruge *unicode* og specielt det vi kalder UTF-8-encoding. Dette er en stor kompliceret encoding som indeholder en enorm mængde tegn, lige fra dem vi anvender i Vesteuropa

1. Den forunderlige L^AT_EXniske verden

over Østeuropa til Asien. Mange flere end de 256 tegn som der f.eks. ligger i LATIN1-encodingen. Så dette er den encoding vi anbefaler på langt sigt.

Desværre er verden bare ikke så let. Der er flere populære editorer til Windows¹² som endnu ikke understøtter UTF8. Hvilket kan betyde at hvis man sender en L^AT_EX-kildekodefil til en kammerat så kan han/hun ikke fremvise filen korrekt i den editor han/hun anvender. Derfor vil vi normalt anbefale at man anvender encodingen LATIN1¹³ også kendt som ISO-8859-1. Moderne editorer (f.eks. TeXShop til MAC samt TeXWorks) tilbyder brugeren selv at angive hvilken encoding filen skal gemmes med.

Derfor arbejder man på Windows, vil det bedste råd pt. være at man anvender LATIN1 encoding, og håbe på at de forskellige udviklergrupper bag editorerne snart får taget sig sammen og får implementeret UTF8 understøttelse.



Er man kommet i problemer vedr. encodings, enten at man har modtaget en fil som ser underlig ud, enten med at f.eks. »æ« erstattes med et underligt tegn eller $\langle tal \rangle$, eller at L^AT_EX brokker sig over f.eks.

```
! LaTeX Error: Command \textbrokenbar unavailable in encoding T1.
```

Så skal man enten til at rode med sin editor (evt. finde en anden editor), eller hvis det er det sidste tilfælde, så skal man angive en anden encoding.

Der er umiddelbart 2 plus 2 encodings vi skal forholde os til. De to mindre anvendte er

applemac

Standard encoding for TeXShop editoren. Af hensyn til udveksling af filer, vil det være en fordel at omkonfigurere TeXShop til at anvende LATIN1 eller UTF8 encoding i stedet.

ansinew

Windows standard encoding, i de fleste tilfælde vil man her også kunne klare sig med LATIN1.

De to mest almindelige er

latin1

Dækker de Vesteuropæiske sprog, dvs. indeholder é, ö etc. Dette er den de fleste anvender.

utf8 Dette er fremtidens encoding, inkluderer Vest- og Østeuropæiske sprog, samt diverse andre i et hug. Desværre er der en del populære L^AT_EX editorer til Windows som *ikke* forstår UTF8, hvilket er en af grundene til at vi anbefaler at man omkonfigurerer TeXWorks editoren til at gemme i LATIN1.

Man bliver altså nødt til selv at finde ud af hvilken encoding ens editor er konfigureret til at anvende. Får man problemer, så prøv at angive en anden encoding i L^AT_EX-dokumentet.

Efter en længere diskussion er vi blevet enige om at anbefale følgende når man anvender UTF8 som inputencoding i sin editor:¹⁴

¹² F.eks. TeXnicCenter.

¹³ Evt. LATIN9, hvilket er latin1 hvor man har erstattet et par mindre anvendte tegn med bl.a. Euro tegnet.

¹⁴ Der er også en anden encoding angivelse og en ekstra pakke, som man ser i noget litteratur på nettet, men disse anbefales ikke.

Når noget er gået galt

```
\usepackage[utf8]{inputenc}
```

Eksempel
1.6

Forudsat at man har en L^AT_EX-distribution som ikke har flere år på bagen. Så skulle man uden problemer kunne anvende de almindelige specialtegn i UTF8. Der er naturligvis ikke tale om en unicode implementation¹⁵, så UTF8 konfigurationen giver kun adgang til de tegn som inputenc i forvejen giver adgang til.

1.7 Når noget er gået galt

Der er *ingen* som skriver perfekt kode, så alle vil komme ud for problemer ved oversættelsen mellem kildekoden og outputtet. Derfor er det meget nyttigt at lære hvordan man debugger et L^AT_EX-dokument når der opstår en fejl.

Den største kilde til problemer for en nybegynder (og andre) er når man staver makronavne forkert. Prøv f.eks. at gemme følgende som eks2.tex og kørs det igennem L^AT_EX.

```
\documentclass[a4paper]{article}
\begin{document}
hmm \foo virker ikke
\end{document}
```

Eksempel
1.7

hvilket giver

```
daleif@daleif:~/tmp/mb> latex eks2.tex
This is TeX, Version 3.14159 (Web2C 7.3.7x)
(./eks2.tex
LaTeX2e <2001/06/01>
Babel <v3.7h> and hyphenation patterns for english, dumylang, nohyphenation, danish, ukenglish, loaded.
(/usr/TeX/texmf/tex/latex/base/article.cls
Document Class: article 2001/04/21 v1.4e Standard LaTeX document class
(/usr/TeX/texmf/tex/latex/base/size10.clo))
No file eks2.aux.
! Undefined control sequence.
l.3 hmm \foo
      virker ikke
?
```

L^AT_EX fortæller her, at der er en kontrolsekvens (en makro), som den ikke kender (»*Undefined*«) og denne befinder sig på linie 3 i eks2.tex. For at komme videre med oversættelsen kan man taste:

x for at stoppe oversættelsen her. Så kan man rette fejlen og starte oversættelsen igen.

h for at få hjælp (hvis der er nogen).

q for at fortsætte oversættelsen og ignorere *alle* fejl.

RETURN/ENTER for at lade L^AT_EX lave et (intelligent) forslag og lade kompileringen fortsætte, så kan man rette fejlen senere. Anvendes desuden hvis man bare ønsker at ignorere denne fejl, måske fordi man ikke lige selv kan finde ud af at rette fejlen.

I eksemplet er makroen »\foo« naturligvis problemet da der ikke automatisk findes en makro med dette navn.

¹⁵ I unicode findes stort set alle tegn fra alle sprog i hele verden.

1. Den forunderlige L^AT_EXniske verden

Sidebemærkning 1.5. Bemærk at mange L^AT_EX-kapable teksteditorer oversætter teksten således at L^AT_EX ikke stopper ved fejl, men editoren vælger så bagefter at gøre opmærksom på at der var en fatal fejl i den og den linie. (L^AT_EX har forskellige modes den kan køre i, noget vi ikke behandler nærmere i denne bog.)

Det at kunne afkode en L^AT_EX-fejl kan godt være lidt af en kunst i sig selv, men det er noget de fleste kan lære. Desuden er de fleste fejl ofte selvindlysende. Om ikke andet er her en kort lille guide til hvordan man debugger et L^AT_EX-dokument, som ikke længere vil oversættes korrekt.

- (i) Kig på hvad det er L^AT_EX siger der er galt. Kan man ikke lige gennemskue betydningen af selve fejlmeddelelsen, kan man forsøge at slå den op i Appendix D på side 399, hvor man finder en lang liste (med forklaringer) over de mest almindelige fejlmeddelelser.
- (ii) Sammen med fejlmeddelelsen fortæller L^AT_EX desuden hvilken linie (sådan cirka) hvor fejlen opstod. Anvend dette til at kunne gå tilbage til kildekode for at lokalisere fejlen.

Her skal bemærkes at man bør holde øje med hvilken fil L^AT_EX for øjeblikket er i gang med at oversætte.¹⁶ Dette kan man se i outputtet (eller i .log-filen) som f.eks.

```
(./introduktion.tex [12
```

dvs. her har L^AT_EX åbnet filen »introduktion.tex« og er gået i gang med at skrive side 12 i outputtet. En fejl i »introduktion.tex« vil så fremstå som en fejl på linie 230 i »introduktion.tex«. Men man skal selv holde øje med hvilken fil der er tale om. (Dette er dog noget visse editorer kan hjælpe med.)

- (iii) Kan man slet ikke overskue hvad i al verden det er som er gået galt, kan man begynde at eksperimentere med at indsætte `\end{document}` på udvalgte steder i teksten. Vi husker jo at alt efter »`\end{document}`« bliver ignoreret, så hvis en ny-placeret »`\end{document}`« fjerner fejlen, så ved vi at fejlen kommer efter »`\end{document}`«-placeringen i kildekoden.

Har man gang i en inkluderet fil (se afsnit 11.1 på side 291) så kan man også anvende `\endinput` i stedet for `\end{document}`.

- (iv) Kan man slet ikke finde problemet, så kan man udkommentere de problematiske område og arbejde videre med sit dokument, og når man så har tid til det, kan man spørge en ven eller en lokal T_EX-guru. Har man ikke lige sådan en på hånden kan man spørge på DK-TUGs mailingliste (som også kan anvendes som nyhedsgruppe, se <http://www.tug.dk/maillingliste>), eller på nyhedsgrupperne `dk.edb.tekst` eller `comp.text.tex`.

1.8 Hvordan finder man hjælp?

1.8.1 Hvordan finder man manualen til pakken X?

Har man en moderne L^AT_EX-distribution (eller installation om man vil) så kan man for mange pakker findes deres manual via

¹⁶ Man kan sagtens opdele sit projekt i flere filer se afsnit 11.1 på side 291.

Når man skal spørge om hjælp

texdoc memman

i f.eks. et terminalvindue på Linux eller MAC. På Windows kan man anvende en kommandoprompt (dos-vindue), udføre det gennem kommandoen `Kør . . .` og på Vista/Win7 kan man anvende søge feltet i startmenuen.

Kommandoen `texdoc` virker klart bedst på T_EX Live-distributionen, her har man lavet et ret avanceret søgesystem således at hvis pakken ved navn *X* har kaldt sin manual for *manual*, så vil `texdoc X` fortsat finde manualen for *X*. Dette virker ikke så godt med en MikTeX-distribution.



Alternativt vil man kunne finde manualen til alle pakke på nettet via CTAN. Bare gå til <http://www.ctan.org/search.html#byName> og søg efter pakkenavnet. Her vil man oftest kunne finde en PDF-fil eller lignende.

Sidebemærkning 1.6. Husk at manualen man finder på CTAN, nemt kan være til en nyere version af pakken end den man selv har installeret på sit system. Hvilket er en god grund til at huske at opdatere sin L^AT_EX-installation.

1.8.2 Når man skal spørge om hjælp

Før eller siden kommer alle (!) L^AT_EX-brugere ud på så dybt vand at de får brug for hjælp fra andre. Heldigvis er L^AT_EX-brugere almindeligvis et *meget* hjælpsomt folkefærd, så er der mange steder man kan søge hjælp.

- Find ud af om der er en i din egen organisation som ved mere om L^AT_EX end du selv gør, henvend dig så til vedkommende (men lad være med at udnytte det til trivialiteter).
- På dansk er der flere fora som man kan henvende sig til
 - Dansk T_EX-brugergruppe har en mailingliste/nyhedsgruppe, som alle er velkomne til at tilmelde sig (man behøver ikke være medlem af foreningen, men af hensyn til SPAM skal man registrere sig for at kunne stille spørgsmål på listen). Se <http://www.tug.dk/maillingliste>.
 - Nyhedsgruppen `dk.edb.tekst` kan anvendes til L^AT_EXniske spørgsmål, der er dog ikke så meget aktivitet mere.
 - På <http://www.eksperten.dk> kan man finde et forum til L^AT_EX, se <http://www.eksperten.dk/spm/Programmer/Tekstbehandl/TeX/>.
 - StudiePortalen har også et L^AT_EX-forum som især kan være interessant for folk i gymnasiet, se <http://www.studieportalen.dk/Forums/Threads.aspx?id=26>.
- På udenlandsk er det vigtigste forum, nyhedsgruppen `comp.text.tex`. Det er ret godt sted at søge efter løsninger, husk man skal søge på Google Groups, resultater fra grupperne kommer ikke med i Googles almindeliges søgninger.
- Er man mere til fora så er <http://www.latex-community.org> et besøg værd.
- På StackExchange¹⁷ er der også folk som svarer på L^AT_EX-spørgsmål, <http://tex.stackexchange.com/>.

¹⁷ Googler man kan man også finde StackOverflow, det er vist samme site.

1. Den forunderlige L^AT_EXniske verden

- Det er også værd at tage et kig i <http://www.tex.ac.uk/cgi-bin/texfaq2html?introduction=yes>, der står rigtig mange gode løsninger, desværre er søgningen på siden ikke altid optimal.

For at andre skal kunne yde den bedste hjælp er det en god ide at lære at stille spørgsmål på den rigtige måde. Det er ret svært at hjælpe med spørgsmål så som

mit dokument er gået i ged – hjælp!!

Et godt stillet spørgsmål er kendetegnet ved at

- (i) man forklarer hvad det er man gerne vil opnå, gerne på en sådan måde at andre kan forstå stå det.
- (ii) har man prøvet at kode noget, men har fået en fejl, så begræns dokumentet (evt. under et nyt navn) til et minimalt eksempel (se afsnit 1.8.3) og inkluder/vedhæft dette sammen med dit spørgsmål. Dette gør det ikke bare nemmere at løse problemet, thi andre kan enten reproducere problemet på deres eget system, eller hurtigt se hvad problemet er, det gør det også mere sandsynligt at andre vil bruge tid på at hjælpe (det er hurtigere at komme i gang med at debugge problemet).

1.8.3 Minimalt eksempel

Når man stiller spørgsmål gennem/på diverse medier/fora så vil man hurtigt opdage at man bliver bedt om at stille med et *minimalt eksempel*.

Dette er faktisk er ret vigtigt fejlfindingsredskab, så det er et begreb de fleste vil have fordel af at lære. Et *minimalt eksempel* karakteriseret ved:

- (a) Et dokument som kan oversættes, dvs. det har `\documentclass{<klassenavn>}` samt `\begin{document}` og `\end{document}`
- (b) Preamblen (hvad dette er forklares i afsnit 1.6.3) skal være så kort som mulig mens man fortsat har samme problem som man ønsker hjælp til.
- (c) Selve dokumentet, dvs. det som står mellem `\begin{document}` og `\end{document}` skal også være så kort som muligt. Man har sjældent brug for at inkludere en helt 10-siders rapport for at få hjælp til en formel som ser underlig ud.

Den nemmeste måde at lavet et minimalt eksempel er ved at man gemmer sin problematiske fil under et nyt navn (kan være en ide at lave en ny folder og lægge de problematiske filer der mens man debugger, så kommer man ikke ved en fejl til at slette noget vigtigt) og så går i gang med at fjerne ting, mens man husker at oversætte dokumentet ret ofte for at se om problemet fortsat er til stede. Så finder man ud af hvilke pakker der er relevante og hvilken det af dokumentet som er irrelevant for problemet.

I mange tilfælde ender man op med selv at finde ud af hvad løsningen var, f.eks. at der var en pakke som gav problemer (giver så udslag i at problemet f.eks. forsvinder hvis man fjerner en bestemt pakke).

På nettet kan man finde andre introduktioner til begrebet *minimalt eksempel* <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=minxaml> og en lidt bedre forklaring findes via <http://www.minimalbeispiel.de/mini-en.html>.



I det følgende antages det, at man har en fungerende L^AT_EX-installation og at man ved hvordan ens valgte editor virker (primært hvilken knap eller tastatur-kombination man skal anvende for at køre `latex` eller `pdflatex`). I de følgende par eksempler vil vi dog først beskrive hvordan man kan oversætte et L^AT_EX-dokument, uden at kompilere direkte fra editoren, fordelen ved denne viden er naturligvis at skulle noget gå helt galt med editoren så kan man altid ty til en kommandolinieprompt for at oversætte sit dokument.

1.9 Dansk opsætning

Alt i alt bliver den gode opsætning for danskere, at man skal *altid* huske følgende fire linier

```
\usepackage[latin1]{inputenc} % eller med applemac eller utf8
\usepackage[danish]{babel}
\renewcommand{\danishhyphenmins}{22} % bedre orddeling
\usepackage[T1]{fontenc}
```

inputenc
babel
fontenc

Eksempel
1.8



Desuden skal man naturligvis huske »a4paper« som option til dokumentklassen (hvis det senere er meningen at det skal kunne printes).

Sidebemærkning 1.7. Linien

```
\renewcommand{\danishhyphenmins}{22}
```

Gør at der som minimum skal blive to tegn på linien ved orddeling og minimum flyttes to tegn ned på næste linie. Desværre er værdien anvendt af `babel` »12«¹⁸, hvilket kan give orddelingen »h-vor«.

Det kan være en fordel at flytte angivelsen af sproget op blandt options til dokumentklassen.

¹⁸ Dette er faktisk korrekt dansk, altså at der ved orddeling gerne må deles således at der tilbage før bindestrengen står kun et tegn. Dette ser dog ikke pænt ud, så vi anbefaler »22«.

Almindelig tekst

Dette kapitel handler kort om behandlingen af den almindelige tekst, tegnsætning, specialtegn, skift af fontvariant og lignende. Vi skal i et senere kapitel komme med nogle ekstra kommentarer som ligger lidt over niveauet i dette kapitel.

Kapitlet er introducerende vedr. behandlingen af tekst, og er den grundlæggende vide alle bør kende til. I Kapitel 9 på side 223 fortsætter vi med flere tekstrelaterede emner, som forudsætter et vist kendskab til L^AT_EX, bl.a. vedr. fonte.

2.1 Tekstopdeling

Når man skriver i de fleste teksteditorer og kommer til højre kant af editoren (eller hvad den nu er konfigureret til) så vil editoren ofte automatisk gå til en ny linie og skrive videre der. I de fleste tilfælde vil editoren bryde linien ved et mellemrum således at den tekst man har er nem at læse. Dette automatiske editorindsatte lineskift betyder *intet* for outputtet fra L^AT_EX (lige som i HTML).

Hvis man bare manuelt skifter linie i kildekoden sker der ikke noget i det resultat man får ud.

Hvis man bare manuelt skifter linie i kildekoden sker der ikke noget i det resultat man får ud.

Eksempel 2.1

Nyt afsnit

På dansk har begrebet »*afsnit*« tilsyneladende to betydninger. Vi har betydningen *nyt afsnit*, hvilket tilsyneladende defineres som en samling af sætninger i en samlet sammenhæng. Når man så starter på noget nyt starter man på et *nyt afsnit*.

Her starter vi så på et *nyt afsnit*, idet den anden betydning af *afsnit* anvendes ofte når man ønsker at henvise til navngivne steder i en tekst. Man kunne også kalde det for et *tekstafsnit* (det har jeg dog aldrig hørt nogen gøre i daglig tale). Vi er f.eks. i underafsnittet »*Nyt afsnit*« i det nummererede afsnit 2.1.

Et nyt afsnit i kildekoden, markeres i L^AT_EX ved at indsætte en eller flere blanke linier.

Noget tekst, bla bla
en blank linie som starter et nyt afsnit.

Noget tekst, bla bla
en blank linie som starter et nyt afsnit.

Eksempel 2.2

Starten på det næste afsnit i outputtet vil så automatisk blive indenteret.

Sidebemærkning 2.1. Det er en europæisk tradition, at man markerer et nyt afsnit ved indrykning og ingen ekstra afstand mellem afsnit. Det er desuden den opsætning de fleste typografiske bøger anbefaler.

Se desuden afsnit E.1 på side 407.

2. Almindelig tekst

Traditionen man ser i Word, med at der er afstand mellem afsnit, bør man holde sig fra. Det tjener ikke noget formål i forbindelse med forståelsen af teksten og kan i mange tilfælde endda virke forvirrende. Det eneste sted, hvor det er umiddelbart normalt at have afstand mellem afsnit, er i breve (her sørger de forskellige brevklasser (f.eks. *letter* eller bedre *scr/ttr2*) automatisk for at denne konfiguration kommer med). Afstand mellem afsnit kan naturligvis også anvendes i meget specialiserede publikationer, såsom madopskrifter og lyrik. Til f.eks. bachelorprojekter og specialer giver det ingen mening, specielt ikke hvis dokumentet indeholder fremhævet matematik.

Sidebemærkning 2.2. Man kan sagtens anvende en blank linie eller lignende til at markere starten på noget nyt. Man skal bare ikke indsætte luft mellem alle afsnit i teksten.

I denne bog anvendes en lidt speciel konstruktion som *memoir*-klassen (som vi kraftigt anbefaler) stiller til rådighed. Den er cirka lig med

Eksempel 2.3

```
\fancybreak{${*\quad*\quad*}$} % kræver memoir klassen
```

Makroen vil enten efterlade en blank linie eller indsætte den angivne tekst midt på linien og afsnittet efter vil ikke blive indenteret. Et eksempel kan ses på side [xix](#).

Bemærk at koden til `\fancybreak` er ret nemt at kopiere fra kildekoden til *memoir*-klassen.

Afsnitsindrykning

`\noindent`

Som nævnt indrykkes nyt afsnit automatisk. Skulle det ske at dette ikke lige var det som var meningen kan man starte linien med `\noindent`. Det *er* muligt helt at fjerne denne indentering, men det anbefales kun til breve (i de specielle brevklasser er indenteringen automatisk sat til nul) eller f.eks. lyrik.

Anvendelsen af `\noindent` vil oftest komme på tale i forbindelse med tekst efter environments som af en eller anden grund gerne vil have at teksten efter environmentet starter et nyt afsnit (hvilket ofte ikke er nødvendigt).

Sidebemærkning 2.3. Jeg havde på et tidspunkt en bruger som klagede over, at i vedkommendes dokument var stort set hver eneste sætning indenteret. Det viste sig at være fordi denne skribent, af overskuelighedsmæssige årsager, havde indsat en masse blanke linier rundt omkring i teksten.

Moralen er her, hvis man ønsker at indsætte blanke linier i sin kode for at gøre den mere overskuelig, så skal man huske at starte disse 'blanke' linier med et `%`-tegn, så ignoreres linien af L^AT_EX og forstyrrer ikke noget.

Tvungen sideskift

`\newpage`

Man kan fremtvinge en ny side via den åbenlyse makro »`\newpage`«. Der er dog visse naturlige steder hvor denne makro ikke virker, eksempelvis midt i noget fremhævet matematik, hvor man så i stedet kan anvende andre teknikker, se eksempelvis afsnit [4.2.6 på side 98](#).

`\clearpage`
`\cleardoublepage`

Der er desuden to ekstra makroer som virker lidt på samme måde som `\newpage`, nemlig `\clearpage` og `\cleardoublepage`. De har begge også indflydelse på placeringen af såkaldte flydende figurer (*floats*, se afsnit [6.2 på side 142](#)). Alt efter hvilken

Overskriftsopdeling af teksten

dokumentklasse man anvender og hvordan dokumentet ellers er konfigureret vil `\cleardoublepage` desuden sørge for at teksten efter, først starter på den næste højreside. Konstruktionen som anvendes til at starte et nyt kapitel, anvender internt `\cleardoublepage`.

Overskriftsopdeling af teksten

Større samlinger af afsnit, giver man ofte en overskrift som i mange tilfælde vil være nummereret.

En af fordelene ved \LaTeX , som måske ikke bliver nævnt så tit er at \LaTeX lægger op til en opdeling mellem tekst og layout. Lidt på samme måde som HTML med CSS. Som skribent skal man bare sørge for at skrive teksten, man skal ikke rode med layoutmæssige aspekter undervejs. Dette betyder naturligvis også at man heller ikke skal begynde at konstruere sine egne afsnitoverskrifter, den slags har man makroer til at tage sig af.

Overskrifter i \LaTeX skrives via en af makroerne som kan ses i Tabel 2.1.

Tabel 2.1: Makroer til at lave afsnitoverskrifter

<code>\part{\langle tekst \rangle}</code>	<code>\subsection{\langle tekst \rangle}</code>
<code>\chapter{\langle tekst \rangle}</code> ^a	<code>\subsubsection{\langle tekst \rangle}</code>
<code>\section{\langle tekst \rangle}</code>	<code>\paragraph{\langle tekst \rangle}</code> ^b
	<code>\subparagraph{\langle tekst \rangle}</code> ^b

^a Findes ikke i *article*-klassen og lignende.

^b Anvendes sjældent, så den kan man til tider anvende til sjove ting.

I Eksempel 2.4 finder man et simpelt eksempel på brugen af dette, lavet via *article*-klassen.

<pre>\section{En sektion} tekst \subsection{En undersektion} \subsection{En undersektion mere} tekst \subsubsection{En underundersektion} \section{En ny sektion} \subsection{En ny undersektion} \paragraph{en paragraf overskrift} tekst</pre>	<pre>1 En sektion tekst 1.1 En undersektion 1.2 En undersektion mere tekst 1.2.1 En underundersektion 2 En ny sektion 2.1 En ny undersektion en paragraf overskrift tekst</pre>
---	--

Eksempel 2.4

Man observerer at kommandoerne `\section`–`\subsubsection` automatisk nummereres og at de underliggende numre arver modernummeret. Dog vil `\paragraph` normalt ikke blive nummereret, men det er blot et konfigurationsspørgsmål.

2. Almindelig tekst

Man kan desuden meget nemt lave bilag/appendices, se afsnit 9.1.1 på side 223 hvor vi også vil sige lidt mere om afsnit `\section` og vende og deres samspil med indholdsfortegnelsen.

Afsnitoverskrifter uden nummer

Alle overskriftsmakroerne fra tabel 2.1 på foregående side har desuden en variant (kaldet stjernet) som ikke skriver noget nummer. Man får disse ved at indskyde et »*« mellem afsnitmakroen og argument som f.eks. `\section*{titel}`, se Eksempel 2.5.

Eksempel 2.5

```
\section*{En unummereret sektion}

tekst
\subsection*{Og en undersektion
uden nummer}
```

En unummereret sektion

tekst

Og en undersektion uden nummer

Sidebemærkning 2.4. Det er en god ide at skrive et par blanke linier før og efter en overskriftsmakro. Det har ingen betydning for hvordan overskriften bliver skrevet, men gør koden langt mere overskuelig.

Fordelen ved at anvende L^AT_EXs indbyggede afsnitmakroer er at man så automatisk får adgang til et væld af ekstra muligheder. Blandt andet kan afsnitstitlerne anvendes i en sidehoved, kapitler vil normalt automatisk starte på en ny højreside og L^AT_EX kan automatisk lave en indholdsfortegnelse, se afsnit 2.6 på side 36.

Bemærk for øvrigt at et afsnit efter en overskrift *ikke* indenteres. Det er der ingen grund til. Indenteringen markerer jo at man går i gang med noget nyt, og det har man allerede gjort opmærksom på via overskriften. (Skulle man have en pedantisk vejleder/universitet, så tag et kig på pakken `indentfirst`.)

2.2 Tegnsætning

Indenfor L^AT_EX skriver man bare sin almindelige tekst som man normalt ville gøre. Sæt punktum, komma, kolon og semikolon som man plejer. Men for resten af tegnsætningen gælder der helt andre regler som man nok ikke er stødt på i ens brug af mere gængse tekstbehandlingsprogrammer.

Orddeling

babel
fontenc

En korrekt installeret¹ L^AT_EX installation vil selv være i stand til at orddele langt de fleste ord, især i samarbejde med `babel` og `fontenc`. Der vil altid være visse ord som L^AT_EX ikke selv kan finde ud af, eller visse ord som man slet ikke vil have ord delt.

Når L^AT_EX ikke selv kan finde ud af at orddele et bestemt ord vil man ofte se, at ordet stikker ud i højre margin. Man kan så hjælpe L^AT_EX ved at indsætte »\-« mellem de stavelser hvor man vil tillade L^AT_EX at orddele ordet. Det er så op til L^AT_EX at finde det bedste sted at lave en eventuel orddeling. Her er et (dårligt) eksempel.

¹ MiK_TE_X-brugere skal huske at tilføje dansk som sprog i »MiK_TE_X Options« efter installationen. For andre installationer skal man konsultere dokumentationen eller huske at tage højde for det ved installationen.

Anførelstegn

```
Noget tekst, text og mere tekst og  
overs\k\r\i\f\t
```

```
Noget tekst, text og mere tekst og oversk-  
rift
```

Eksempel
2.6

Eksempel 2.6 viser, at når man anvender »\-« godtages alle disse punkter som de-le punkter mens de delingspunkter som orddelingsmønstrene normalt ville vælge ignoreres.²

Har man nogle ord som L^AT_EX bare bliver ved med ikke at kunne orddede, kan man i stedet fortælle globalt hvordan man gerne vil have ordet orddelt. Dette sker med makroen `\hyphenation{<liste af ord separeret af mellemrum>}`, man markerer stavelserne ved at indsætte »-«, altså *ikke* med »\-«!

\hyphenation

```
\hyphenation{ord-de-lings-al-go-rit-me_FORTRAN}
```

Eksempel
2.7

Man kan også på denne måde bestemme, at man f.eks. kun måtte orddede *orddelingsalgoritme* mellem *orddelings* og *algoritme* ved kun at indsætte en »-« mellem de to ord. Eksempel 2.7 betyder desuden, at ordet FORTRAN og varianter af denne *ikke* må orddedes.

`\hyphenation` placeres i preambelen og man må gerne lave flere af dem, alle ordene bliver bare tilføjet til en intern liste. Hvis et ord optræder uden »-« vil dette ord slet ikke blive orddelt.

Ønsker man et enkelt sted at undgå et ord bliver orddelt kan man eksempelvis anvende `\mbox{<tekst>}`.

\mbox

I afsnit 9.3 på side 226 kigger vi lidt nærmere på orddeling og hvordan man kan have orddeling på forskellige sprog. Her ser vi også på problemet vedrørende ordsammensat med bindestreg. Se også afsnittet *Streger og orddeling* på side 27.

Anførelstegn

Indenfor L^AT_EX og (stort set) resten af typografiens verden anvender man *ikke* tegnet »"« til at angive anførelstegn. Bemærk, at start- og slut-anførelstegnet i de fleste bøger ikke er det samme, i denne font er det „...“ eller “...”, dvs. man skal kunne se hvilket anførelstegn fungerer som starttegn og hvilken anvendes som sluttegn – derfor dur det ikke med det samme symbol som start- og sluttegn. Tegnet »"« (fra tastaturet) benyttes derfor (f.eks. af babel) til andre ting. I stedet anvendes tre (eller fire) forskellige former for anførelstegnsmarkeringer, se Tabel 2.2 på den følgende side.

babel

Sidebemærkning 2.5. Der findes også en pakke (`csquotes`) som helt kan overtage måden hvorpå anførelstegn skal skrives. Pakken understøtter bl.a. forskellige sprog, således at har man et citat på engelsk da vil anførelstegn automatisk blive skrevet på den traditionelle engelske vis i stedet på den normale danske. Nærmere behandling af pakken ligger udenfor rammerne af denne bog, interesserede henvises i stedet til [Lehman \(2006\)](#).

csquotes

```
\usepackage[danish=quotes]{csquotes}
```

— slut preamble —

```
\enquote{Hvad var det \enquote{han}  
sagde?}
```

„Hvad var det 'han' sagde?“

Eksempel
2.8



² Det er muligt at omdefinere \- således at den tilføjer ombrydningspunkter i stedet for at diktere dem. Det er dog ikke noget vi kommer videre ind på her.

2. Almindelig tekst

Tabel 2.2: Anførselstegn i \LaTeX

Kode	Output	Beskrivelse
' '	'tekst'	Enkelt anførselstegn. ^a
" "	"tekst"	Engelske dobbeltanførselstegn.
"' "'	„tekst“	Danske dobbeltanførselstegn. Bemærk brugen af »"«. ^b
>> <<	»tekst«	Kaldes guillemets. Bemærk, at det er fontenc-pakken der giver denne genvej. ^c

^a Bemærk, at start- og sluttegn er to forskellige tegn. Den første er accent-grave (eller back-ping) (specialtegn fra tastaturet, på et dansk tastatur er det det øverste tegn på tasten med de to stregaccenter se figur 1.1 på side 9, på et amerikansk tastatur findes det under ESC-tasten). Sluttegnet er apostrof (findes lige ved siden af Enter/Return-tasten).

^b Jeg må ærligt indrømme, at jeg ikke ved om dette er den type anførselstegn som anbefales i dagens DK, men nuvel det var dem jeg lærte i skolen. Jeg anvender dog normalt de engelske.

^c Bemærk at det er babel som gør at vi bare kan skrive to større-end og to mindre-end, man kan også benytte `\guillemotright` (») og `\guillemotleft` («). Grunden til, at vi benytter `-right`-varianten som starttegn er at «» er den franske notation. Mens rækkefølgen er omvendt på dansk (også kaldet den tyske fortolkning). Man kan også få ◊.

Om streger (bindestreg, tankestreg, og venner)

De fleste tror nok, at der kun findes en form for streg, nemlig »-« som ligger på tastaturet. Men indenfor typografiens verden er der faktisk (mindst) fire, med hver deres funktion. Heldigvis har \LaTeX gjort dem nemme, at gå til, de er nemlig alle bare forskellige antal »-« skrevet efter hinanden. \LaTeX laver dem så om til det man kalder ligaturer.

Tabel 2.3: Typografiske streger i \LaTeX .

Kode	Output	beskrivelse
a-b	a-b	Bindestreg.
a - b	a–b	En-dash. Se forklaring i teksten.
a - - b	a—b	Em-dash.
$\$a-b\$, \$-c\$\$	$a - b, -c$	Matematisk minus.

Bindestregen benyttes som navnet antyder som bindestreg, men *kun* som bindestreg! En en-dash anvendes ofte til at indikere et interval³ man skriver altså 12–19 (12 - 19) og *ikke* 12-19 (12-19). På dansk benytter man også en en-dash som tankestreg med et mellemrum i begge ender, altså _ - _ . En en-dash er lig en halv em-dash.

Em-dash benyttes stort set udelukkende som engelsk tankestreg, hvor man så *ikke* anvender mellemrum i nogen af enderne. En almindelig praksis er dog at anvende den (centreret) i tabeller for at markere en tom celle, se Kapitel 7 på side 165.

Sidebemærkning 2.6. Jeg er blevet gjort opmærksom på, at man i den danske retskrivningsordbog stort set kun anvender bindestreg til at angive intervaller med, dvs. 9-15 i stedet for 9–15. Jeg holder mig selv til den engelske anbefaling som også er den som er beskrevet her.⁴

³ Dette er en meget almindelig fejl som jeg retter alt for ofte, så husk det nu.

Streger og orddeling

Sidebemærkning 2.7. Bemærk desuden at udseendet af bindestreg og tankestreger afhænger ofte af den anvendte font. Der er visse fonte hvori bindestregen har en hældning i stedet for vandret som man oftest ser.

Streger og orddeling

Man skal bemærke, at L^AT_EX gerne må linieombryde ved bindestreg (eller ved en intervalbindestreg). Hvis det er et problem (f.eks. ved ordet »*p*-adisk«) kan man benytte `\nobreakodash` som stammer fra pakken `amsmath`.⁵ Man sætter så `\nobreakdash` foran den streg ved hvilken man gerne vil undertrykke linieombrydningen. Eksempelvis

`\nobreakodash`

```
$p$\nobreakdash-adisk
```

**Eksempel
2.9**

I den modsatte grøft anvender vi jo på dansk masser af sammensatte ord både med og uden bindestreg. Givet et sammensat ord af typen før-efter, da vil L^AT_EX *kun* linieombryde ordet *ved bindestregen*! Dvs. man kan nemt komme ud for, at et sammensat ord fortsætter ud i marginen fordi L^AT_EX ikke orddeler ordet. Løsningen på dette problem er ved brug af dansk `babel` at anvende »" - « i stedet for »-«, eller manuelt at indsætte nogle »\ -«-er.

```
(forkert) tekst tekst tekst tekst  
tekst retskrivnings-regelrytter
```

```
(rigtig) tekst tekst tekst tekst  
tekst retskrivnings"-regelrytter
```

```
(forkert) tekst tekst tekst tekst tekst  
retskrivnings-regelrytter
```

```
(rigtig) tekst tekst tekst tekst tekst ret-  
skrivnings-regelrytter
```

**Eksempel
2.10**

På engelsk er det ikke så tit man anvender bindestreger i ord, så derfor finder man *ikke* »" - « i den engelske `babel`. Men den er nem at tilføje

```
\usepackage[english]{babel}  
\makeatletter  
\usesorthands{"}  
\defineshorthand{"-}{\nbreak-\bbl@allowhyphens}  
\makeatother
```

**Eksempel
2.11**



Ligaturer

I mange fonte ser man at bogstavkombinationerne `ff`, `fi`, `fl`, `ffi` og `ffl` bygges sammen til en kompakt enhed kaldet en *ligatur*. Dette gør L^AT_EX automatisk i det omfang at fonten er sat op til det. Alt efter font kan der være mange forskellige af disse. Et ret voldsomt eksempel kan ses ved fonten `zapfino`: <http://www.tug.org/mactex/src/Demos/XeTeX-showcase/story-zapfino.pdf> eller http://www.tug.org/mactex/src/Demos/texshowcase/peace_on_earth.pdf.

I visse sammenhænge, specielt vedrørende sammensatte ord, er det ikke hensigtsmæssig at der fremkommer en ligatur på et bestemt sted. Dette kan man så fjerne ved indsættelse af `{}` på det sted man ikke ønsker ligatur. Her er et par eksempler (vi har skiftet til fonten `Latin Modern` da den har tydeligere ligaturer).

⁴ Man kunne forledes til at tro at grunden til at man ikke følger den engelsksprogede tradition på dette punkt kunne være, at de fleste slet ikke ved at der er to forskellige streger.

⁵ Den anvender man jo alligevel altid...

2. Almindelig tekst

Eksempel 2.12



<pre>\usepackage{lmodern} ----- slut preamble ----- stof{}fri giraf{}føl telegraf{}ingeniør fi fl ffi ffl</pre>	stoffri girafføl telegrafingeniør fi fl ffi ffl
---	---

Tvungen mellemrum uden linieombrydning

Med *tvungen mellemrum* menes her et mellemrum hvor det ikke må være lineskift, på engelsk kaldes dette »*non breaking space*«. Dette opnår man i L^AT_EX via tegnet »~« (tilde), man sætter det simpelthen i stedet for mellemrummet, L^AT_EX må så finde et andet sted at dele linien. I L^AT_EX sprog kaldes en *tilde* i denne forbindelse ofte for en »*tie*« efter »*tie something together*«.

De fleste burde faktisk anvende tilde noget mere. Huskereglen er:

*Det vil se dumt ud hvis der kommer
linieombrydning ved dette mellemrum*

Nogle klassiske eksempler følger herunder:

Hr. ~Hansen	af dimension~\$1\$.
Sætning~\ref{thm:5}	side~4 og~5

Det er egentlig noget man bør overveje mens man skriver, thi så undgår man at skulle rette en masse linier når man evt. senere ændrer marginer etc. En tie er også løsningen som bør anvendes når man til slut gennemgår alle venstre- og højrekanter af teksten for at tjekke om linieombrydningerne er i orden.

Dette er en af de mest almindelige fejl jeg ser når jeg tilretter preprints, thi en af de ting jeg ændrer er marginerne hvorefter alle linieombrydningerne skifter og man får en masse henvisning som er blevet brudt, med mindre man naturligvis har sørget for at anvende en tie.

Engelsk punktum og mellemrum

Hvis man af en eller anden grund *ikke* anvender dansk babel, f.eks. skriver på *engelsk*, skal man være opmærksom på en lille feature ved kombinationen af *små bogstaver* og *punktum*. På engelsk er der en tradition for, at der er større mellemrum mellem sætninger end mellem ord. Dette sørger L^AT_EX automatisk for, således, at efter f.eks. kombinationen »*lille bogstav*«. kommer der et dobbelt mellemrum. Dette gør man ikke i de fleste kontinental-europæiske sprog, så her sørger babel at slå denne feature fra (via `\frenchspacing`).

Men ved engelsk skal man altså være opmærksom på dette og derfor passe på med forkortelser etc. (Vi har skiftet fonten til Computer Modern da effekten her er tydeligere.)

Eksempel 2.13

<pre>\nonfrenchspacing e.g. tekst \frenchspacing e.g. tekst</pre>	e.g. tekst e.g. tekst
---	--------------------------

Det man selvfølgelig gør i denne situation er at skrive dette forkortelsespunktum som »._«. Og ønsker man helt at slå featuren fra smider man bare en `\frenchspacing` i preambelen.

Ellipsis

Ellipsis ... (eller bare hvordan man skriver prik prik prik)

Typografisk set kaldes »...« *ellipsis*, på dansk er det bare »prik prik prik«. Men hvad mange ikke lægger mærke til er, at dette *ikke* er det samme som at sætte tre punktummer »...« (eller med mellemrum ». . .«). I den første er ».«-erne for tæt sammen, mens man ved den anden risikerer at få lineskift mellem ».«-erne (desuden er der for stor afstand). I stedet anvendes makroen `\dots`, som også kan anvendes i matematik, se afsnit 3.2.6 på side 61.

`\dots`

En, to, `\dots` og endnu mere.

En, to, ... og endnu mere.

Eksempel 2.14

I teksten skal man lige huske f.eks. at anvende `_` efter `\dots`, thi den jo ellers æder et mellemrum. Husk desuden at hvis *prik prik prik* afslutter en sætning, så behøver man ikke sætte punktum efter, det er unødvendigt.

Det skal nævnes at hvis man er meget pernittegrøn, vil man opdage at spacingen ved `\dots` er en smule forkert når den anvendes i teksten. Hvis man er en af disse, kan det være en ide at tjekke pakken `ellipsis`.

`ellipsis`

Specialbogstaver og accenter

Kombinationen af pakkerne `inputenc` og `fontenc` gør det muligt for \LaTeX at forstå tegn som \acute{e} eller \csc skrevet direkte fra tastaturet. Men det er jo ikke alle som har adgang til f.eks. \T , \E , \G eller \n fra tastaturet. Dette løser \LaTeX ved at give genveje til disse tegn, se tabel 2.4 på den følgende side. Desuden er `\aa` og `\AA` også to (af danskere) meget anvendte genveje (til `\r{a}` og `\r{A}`, i.e. \aa og \AA). På denne måde kan man skrive dansk selv hvis man ikke har adgang til et dansk tastatur.

`inputenc`
`fontenc`

`\aa` (\aa)
`\AA` (\AA)

Sidebemærkning 2.8. Husk, at man i matematik har helt andre konstruktioner til at lave matematiske accenter. Konstruktionerne beskrevet her kan/bør *ikke* anvendes i matematik.

Bare lige for god ordens skyld gentager vi lige de danske bogstaver her

danske bogstaver

`\ae`, `\o`, `\r{a}`, `\aa`, `\AE`, `\O`, `\r{A}` og `\AA`.

\ae , \o , \aa , \AA , \AE , \O , \AA og \AA .

Eksempel 2.15

Husk dog, at man med pakken `inputenc`, med den rigtige option, gerne må skrive \ae , \o eller \aa (eller \o , $\text{\acute{e}}$ eller \csc) direkte fra tastaturet.

`inputenc`

Tekstsymboler

I de fleste fonte finder man ud over bogstaver og specialtegn desuden visse tekstsymboler, så som f.eks. \copyright . Visse tekstsymboler kan man gå til direkte i \LaTeX men for de flestes vedkommende skal man bruge pakken `textcomp`.

`textcomp`

Man skal naturligvis være opmærksom på det sagtens kan ske at visse symboler ikke findes i visse fonte, det kan derfor være en ide at give optionen `safe` til pakken, så vil den kun stille de tekstsymboler til rådighed som der med nogenlunde sikkerhed er i de fleste fonte.

I Tabel 2.5 på næste side vil vi bare give et par eksempler på hvilke symboler man vil kunne finde. For yderligere symboler se listerne i [Pakin \(2003\)](#) eller se Tabel 7.6 i [Mittelbach og Goossens \(2004\)](#). Som man desuden vil kunne se i [Pakin \(2003\)](#) så er der desuden en masse andre pakker der kan give adgang til forskellige tekstsymboler. Bemærk at udseendet at symbolerne afhænger af den valgte tekstfont.

2. Almindelig tekst

Tabel 2.4: Accenter og specialbogstaver i L^AT_EX

Kode	Eksempel	Effekt	Navn	Kode	Effekt
\'	\'{e}	é	accent aigu ^a	\AE	Æ
\`	\`{a}	à	accent grave ^a	\DH	Ð
\"	\"{o}	ö	umlaut ^b	\DJ	Đ
\.	\.{Z}	Ž	accent prik over ^c	\L	Ł
\^	\^{o}	ô	accent circumflex	\NG	Ń
\~	\~{n}	ñ	accent tilde	\O	Ø
\=	\={y}	ȳ	accent macron ^c	\OE	Œ
\b	\b{a}	ā	accent understreg ^c	\TH	Þ
\c	\c{c}	ç	accent cedille	\ae	æ
\k	\k{E}	Ę	accent ogonek	\dh	ð
\r	\r{u}	ŕ	accent bolle	\dj	đ
\t	\t{oo}	ōō	bind efter ^c	\l	ł
\u	\u{G}	Ĝ	accent breve	\ng	ŋ
\v	\v{D}	Ǿ	accent háček	\o	ø
\H	\H{O}	Ő	ungarsk umlaut	\oe	œ
\d	\d{e}	ẹ	accent prik under ^c	\th	þ
				\ss	ß

^a <http://da.wikipedia.org/wiki/Accenttegn>

^b På dansk kaldes denne *trema*, dog skulle den fonetiske virkning være anderledes, se <http://da.wikipedia.org/wiki/Trema>.

^c Hvis nogen kender et godt dansk navn for denne må man gerne kontakte mig.

Tabel 2.5: Nogle tekstsymboler fra *textcomp*-pakken

Kode	Resultat	Kode	Resultat
\copyright	©	\textregistered	®
\textperthousand	‰	\textsterling	£
\texttrademark	™	\texteuro	€
\textdagger	†	\textparagraph	¶ ^b
\textsection	§ ^a	\textcelsius	°C ^c
\textdegree	° ^c	\textmu	μ ^c

^a Alias \P

^b Alias \S

^c Undlad at anvende disse til at angive enheder. Her er pakken *siunitx* meget bedre, se afsnit 10.2.1 på side 263.

2.3 Om fontvalg og -størrelser

Fontstørrelser

Som allerede nævnt tidligere er det normalt dokumentklassen samt dens indbyggede options som styrer den overordnede fontstørrelse. Men lokalt kan man sagtens ændre den. Der er dog en meget vigtig lektion at lære her. Makroerne som beskrives i det efterfølgende er såkaldte *deklarationsmakroer*, dvs. de tager *ikke* noget argument, men ændrer i stedet tingenes tilstand herfra og frem, man skal derfor passe lidt på.

Skift af fontvariant

Makroerne er beskrevet i tabel 2.6.

```
F.eks. \ vil teksten også være i \verb+\tiny+
efter
\verb+\tiny{tekst}+ -- se selv \tiny{tekst}
mere tekst
```

```
F.eks. vil teksten også være i \tiny efter
\tiny{tekst} – se selv tekst mere tekst
```

**Eksempel
2.16**

Hvis det er til en forside eller noget lignende kan man benytte syntaksen `{\makro ...}`, hvor så `{ }` beskytter det efterfølgende, se evt. afsnit A.2. Det skal dog indskydes, at man også kan få andre størrelser, men det skal vi ikke komme nærmere ind på her, se i stedet afsnit 9.6.4 på side 237. De faktiske størrelser af de kommandoer som er opstillet i tabel 2.6 afhænger af den valgte dokumentklasse samt naturligvis den valgte dokument fontstørrelse (gennem 10pt, 11pt eller 12pt klasseoption).

Tabel 2.6: Fontstørrelser i \LaTeX

Kode	Effekt	Kode	Effekt
<code>\tiny</code>	tekst	<code>\scriptsize</code>	tekst
<code>\footnotesize</code>	tekst	<code>\small</code>	tekst
<code>\normalsize</code>	tekst	<code>\large</code>	tekst
<code>\Large</code>	tekst	<code>\LARGE</code>	tekst
<code>\huge</code>	tekst	<code>\Huge</code>	tekst

Skift af fontvariant

Under normale omstændigheder har man i sit dokument adgang til forskellige fontvarianter. Man kan selvfølgelig også skifte selve fonten, hvilket vi taler lidt om i afsnit 9.6.1 på side 231.

\LaTeX deler fontbehandlingen op i tre dele, en *serie*, en *form* (engelsk: *shape*) og en *familie*. Disse deklarationer kan så blandes til f.eks. f.eks. at give en fed kursiv tekst, se tabel 2.7 på næste side. Man skal dog være opmærksom på at det afhænger af fontvalget om visse kombinationer faktisk eksisterer. Eksempelvis er det de færreste sans serif fonte som faktisk understøtter small caps. Så vil `\textbf{\textit{tekst}}` (*tekst*) så give fed kursiv tekst. Man skal dog vide, at det kommer helt an på fontopsætningen hvilke sammensætninger som virker. F.eks. er det sjældent, der findes en kursiv (eller fed) small caps (disse kan dog i visse sammenhænge blive simuleret i stedet).

Forskellen mellem den *slantede* font og den *kursive* er, at den slantede er konstrueret ud fra den normale tekstfont (tiltet en smule) mens den kursive normalt er en designet font. Forskellen mellem de to kan ses via `2 (\textsl{2})` versus `2 (\textit{2})` ved brug af Computer Modern fonten.

De alternative variantskiftende makroer i Tabel 2.7 på den følgende side er igen deklaramakroer og tager altså *ikke* noget argument. De korte `\text...` kommandoer kan kun tage korte tekster, svarende til et enkelt afsnit (dvs. ingen blanke linier). Har man brug for mere skal man anvende den alternative metode. Bemærk at der i de alternative navne indgår postfix *family*, *series* og *shape*. Alt efter den anvendte skrifttype kan disse frit blandes med en fra hver type.

2. Almindelig tekst

Tabel 2.7: \LaTeX s metoder til skift af skrifttype variant.

Tekstkommando	Alternativ	Effekt
<code>\textrm{...}</code>	<code>{\rmfamily ...}</code>	romersk (antikva) ^a
<code>\textsf{...}</code>	<code>{\sffamily ...}</code>	sans serif
<code>\texttt{...}</code>	<code>{\ttfamily ...}</code>	monospace
<code>\textmd{...}</code>	<code>{\mdseries ...}</code>	medium ^b
<code>\textbf{...}</code>	<code>{\bfseries ...}</code>	fed
<code>\textit{...}</code>	<code>{\itshape ...}</code>	<i>kursiv</i>
<code>\textup{...}</code>	<code>{\upshape ...}</code>	opret ^c
<code>\textsl{...}</code>	<code>{\slshape ...}</code>	<i>slantet</i>
<code>\textsc{...}</code>	<code>{\scshape ...}</code>	SMALL CAPS
<code>\textnormal{...}</code>	<code>{\normalfont ...}</code>	normal dokument font ^d
<code>\emph{...}</code>	<code>{\em ...}</code>	<i>fremhævet, se side 33</i>

^a Eller *opretstående*, dette er (oftest) standarden.

^b Nogle gange kendt som ikke-fed. Forskellen mellem denne og brug af f.eks. `\textnormal` er at `\textmd` kan i et område med fed kursiv skrift, skifte tilbage til ikke-fed kursiv skrift. Brug af `\textnormal` ville fjerne det kursive element. Medium er (oftest) standarden.

^c Dette er det modsatte af kursiv, i.e. ophæver kursiv, hvis teksten allerede er kursiv.

^d Denne justerer tilbage til dokumentets standard fontopsætning, bog uden at røre fontstørrelsen.



Advarsel 2.9 (\TeX kommandoer bør ikke anvendes).

I visse sammenhænge ser man også makroerne `\it`, `\bf`, `\sc` etc. Disse bør man *holde sig fra* og i stedet holde sig til `\text...` varianterne. Disse makroer stammer fra \TeX og deres brug kan være forvirrende for menigmand. (Prøv at sammenligne `{\bf\it tekst}` med `{\it\bf tekst}`, det er nok de færreste som havde forventet dette. Der er dog ingen problemer med at mikse de lange varianter `\itshape`, `\bfseries`, `\scshape` etc.)

Sidebemærkning 2.10. Hvis man i tabel 2.7 tager de alternative navne og fjerner »\«, så vil disse navne faktisk kunne anvendes som environments!

Eksempel 2.17

```
noget tekst\begin{bfseries}
  mere tekst
\end{bfseries} senere tekst
```

```
noget tekst mere tekst senere tekst
```

Fontændringen sker så naturligvis kun indenfor dette environment.

Fremhævning af tekst

Tip 2.11 (Hvor sker ændringer).

Fra tid til anden kommer jeg forbi konstruktioner a la

```
\begin{theorem}
{\upshape
... lang tekst ...
}
\end{theorem}
```

hvor tanken (fra skribenten) er, at man jo skal huske at beskytte verden udenfor, når man anvender de alternative fontmakroer. Men {}-parret er faktisk overflødige i dette tilfælde. Er man mellem en `\begin{...}` og tilsvarende `\end{...}` så gælder fontændringen *kun* frem til `\end{...}`. Det samme gælder faktisk de fleste ændringer af makroer, længder etc., se det følgende eksempel.

```
tekst
\begin{center}
  \bfseries fed tekst
\end{center}
mere tekst
```

```
tekst
                                fed tekst
mere tekst
```

Eksempel
2.18

Desuden er det lidt noget pjat at anvendes en konstruktion som ovenstående sammen med sætningskonstruktioner, man kunne lige så godt lære at konfigurere sætningerne, se Kapitel 5.

Fremhævning af tekst

Til fremhævning af noget tekst anvendes altså makroen `\emph` der normalt sætter teksten i kursiv. Men i modsætning til `\textit` holder `\emph` øje med den omkringliggende tekst og sørger for at ændre opførsel derefter.

`\emph`

```
noget almindeligt tekst \emph{fremhævet}
og mere tekst. Så kommer der noget i
\textit{kursiv og her \emph{fremhæver}
vi igen}.
```

```
noget almindeligt tekst fremhævet og mere
tekst. Så kommer der noget i kursiv og her
fremhæver vi igen.
```

Eksempel
2.19

Som man kan se bliver fontvalget i det første eksempel til kursiv, mens det i det andet bliver den normale *upright* font. Denne forskel kan være lidt svær at se. Med pakken `fixltx2e`⁶ får man makroen `\emphshapen` som kan ændres til et andet valg.

`\emphshapen`

```
\usepackage{fixltx2e}
\renewcommand*\emphshapen{\scshape}
```

```
noget almindeligt tekst
\emph{fremhævet} og mere tekst.
Så kommer der noget i \textit{kursiv
og her \emph{fremhæver} vi igen}.
```

```
noget almindeligt tekst fremhævet og mere tekst.
Så kommer der noget i kursiv og her
FREMHÆVER vi igen.
```

Eksempel
2.20



Personligt vælger jeg normalt at `\emphshapen` skal være fed kursiv.

⁶ Som man anbefales at anvende under alle omstændigheder. Er man kun interesseret i `\emphshapen` så er denne makro også inkluderet i *memoir*-klassen.

2. Almindelig tekst

Sidebemærkning 2.12. Bemærk at man i den skrevne tekst *aldrig* understreger tekst for at fremhæve den, ej heller til at markere at dette er en overskrift. Dette anses for at være dårlig typografi og er noget man gør på en tavle eller ved anden form for håndskrift, men ikke noget som anvendes på tryk. Anvend i stedet f.eks. `\emph`.

2.4 Krydsreferencer

Alt hvad L^AT_EX selv nummererer kan man senere henvise til og få både nummeret samt eventuelt et side tal. Man kan således krydsreferere til alt lige fra nummererede overskrifter, nummererede punkter i en liste, nummererede sætningskonstruktioner og især formelnumre.

Krydsreferencer i L^AT_EX er en to-trins proces, først placeres

```
\label      \label{\unik nøgle}
```

efter det man gerne vil referere til. Den *unikke nøgle* er bare en tekststreng og må indeholde stort set alle ASCII-tegn (*i.e. ikke de danske*) ej heller tegnet %. Den label man nu har placeret, vil så gemme nummeret på det man kan referere til lige her, samt siden hvorpå labelen er placeret. Et simpelt eksempel er naturligvis.⁷

Eksempel 2.21

```
\section{Overskrift}
\label{sec:overskrift}
```

Man referer så via de to makroer

```
\ref      \ref{\unik nøgle}
\pageref  \pageref{\unik nøgle}
```

Den første giver det nummer man henviser til, mens den anden giver sidetallet for den side hvorpå `\label{\unik nøgle}` er placeret.

Sidebemærkning 2.13. Referencer fungerer via den eksterne `.aux`-fil, hvilket vil sige, at når man har sat en ny label skal man lige huske at oversætte dokumentet to gange (en gang for at få en ny label skrevet til `.aux`-filen, og en gang for at `\ref` kan få fat på den nye label). Det samme gør sig gældende når man efter at have indsat noget tekst får flyttet labels over på andre sider. Man vil så se advarslen

```
LaTeX Warning: Label(s) may have changed. Rerun to get cross-references right.
```

til slut i oversættelsesinformationerne fra L^AT_EX.

Sidebemærkning 2.14. Husk, at sørge for at anvende unikke labels. Det er ikke nogen fejl i L^AT_EX at have flere labels med det samme navn, det giver bare ukorrekte krydsreferencer. L^AT_EX giver følgende advarsel:

```
LaTeX Warning: Label `key' multiply defined.
```

og i slutningen af meddelelserne

```
LaTeX Warning: There were multiply-defined labels.
```

⁷ At jeg her anvender `sec:` som en del af den unikke nøgle, har ikke den større betydning, så er det bare lettere at se hvilken type reference der er tale om.

Fodnoter

Sidebemærkning 2.15. Skulle man have refereret til en label som L^AT_EX ikke kan finde vil den skrive ?? på det sted i teksten hvor man refererer til denne label og i meddelelserne skrive

```
LaTeX Warning: Reference `key1' on page 1 undefined on input line 6.
```

```
...
```

```
LaTeX Warning: There were undefined references.
```

Dette kan være fordi man mangler at kompilere en gang til eller fordi man har stavet nøglen forkert.

Eksempel

```
En label: \label{refkst} lige her, vil referere til det seneste refererbare i dokumentet. Her er det >>\ref{refkst}<< (afsnittet vi er i) på side-\pageref{refkst} (i.e. denne side).
```

```
En label: lige her, vil referere til det seneste refererbare i dokumentet. Her er det »2.4« (afsnittet vi er i) på side 35 (i.e. denne side).
```

Eksempel 2.22

Sidebemærkning 2.16. Det er en god ide, at man lader være med at anvende mellemrum i den unikke nøgle. Det potentielle problem ved mellemrum er, at ét og to mellemrum er to forskellige ting så har man labelen `h_n` og man ved en fejl er kommet til at referere til `h_n` så kan det være ret svært at finde fejlen bagefter.

Sidebemærkning 2.17. Med klassen *memoir* findes der desuden en `\titleref` makro som ved en label til en overskrift faktisk vil give overskriften. Den anvender vi her i bogen for at kunne henvise til subsections, da disse ikke er nummereret.

`\titleref`

2.5 Fodnoter

Fodnoter skrives naturligvis med den ret oplagte makro

```
\footnote{<tekst>}
```

`\footnote`

Som stort set kan anvendes overalt i teksten. Der er dog visse steder man enten ikke bør anvende fodnoter eller skal anvende dem på den helt rigtige måde for at det fungerer.

Man bør ikke lave fodnoter indeni noget matematik, idet nummereringen (som normalt er et hævet tal) nemt kan forveksles med en potens.

I afsnitsoverskrifter, figur- og tabelcaptions skal man passe ekstra meget på. I afsnit 9.2 på side 225 forklarer vi blandt andet hvad man skal gøre i denne situation.

En fodnote vil som navnet antyder blive skrevet ved sidens fod. Indenfor visse områder har man i stedet tradition for at skrive (fod)noterne i slutningen af dokumentet. Dette kaldes også *endnotes* eller *slutnoter*. Hvordan man anvender slutnoter i L^AT_EX behandles også i afsnit 9.2 på side 225.

endnotes
slutnoter

Man kan i L^AT_EX også sagtens lave noter i marginen af ens dokument, hvilket vi f.eks. har anvendt i denne bog. Dette vil vi ikke komme nærmere ind på, hverken her eller i senere kapitler. Den interesserede læser kan i stedet slå op i [Mittelbach og Goossens \(2004\)](#) eller evt. [Wilson \(2010\)](#) som også behandler emnet generelt.

2. Almindelig tekst

2.6 Indholdsfortegnelse

Indholdsfortegnelsen aktiveres i L^AT_EX via makroen

```
\tableofcontents \tableofcontents
```

som placeres på det sted hvor man ønsker at indsætte sin indholdsfortegnelse.

Sidebemærkning 2.18. Man skal så huske, at første gang `\tableofcontents` aktiveres i et dokument, skal man lige kompilere dokumentet mindst to gange.⁸ Det samme gælder hvis man har tilføjet et nyt afsnit, så skal dokumentet kompileres mindst to gange før indholdsfortegnelsen er helt opdateret.

Hvad der så rent faktisk kommer med i indholdsfortegnelsen kommer så an på konfigurationen (standard opsætningen varierer fra klasse til klasse). Som oftest vil man få overskrifterne (titlerne) fra `\part`, `\chapter`, `\section` og `\subsection`. Hvor langt man går ned i hierarkiet af afsnitoverskrifter kan nemt konfigureres se afsnit 14.3 på side 345.

Det er desuden vigtigt at bemærke at alle de stjernede varianter af afsnitmakroerne *ikke* kommer med i indholdsfortegnelsen. Dette kan man dog også komme uden om, hvis man ønsker at have overskrifterne med i indholdsfortegnelsen, men ikke ønsker nogen afsnitsnumre. Se igen afsnit 14.3 på side 345. I [Eksempel 2.23](#) finder man et simpelt dokument med en indholdsfortegnelse.

Sidebemærkning 2.19. Der er en lille hage ved denne automatisk genererede indholdsfortegnelse: Alt hvad man skriver i en afsnitoverskrift kommer med i indholdsfortegnelsen, f.eks. et tvunget lineskift. Hvad man skal gøre for at redde sig ud af denne situation forklares i afsnit 14.3 på side 345.

Eksempel 2.23

```
\tableofcontents
\section{En sektion}
\subsection{En undersektion}
\section{En anden sektion}
\paragraph{En paragraf overskrift}
\section*{En uden nummer}
```

Indhold

1 En sektion	1
1.1 En undersektion	1
2 En anden sektion	1
1 En sektion	
1.1 En undersektion	
2 En anden sektion	
En paragraf overskrift	
En uden nummer	

⁸ Indgangene til indholdsfortegnelsen skrives til en fil (`.toc`) som så automatisk inkluderes af `\tableofcontents` (hvis den findes). Men hvis man ikke har bedt om en indholdsfortegnelse, er der jo ingen grund til at registrere noget. Så når man første gang har `\tableofcontents` med i sit dokument, så vil L^AT_EX begynde at registrere afsnit, hvorefter indholdsfortegnelsen er til rådighed fra næste oversættelse.

Justering af tekst

Se desuden afsnit 9.1 på side 223.

2.7 Justering af tekst

Centrering

Et stykke tekst som skal fremhæves ved centrering, pakker man ind i environmentet *center*.⁹

```
tekst tekst tekst tekst tekst før
\begin{center}
\itshape
centreret tekst
\end{center}
tekst tekst tekst tekst tekst efter
```

```
tekst tekst tekst tekst tekst før
          centreret tekst
tekst tekst tekst tekst tekst efter
```

center

Eksempel 2.24

Bemærk at der *ikke* vil være nogen form for ordddeling indenfor *center*-environmentet.

Sidebemærkning 2.20. Hvis man ønsker at centrere teksten i et allerede eksisterende environment, så anvender man *ikke center*-environmentet, man anvender i stedet makroen `\centering`, som bare siger at nu skal teksten altså være centreret. Fordelen er at `\centering` *ikke* efterlader sig ekstra vertikalt mellemrum, hvilket vi jo kan se at *center* jo gør. Det mest almindelige sted man ser brugen af `\centering` er inden i *figure*-environmentet, for at centrere figuren, se Kapitel 6 på side 135.

`\centering`

Venstre-/højrejustering af tekstafsnit

Gøres med hhv. *flushleft* og *flushright*. Bemærk, at dette slår også orddelingen fra indenfor dette environment.

flushleft
flushright

```
tekst før
\begin{flushright}
Dette er noget lang tekst\
som bare bliver ved og ved
\end{flushright}
tekst efter
```

```
tekst før
          Dette er noget lang tekst
          som bare bliver ved og ved
tekst efter
```

Eksempel 2.25

flushright kan jo f.eks. anvendes til at lave en underskrift på en afleveringsopgave. Man skal bemærke at inden for dette environment kommer der ingen ordddeling.

Til brug inden i andre environments, findes der i dette tilfælde også makroer svarende til `\centering`:

```
environment flushleft ↔ makro \raggedright
environment flushright ↔ makro \raggedleft
```

`\raggedright`
`\raggedleft`

Justering med indbygget orddeling

Skulle man have brug for justering som beskrevet ovenfor, så skal man anvende pakken *ragged2e* som giver makroerne

ragged2e

⁹ FixMe Note: fejl: manglende luft

2. Almindelig tekst

```
\Centering
\RaggedRight
\RaggedLeft
```

Disse er især anvendelige i f.eks. tabeller med meget tekst, samt i smalle søjler. Tilsvarende environments:

```
Center
FlushRight
FlushLeft
```

Pakken giver også en makro så man kan komme tilbage til den normale tilstand, se [Schröder \(2003\)](#) for mere information.

Citater

quote
quotation

L^AT_EX har to environments til at tage sig af citater – *quote* og *quotation*. Begge laver afstand til resten af teksten og rykker hele teksten lidt ind fra højre og venstreside.

Eksempel 2.26

```
Tekst
\begin{quote}
  x x x x x x x x x x x
  x x x x x x x x x x x
  x x x x x x x x x x x
  x x x x x x x x x x x
\end{quote}
Noget mere tekst
```

```
Tekst

  x x x x x x x x x x x x x
  x x x x x x x x x x x x x
  x x x x x x x x x x x x x

Noget mere tekst
```

Det er lidt hip som hap hvilken en man benytter. Man kan så sammensætte dette med nogle font- og størrelsesændringer.

Eksempel 2.27

```
Tekst
\begin{quotation}
  \footnotesize\itshape
  x x x x x x x x x x x
  x x x x x x x x x x x
  x x x x x x x x x x x
\end{quotation}
Noget mere tekst
```

```
Tekst

  x x x x x x x x x x x x x x x
  x x x x x x x x x x x x x x x

Noget mere tekst
```

Bemærk, at teksten i [Eksempel 2.27](#) er indenteret, sammenlign med [Eksempel 2.26](#).

Se desuden [Eksempel A.18 på side 357](#) for et eksempel på hvordan man kan anvende *quote*-environmentet og *babel*-pakken til at lave sit eget specielle citatenvironment beregnet på engelske citater og hvor man så automatisk skifter til engelsk orddeling indeni dette environment.

2.8 Lister

L^AT_EX har grundlæggende tre typer lister – en punktopstilling, en nummereret liste samt en beskrivende liste.

Punktopstilling

itemize
\item

En punktopstilling laves med environmentet *itemize*. Hvert punkt startes med *\item*.

Nummereret liste

<pre>tekst \begin{itemize} \item Punkt et \item Punkt to \item Punkt tre \end{itemize} tekst</pre>	tekst <ul style="list-style-type: none">• Punkt et• Punkt to• Punkt tre tekst
--	--

Eksempel
2.28

Man skal være opmærksom på, at •-en ændrer sig når man sætter lister inden i hinanden.

<pre>tekst \begin{itemize} \item Punkt et \begin{itemize} \item punkt \end{itemize} \item Punkt tre \end{itemize} tekst</pre>	tekst <ul style="list-style-type: none">• Punkt et<ul style="list-style-type: none">– punkt• Punkt tre tekst
---	--

Eksempel
2.29



Formaning 2.21 (Beviser og lister).

Det er ikke altid verdens bedste ide at skrive en opgaveløsning som en liste, eller opdele et bevis som en liste. Det er spild af plads idet de fleste lister jo er indrykket fra venstre margin. Det er især slemt hvis nogle af punkterne så fylder flere sider.

Der er flere måder man kan løse dette på. Den første er at lave være med at anvende lister og i stedet anvende kommandoen

```
\paragraph{<tekst>}
```

Den fungerer på samme måde som f.eks. `\section{<tekst>}`, bare uden nummer, og den skifter *ikke* linie efter `<tekst>`. Makroen sørger selv for at der kommer noget luft over. Er man *memoir*-bruger, kan den justeres på samme måde som herunder (koden som er anvendt er standard indstillingen for den i *memoir*)

```
\setbeforeparaskip{3.25ex plus 1ex minus .2ex} % luft før
\setparaheadstyle{\normalsize\bfseries} % font justering
```

Den anden metode er at anvende en type liste som ikke laver indrykning af indgangene. Dette kan `enumitem` (beskrives på side 241), se [Bezos \(2011\)](#) og konfigurationen `wide`.

Eksempel
2.30

Nummereret liste

En nummeret liste laves med environmentet *enumerate*

enumerate

<pre>tekst \begin{enumerate} \item et \item to \item tre \end{enumerate} tekst</pre>	tekst <ol style="list-style-type: none">1. et2. to3. tre tekst
--	---

Eksempel
2.31

2. Almindelig tekst

`enumerate` Nogle gange vil man gerne kunne benytte f.eks. a, b, c, ... som nummerator. Til dette formål har man pakken `enumerate`, som så giver `\begin{enumerate}` et valgfrit argument, f.eks.

Eksempel 2.32



```
\usepackage{enumerate}
----- slut preamble -----
\begin{enumerate}[(a)]
\item et
\item to
\item tre
\end{enumerate}
```

(a) et
(b) to
(c) tre

Når `enumerate`-pakken anvendes vil *enumerate* kigge efter tegnene »a« (små bogstaver), »A« (store bogstaver), »i« (små romertal), »I« (store romertal) og »1« (almindelige tal, standard) og så skrive selve tælleren (nummereringen) efter dette valg. Udover denne begrænsning er det helt op til en selv at bestemme hvad man vil skrive.

`enumerate` Da *enumerate* reagerer på *alle* de nævnte 5 tegn, kan man se sig nødsaget til at skulle beskytte (med et {}-par) visse dele af argumentet angivet i []. Se de første to punkter i [Eksempel 2.33](#).

Eksempel 2.33



```
\usepackage{enumerate}
----- slut preamble -----
\begin{enumerate}[Aksiom A]
\item $1+1\neq 3$
\item $e^{\pi i}+1=0$\\
noget tekst på næste linie
\end{enumerate}
og den rigtige løsning
\begin{enumerate}[{Aksiom} A]
\item $1+1\neq 3$
\item $e^{\pi i}+1=0$\\
noget tekst på næste linie
\end{enumerate}
```

AksAom A $1 + 1 \neq 3$
BksBom B $e^{\pi i} + 1 = 0$
noget tekst på næste linie
og den rigtige løsning
Aksiom A $1 + 1 \neq 3$
Aksiom B $e^{\pi i} + 1 = 0$
noget tekst på næste linie

Man bemærker også, at indenteringen automatisk justeres alt efter bredden af det valgfri argument.

Dette med, at man kan beskytte dele af argumentet, forklarer desuden hvorfor `\textbf{a}` ikke ville give os en fed tæller med små bogstaver. Tegnet »a« er simpelthen skjult fra *enumerate*-parseren. For at løse dette skal man bare sørge for at anvende de lange fontmakroer (her `\bfseries`).

Eksempel 2.34



```
\usepackage{enumerate}
----- slut preamble -----
\begin{enumerate}[\bfseries a.]
\item et
\item to
\item tre
\end{enumerate}
```

a. et
b. to
c. tre

Som nævnt andetsteds kan man i \LaTeX referere til stort set hvad som helst som er autonummereret af \LaTeX . Dette gælder naturligvis også punkterne i en nummeret liste. Bare smid en `\label` efter `\item`.

Beskrivende liste

```
\begin{enumerate}
\item \label{enum:1} Punkt et
\item \label{enum:2} Punkt to
\end{enumerate}
\ref{enum:1} og \ref{enum:2}
```

1. Punkt et
2. Punkt to

1 og 2

Eksempel
2.35

Man skal være opmærksom på, at man får kun fat på nummeret (eller bogstavet), *ikke* formateringen.

For at få endnu bedre kontrol over udformningen af listerne samt formateringen med i referencerne se pakken `enumitem`, lettere beskrevet i afsnit 9.7.2 på side 241.

`enumitem`

Beskrivende liste

En beskrivende liste genereres med *description*. Teksten man gerne vil beskrive, angives som valgfrit argument til `\item`.¹⁰

`description`

```
\begin{description}
\item[Aksiom A]  $1+1 \neq 3$ 
\item[Aksiom 28]  $e^{\pi i} + 1 = 0$  \
noget tekst på næste linie
\end{description}
```

Aksiom A $1 + 1 \neq 3$
Aksiom 28 $e^{\pi i} + 1 = 0$
noget tekst på næste linie

Eksempel
2.36

Bemærk, at teksten er skrevet med fed (dette kan konfigureres, se [Mittelbach og Goossens \(2004\)](#)) og at en lang titeltekst bare rykker ind i selve beskrivelsen.

Vi vil i et senere kapitel (9 på side 223) se lidt nærmere på andre pakker som kan lave lister, samt hvordan man kan lave egne liste konstruktioner. Se f.eks. listen på side 10 som viser dokumentklasse options. Denne beskrivende liste er lavet således at hvis argumentet er langt, laves der automatisk lineskift før teksten.

Iøvrigt kan man desuden tage et kig på L^AT_EX sætningskonstruktioner som egentlig kan ses som specielle lister, hvor hvert punkt (som er et environment for sig selv) har en overskrifter og et nummer. For at starte et nyt punkt i denne specielle form for liste, skal man bare anvende environmentet endnu en gang. Se kapitel 5 på side 115.

2.9 Simpel indsættelse af titel, forfatternavn og dato

Til mindre dokumenter (artikler, opgavebesvarelser, etc.) kan har man tit brug for at kunne tilføje en simpel titel, forfatterinformation samt en dato. Den nemmeste metode er at anvende følgende syntaks:

```
\title{\{titeltekst\}}
\author{\{forfattertekst\}}
\date{\{datotekst\}}
\maketitle
\today
```

`\title`
`\author`
`\date`
`\maketitle`
`\today`

Det er vigtigt at de tre første makroer kommer *før* `\maketitle` og at `\maketitle` selv kommer *efter* `\begin{document}`. Hvis man undlader at anvende `\date`, vil man få datoen for sidste kompilering (hvilket er det samme som at have anvendt `\date{\today}`). Skulle man omvendt ikke ønske nogen dato, skal man bare anvende `\date{}`.

¹⁰ Man kan desuden anvende dette valgfrie argument ved *itemize* eller *enumerate*, selv om dette sjældent er en god ide.

2. Almindelig tekst

I standardklasserne (*article*, *report* og *book* samt også *memoir*) kan man adskille flere forfattere med makroen `\and`, dvs. man anvender

Eksempel 2.37 `\author{Forfatter1 \and Forfatter2}`

I andre klasser skal man skrive det hele lidt anderledes (eksempelvis i *amsart*). Man har desuden muligheden for til hver forfatter at kunne tilføje en speciel fodnote, kaldet `\thanks`, som kun kan anvendes i `\title` eller `\author`. Dette kunne f.eks. anvendes til at tilføje en email-adresse eller lignende. Her er et eksempel på det hele i brug.

Eksempel 2.38

```
\author{En Forfatter\thanks{%
Støttet af SU-styrelsen.}}
\title{Min opgaveløsning}
\date{\today}
\maketitle
```

Min opgaveløsning

En Forfatter*

13. september 2011

*Støttet af SU-styrelsen.

Tip 2.22. Hvis nu ens lille dokument ikke fylder mere end én enkelt side, så er det jo lidt tosset at skulle have et sidetal på siden. Sidetallet kan i så fald fjernes ved at placere

Eksempel 2.39 `\thispagestyle{empty}`

efter en eventuel `\maketitle`.

Anvender man slet ikke `\maketitle` så kan man i stedet placere

Eksempel 2.40 `\pagestyle{empty}`

i dokumentets preamble.

Faktisk kan dette i visse tilfælde gøres automatisk, se Kodetip 12.3 på side 327.

2.10 Hvordan håndterer man URLer, stier og filnavne

Når man ønsker at skrive navnet på en fil eller ønsker at skrive en URL i teksten, da gælder helt særlige regler for hvordan man må linieombryde disse. Helt specifikt må man jo *aldrig* orddele et filnavn eller en URL, thi hvordan kan man se om bindestregen skal med i navnet eller ej.

`url` Til at løse dette problem anvendes pakken `url` som hovedsageligt giver makroen `\url` (samt makroen `\path` som egentlig bare er `\url` med et andet navn). Denne makro har følgende fortræffeligheder:

- Det må kun linieombrydes efter tegn så som `»/«`, `».«`, `»_«` etc. (listen kan konfigureres, se [Arseneau \(2003d\)](#)).
- Teksten skrives automatisk i skrivemaskinefont (monospace).
- Teksten skrives faktisk mere eller mindre ordret (verbatim, se afsnit 9.10), det vil sige at man uden videre kan anvende tegn så som `»~«` eller `»_«` i en URL eller sti eller filnavn, de skal ikke beskyttes eller skrives om på nogen måde.

Hvordan håndterer man URLer, stier og filnavne

- (d) Anvender man hyperref i sit dokument så laves alle `\url`'er automatisk om til hyperlinks i PDF-filen.

I dette dokument anvender vi naturligvis også `\url`.

```
\url{http://www.imf.au.dk/system/latex/bog}  
\path{/home/daleif/www/}
```

```
http://www.imf.au.dk/system/latex/bog  
/home/daleif/www/
```

**Eksempel
2.41**

Introduktion til matematik

Når man arbejder med en eller anden form for videnskab vil man altid, på et eller andet tidspunkt, få brug for at kunne skrive matematiske formler. Dette kapitel er en introduktion til matematik i L^AT_EX. Der vil blive lagt vægt på de mest almindelige konstruktioner, men mens man stadigvæk holder det hele på et nybegynderniveau. Mere uddybende behandling af matematikken udskydes til det efterfølgende kapitel.

Anser man ikke sig selv som nybegynder, bør man nu alligevel gøre sig bekendt med konstruktionerne anvendt i dette kapitel, det er basal viden for alle L^AT_EX-brugere.

Vi vil til enhver tid antage at man har husket at aktivere pakkerne `amsmath` og `amssymb`, disse vil derfor ikke blive inkluderet i eksemplerne nedenfor. Bemærk at visse klasser eller pakker automatisk loader `amsmath` (eks. `amsart` og `amsbook`).

`amsmath`
`amssymb`

3.1 Matematik i L^AT_EX

Når man selv sidder og skriver matematik på et stykke papir skelner man mellem to forskellige måder at skrive matematikken på. (a) Som del af den almindelige tekst og (b) fremhævede formler på linier for sig selv. Stilen (a) kaldes *textstyle*, *inline/integreret* eller *tekstbaseret matematik*, mens (b) kaldes *fremhævet* eller *displaystyle*. Tilstanden L^AT_EX er i når den behandler matematisk materiale kaldes naturligt nok *matematik-mode* eller *math-mode*.

textstyle
inline
fremhævet
displaystyle
matematik-mode

Den typografiske forskel mellem de to er, at matematikken i *textstyle* bliver tilpasset til at kunne stå på en almindelig tekstlinje, dvs. visse konstruktioner er presset mere sammen, thi ellers ville de forstyrre afstanden mellem tekstlinjerne. På [Eksempel 3.1](#) kan man se forskellen ved brøker og sumtegn.

```
Dette er et lille eksempel med
tekst matematik med
 $\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$ 
og dernæst herunder samme men
nu som fremhævet matematik

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}.$$

```

Dette er et lille eksempel med tekst matematik med $\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$ og dernæst herunder samme men nu som fremhævet matematik

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}.$$

Eksempel

3.1



Fra eksemplet kan man også se at man kan skrive tekst-matematik mellem et `$. . . $`-par og en fremhævet unummereret formel indenfor *equation** environmentet. Før vi går videre skal vi dog først se lidt nærmere på syntaksreglerne indenfor matematik mode.

`$. . . $`
*equation**

3.1.1 Lidt syntaksregler i matematik mode

Mellemrum tæller ikke

På grund af det potentielt store forbrug af symboler samt reglerne for hvilken mellemrum der skal anvendes mellem hvilke symboler, vil mellemrum i kildekoden ikke betyde noget indenfor *math-mode*.

3. Introduktion til matematik

Eksempel 3.2











dvs. $a + b = c$ er det samme som $a + b = c$

dvs. $a + b = c$ er det samme som $a + b = c$

Dette er jo en klar fordel, idet symbolmakroer jo normalt ellers har tendens til at sluge mellemrummet lige efter.

L^AT_EX indsætter normalt selv de mellemrum der skal til omkring symbolerne. Men til tider kan det være en fordel selv at kunne indsætte en eller anden form for standardiseret mellemrum (dvs. af faste størrelser). For eksempel bør man indsætte $\,$, før dx i $\int f(x) dx$ ($\int f(x) \, dx$), hvilket er med til at øge læsbarheden. De forskellige muligheder kan ses i tabel 3.1. Man bør især kende til makroerne: $\,$, $\:$, $\;$, $_$, \enskip , \quad , \qquad , $\!$, \negmedspace og \negthickspace .

Tabel 3.1: Makroer til indsættelse af vandret mellemrum

Kilde	Effekt	Beskrivelse	Størrelse
$\,$		thinspace	$3 \mu^a$
$\:$		medspace	$4 \mu + \dots^b$
$\;$		thickspace	$5 \mu + \dots$
$_$		ord mellemrum	
\enskip		enskip	0.5 em^a
\quad		quad	1 em
\qquad		$2 \times \text{quad}$	2 em
$\!$		negativ thinspace	-3μ
\negmedspace		negativ medspace	$-4 \mu + \dots$
\negthickspace		negativ thickspace	$-5 \mu + \dots$

^a En em er det samme som skriftstørrelsen som er i brug lige nu og en μ (math unit) er $1/18 \text{ em}$. Teksten her er af størrelsen 8.0 pt .

^b $+\dots$ betyder at denne del kan strække sig en smule.

Ingen danske vokaler i matematik

Man kan ikke anvende vokaler (eller andre accent) bogstaver indenfor matematik. Dette vil L^AT_EX brokke sig over i en advarsel, så som »LaTeX Warning: Command \r invalid in math mode ...«, hvis man prøver at skrive Å , indeni noget matematik, og L^AT_EX vil så vælge slet ikke at skrive noget. Man vil under normale omstændigheder ikke have brug for de tre danske vokaler som direkte matematiske variable, de vil snarere komme ind som indices til variable f.eks. $R_{sø}$ for radius af en sø. Hvordan man lave dette rigtigt samt hvordan man behandler fysiske enheder, som f.eks. Å (ångström), skal vi se på senere (se afsnit 3.2.3 på side 57, især Eksempel 3.29 på side 59 samt afsnit 10.4 på side 268).

Begrænset automatisk linieombrydning

En tredje regel er at der i tekst-matematik kun er begrænset mulighed for automatisk ombrydning af en formel. Dette skyldes at der ikke findes nogen universelle regler for hvordan en matematisk formel skal deles hen over en linie.

L^AT_EX har mulighed for at lave automatisk linieombrydning *efter* relationer og operationer, men f.eks. ikke efter komma. Disse ombrydninger er ikke altid optimale. Man finder en større gennemgang af linieombrydning af matematik i afsnit 4.1 på side 83.

Tegnsætning i matematik

Samtidigt *skal* alle linieombrydninger i fremhævet matematik laves af skribenten, thi dette kræver matematisk indsigt, se desuden afsnit 4.1.3 på side 86.¹

3.1.2 Tegnsætning i matematik

Generelt anbefales at man laver sin tekstmæssige tegnsætning som var det normal tekst, dette gælder både i tekstbaseret og fremhævet matematik, med den undtagelse at man i den fremhævede form inkluderer tegnsætningen indenfor matematikken hvilket man ikke gør i tekstbaseret matematik. Se desuden afsnit 4.1.4 på side 89. Her er et simpelt eksempel.

```
For  $f(x)$ ,  $g(x)$  og  $h(x)$  gælder
\l
  f(x)+g(x) = h(x) .
\l
```

For $f(x)$, $g(x)$ og $h(x)$ gælder

$$f(x) + g(x) = h(x).$$

Eksempel
3.3

3.1.3 Fremhævet matematik

Til den fremhævede matematik findes der mange forskellige konstruktioner. Det er umuligt at lave netop een konstruktion, som kan det hele, så derfor må man lære de forskellige konstruktioners normale brug. Det meste af dette udskyder vi til den øvede del af bogen og vil i stedet beskæftige os med de to mest anvendte konstruktioner: *equation* og *align*.

equation
align

Mantra. Der gælder for disse environments og det gælder for de vi skal se senere: Hvis et environment $\langle env \rangle$ til fremhævede formler giver formelnumre da findes der et tilsvarende environment $\langle env^* \rangle$, hvor formelnumrene er slået helt fra.

Sidebemærkning 3.1. Der gælder en enkelt regel for hvordan man skal skrive sine ting i et environment til fremhævede formler:

Man må ikke have blanke linier i et matematik-environment!

Dette gælder for alle environments til fremhævede formler. Ønsker man at gøre koden mere overskuelig via blanke linier, skal man bare huske at starte dem med en »%«.

One-liners

Environmentet *equation* anvendes til formler som kun fylder én linie og man har ingen muligheder for at dele en linie hvis den bliver for lang (der skal man i stedet anvende en anden konstruktion, som evt. kan anvendes indeni en *equation*).

equation

```
\begin{equation}
  f(x)=x^2+1\geq 1\quad
  \forall x\in\mathbb{R}
\end{equation}
```

$$f(x) = x^2 + 1 \geq 1 \quad \forall x \in \mathbb{R} \quad (3.1)$$

Eksempel
3.4

Formlen bliver automatisk nummereret. Ønsker man ikke noget nummer anvendes *equation** i stedet.

*equation**

¹ Dette er ikke helt korrekt, der arbejdes på at implementere denne matematiske indsigt direkte i en L^AT_EX-pakke, men det er ikke ligefrem nogen nem opgave.

3. Introduktion til matematik

Eksempel 3.5

```
\begin{equation*}
e^{\pi i}+1=0
\end{equation*}
```

$$e^{\pi i} + 1 = 0$$

`\[...]` Som et alias for *equation** kan man anvende `\[...]`, det fylder lidt mindre.

Eksempel 3.6

```
\[ \cos^2\theta+\sin^2\theta=1 \]
```

$$\cos^2 \theta + \sin^2 \theta = 1$$

Sidebemærkning 3.2. Det kan dog være en smule vanskelig at se environmentet lavet med `\[...]`, når man er ved at skimme ned gennem sit dokument. Det er derfor være en ide at skrive det på følgende måde

Eksempel 3.7

```
\[
\dots
\]
```

Altså med hhv. »\[« og »\]« på en linie for sig. Med de mere moderne L^AT_EX editorer kan man indsætte

```
\begin{equation*}
...
\end{equation*}
```

nemt via tastaturet eller musen, hvilket er endnu nemmere at overskue. Dette gør det nemmere for skribenten at overskue teksten, men også for senere skribenter som måske har overtaget vedligeholdelsen af dokumentet.

Sidebemærkning 3.3. Generelt er det en god ide at lade `\begin{<env>}` og `\end{<env>}` stå på linier for sig selv, idet dette gør koden nemmere at overskue.²

Flerliniede formler

Når man har et udtryk som af forskellige årsager kræver flere linier vil det oftest være *align* man vil anvende. Som navnet antyder anvendes environmentet til at *flugte* ting med, dvs. sætte ting op, så et fælles punkt står under hinanden. Den simpleste syntaks er:

```
\begin{align}
Venstre & højre \\
Venstre & højre \\
... \\
Venstre & højre \\
\end{align}
```

Forstået på den måde at man via »&« markerer det punkt i sit udtryk hvor man gerne vil justere sine formellinier over hinanden. L^AT_EX sørger så for at punkterne hvor &-erne er placeret bliver justeret (alignet) over hinanden.

Hver formellinie bliver automatisk nummereret.

² Ved godt, at bogen ikke selv følger denne regel i mange af eksemplerne, men det er mere af pladshensyn. I egne dokumenter kan man svine med pladsen lige så meget man har lyst til.

Flerliniede formler

```
\begin{align}
f(x) &= (x-1)(x+1) \\
&= x^2-1
\end{align}
```

$$f(x) = (x-1)(x+1) \quad (3.2)$$

$$= x^2 - 1 \quad (3.3)$$

Eksempel 3.8



Advarsel 3.4. Det &-tegn som markerer alignmentpunktet kan ikke placeres inde i andre konstruktioner, f.eks. midt i en brøk, kvadratrods eller en `\left...\right` konstruktion (alle tre introduceres senere i dette kapitel).

Skulle man ønske kun at nummerere visse af linierne, kan man placere en `\nonumber` (samme som `\notag`), på den formellinie (før en eventuel `\\`) man ikke ønsker nummereret.

`\nonumber`
`\notag`

```
\begin{align}
f(x) &= (x-1)(x+1) \nonumber \\
&= x^2-1
\end{align}
```

$$f(x) = (x-1)(x+1)$$

$$= x^2 - 1 \quad (3.4)$$

Eksempel 3.9

Igen kan man bare anvende ***align****, hvis man slet ikke ønsker nogen nummerering.

*align**

```
\begin{align*}
x &\geq \sin x \\
&\geq x^2, \quad \text{\quad } 0 \leq x \leq 1.
\end{align*}
```

$$x \geq \sin x$$

$$\geq x^2, \quad 0 \leq x \leq 1.$$

Eksempel 3.10

Sidebemærkning 3.5. Vi skal i et senere kapitel se lidt nærmere på hvor man egentlig må dele en formel og hvad man gør i forskellige situationer. Indtil videre vil vi bare nævne at man hovedsageligt bør placere & til *venstre* for relationer eller binære relationer, ligesom vi har gjort i alle vore eksempler. Se afsnit [4.1.1 på side 83](#).

Skulle man alligevel befinde sig i en situation hvor flugtning til højre for en relation er bedste løsning, skal man gøre lidt ekstra, se [Eksempel 3.11](#).

```
\begin{align*}
A = {} & \& B \\
& + C \\
= {} & \& D
\end{align*}
```

$$A = B$$

$$+ C$$

$$= D$$

Eksempel 3.11

Her skal man især bemærke brugen af de to {}-par, som afhjælper mellemrummet omkring lighedstegnet, uden dem ville lighedstegnet komme for tæt på *B* og *D*.

3. Introduktion til matematik

Sidebemærkning 3.6. Umiddelbart er der ikke noget galt i også at anvende *align* til one-liner formler. Hvis der kun er en linie behøver man heller ikke nogen $\&$. Disse kan man så indsætte hvis/når man for brug for flere linier. Men *equation* og *equation** har en indbygget feature som man ikke ser med f.eks. *align*-environmentet. Hvis formlen er kort og teksten lige før formlen er kort, så re-justerer *equation* automatisk sin position i forhold til den foregående tekst, for at kunne udnytte pladsen bedre.

Eksempel 3.12

<pre>tekst \[a=b \] tekst \begin{align*} a=b \end{align*}</pre>	<pre>tekst a = b tekst a = b</pre>
--	--

Desuden vil *equation* kunne trække matematikken lidt sammen for at kunne passe på linien, hvis det er nødvendigt. Det gør *align* ikke.

Sidebemærkning 3.7. Visse L^AT_EX introduktioner (f.eks. Oetiker (2006)) samt visse Journal skabeloner skriver at man kan anvende environmentet *eqnarray* til at skrive flerliniede formler med. Dette er korrekt, men dette environment er så fyldt med fejl at det kun bør anvendes i yderste nødstilfælde, f.eks. hvor man er tvunget til ikke at måtte anvende amsmath-pakken.

Man kan se en opsummering af hvorfor man aldrig skal anvende *eqnarray* i Madsen (2006).

eqnarray

3.1.4 Formelnumre og krydsreferencer

\backslash label Krydsreferencer til nummererede formellinier laves som sædvanligt via \backslash label og en \backslash ref \backslash ref-kommando. \backslash label'en placeres på den formellinie (før en eventuel \backslash) man gerne til referere til.

Eksempel 3.13

<pre>\begin{align} f(x)&=\sqrt{\sin^2\theta-1} \ \backslash\label{eq:6}\backslash g(x)&=32 \ \backslash\label{eq:7} \end{align} (\ref{eq:6}) viser noget og \eqref{eq:7} viser noget andet.</pre>	$f(x) = \sqrt{\sin^2 \theta - 1} \quad (3.5)$ $g(x) = 32 \quad (3.6)$ <p>(3.5) viser noget og (3.6) viser noget andet.</p>
---	--

Eksempel 3.13 viser desuden to forskellige måder at få formelnummeret indkapslet i et $()$ -par, enten pr. håndkraft eller via \backslash eqref. Fordelen ved \backslash eqref frem for $(\backslash$ ref{...}) ses bedst hvis man refererer til noget mens man er i kursiv tekst (f.eks. i formuleringen af en sætning).

\backslash eqref

Eksempel 3.14

\backslash textit{ $(\backslash$ ref{eq:6}) versus \backslash eqref{eq:6}}	(3.5) versus (3.5)
--	------------------------

Her ser man at den første variant ret grim i kursiv (endnu værre i CM fonten), så anvend hellere \backslash eqref til at referere til formelnumre. Faktisk er der visse teksteditorer som automatisk kan gøre dette når man refererer til en formel.

Sidebemærkning 3.8. Det er vigtigt at man ikke blander `\label` og `\nonumber` sammen på samme linie. Environments fra `amsmath` (f.eks. *align*) vil i så fald give en fejlmeddelelse (hvilket også er at forvente), men visse konstruktioner gør ikke (*equation* og *eqnarray*) noget som kan give nogle ret underlige resultater, se f.eks. [Madsen \(2006\)](#).

Man skal også huske at man med `amsmath`-environments kun kan have én label til hvert formelnummer. Placerer man mere end en, vil man få en fejl.

Har man brug for at lave andre ting med formelnummeret henvises man til afsnit [4.4 på side 99](#), nævnes blandt andet hvordan man kan få dokumentet til kun at nummerere de formler som der faktisk refereres til.

3.2 Symboler og operatorer

3.2.1 Forskellige symboler

Symboler skal der jo til – hvor mange vil du ha’?

Følgende er en lille oversigt over nogle af de symboler man har til rådighed i matematik. Fælles for dem alle er at man kun har adgang til dem indenfor `math`-mode.

Sidebemærkning 3.9. Husk at vi som altid antager at man har aktiveret `amssymb`, eller en pakke som tilbyder tilsvarende symboler.

Symbolerne er hentet fra den store symboloversigt [Pakin \(2003\)](#), kapitel 3, så det er her man skal kigge hvis man ikke lige kan finde det symbol man mangler. Husk dog at kigge tabellerne i [Pakin \(2003\)](#) helt igennem, der er visse pakker som det ikke anbefales at man anvender, af forskellige årsager. Umiddelbart er de primære pakker `amssymb`, `amsfonts` og `stmaryrd`.

Skulle man foretrække » ϵ « frem for » ε « kan man nemt udskifte `\epsilon` med `\varepsilon`, se [Eksempel 3.15](#).

```
\epsilon og så skifter vi  
\renewcommand\epsilon{\varepsilon}  
\epsilon
```

ϵ og så skifter vi ε

**Eksempel
3.15**

³ **FIXme Dødelige:** De følgende lister af symboler skal gennemgås for at se om der er flere relevante symboler som skal med

De fleste af relationerne i [Tabel 3.4 på side 53](#) kan negeres ved at placere `\not` foran. `\not`

```
x\not> 1
```

$x \not> 1$

**Eksempel
3.16**

³ **FIXme Dødelige:** De følgende lister af symboler skal gennemgås for at se om der er flere relevante symboler som skal med

3. Introduktion til matematik

Tabel 3.2: Græske bogstaver

Små		Store		Navn
Kode	Output	Kode	Output	
<code>\alpha</code>	α	A	A	alfa
<code>\beta</code>	β	B	B	beta
<code>\gamma</code>	γ	<code>\Gamma</code>	Γ	gamma
<code>\delta</code>	δ	<code>\Delta</code>	Δ	delta
<code>\epsilon</code>	ϵ	E	E	epsilon
<code>\zeta</code>	ζ	Z	Z	zeta (udt. sæta)
<code>\eta</code>	η	H	H	eta (udt. æta)
<code>\theta</code>	θ	<code>\Theta</code>	Θ	theta (udt. tæta)
<code>\iota</code>	ι	I	I	iota (udt. jota)
<code>\kappa</code>	κ	K	K	kappa
<code>\lambda</code>	λ	<code>\Lambda</code>	Λ	lambda (udt. lamda)
<code>\mu</code>	μ	M	M	my
<code>\nu</code>	ν	N	N	ny
<code>\xi</code>	ξ	<code>\Xi</code>	Ξ	xi (udt. ksi)
<code>o</code>	o	O	O	omikrom
<code>\pi</code>	π	<code>\Pi</code>	Π	pi
<code>\rho</code>	ρ	P	P	rho (udt. ro)
<code>\sigma</code>	σ	<code>\Sigma</code>	Σ^a	sigma
<code>\tau</code>	τ	T	T	tau
<code>\upsilon</code>	υ	<code>\Upsilon</code>	Υ^b	ypsilon
<code>\phi</code>	ϕ	<code>\Phi</code>	Φ	phi (udt. fi)
<code>\chi</code>	χ	X	X	khi (udt. ki)
<code>\psi</code>	ψ	<code>\Psi</code>	Ψ	psi
<code>\omega</code>	ω	<code>\Omega</code>	Ω	omega
Alternative				
<code>\varepsilon</code>	ϵ			epsilon
<code>\varkappa</code>	κ			kappa
<code>\varrho</code>	ρ			rho
<code>\varphi</code>	ϕ			phi
<code>\varpi</code>	ω			pi
<code>\varsigma</code>	ζ			sigma
<code>\vartheta</code>	θ			theta

KILDE: Poulsen (2001)

NOTE: Husk at udseendet afhænger meget af den matematikfont der anvendes.

^a Dette er *ikke* sumtegnet, anvend »\sum« i stedet.

^b Alternativt kan man anvende »Y« i stedet for \Upsilon.

Forskellige symboler

Tabel 3.3: Nogle operatorsymboler

<code>\pm</code>	\pm	<code>\mp</code>	\mp	<code>\circ</code>	\circ	<code>\times</code>	\times	<code>\star</code>	\star
<code>\oplus</code>	\oplus	<code>\otimes</code>	\otimes	<code>\oslash</code>	\oslash	<code>\odot</code>	\odot	<code>\ast</code>	\ast
<code>\cap</code>	\cap	<code>\cup</code>	\cup	<code>\cdot</code>	\cdot	<code>\bullet</code>	\bullet	<code>\div</code>	\div
<code>\setminus</code>	\setminus	<code>\sqcap</code>	\sqcap	<code>\sqcup</code>	\sqcup	<code>\vee</code>	\vee	<code>\wedge</code>	\wedge
<code>\barwedge</code>	$\bar{\wedge}$	<code>\veebar</code>	\veebar	<code>\dagger</code>	\dagger	<code>\ddagger</code>	\ddagger	<code>\diamond</code>	\diamond
<code>+</code>	$+$	<code>-</code>	$-$	<code>*</code>	$*$	<code>/</code>	$/$		

Tabel 3.4: Nogle relationssymboler

<code>\sim</code>	\sim	<code>\simeq</code>	\simeq	<code>\approx</code>	\approx	<code>\cong</code>	\cong
<code>\mid</code>	\mid	<code>\propto</code>	\propto	<code>\parallel</code>	\parallel	<code>\perp</code>	\perp
<code>\doteq</code>	\doteq	<code><</code>	$<$	<code>></code>	$>$	<code>=</code>	$=$
<code>\geq</code>	\geq	<code>\leq</code>	\leq	<code>\gg</code>	\gg	<code>\ll</code>	\ll
<code>\neq</code>	\neq	<code>\geqq</code>	\geqq	<code>\leqq</code>	\leqq	<code>\geqslant</code>	\geqslant
<code>\leqslant</code>	\leqslant	<code>\ggg</code>	\ggg	<code>\lll</code>	\lll	<code>\lessapprox</code>	\lessapprox
<code>\asymp</code>	\asymp	<code>\models</code>	\models	<code>\prec</code>	\prec	<code>\preceq</code>	\preceq
<code>\succ</code>	\succ	<code>\backsim</code>	\backsim	<code>\backsimeq</code>	\backsimeq	<code>\circeq</code>	\circeq
<code>\Vdash</code>	\Vdash	<code>\vdash</code>	\vdash				

Tabel 3.5: Nogle inklusioner

<code>\in</code>	\in	<code>\ni</code>	\ni	<code>\notin</code>	\notin	<code>\subset</code>	\subset
<code>\subseteq</code>	\subseteq	<code>\subseteqq</code>	\subseteqq	<code>\supset</code>	\supset	<code>\supseteq</code>	\supseteq
<code>\sqsubset</code>	\sqsubset	<code>\sqsupset</code>	\sqsupset	<code>\sqsubseteq</code>	\sqsubseteq	<code>\sqsupseteq</code>	\sqsupseteq
<code>\supseteqq</code>	\supseteqq	<code>\not\subset</code>	$\not\subset$	<code>\subsetneq</code>	\subsetneq	<code>\subsetneqq</code>	\subsetneqq
<code>\not\supset</code>	$\not\supset$	<code>\supsetneq</code>	\supsetneq	<code>\supsetneqq</code>	\supsetneqq		

Tabel 3.6: Diverse symboler (opfører sig som alm. bogstaver mht. luften omkring sig)

<code>\forall</code>	\forall	<code>\exists</code>	\exists	<code>\in</code>	\in
<code>\ni</code>	\ni	<code>\top</code>	\top	<code>\bot</code>	\bot
<code>\ell</code>	ℓ	<code>\partial</code>	∂	<code>\imath</code>	\imath
<code>\jmath</code>	\jmath	<code>\Im^a</code>	\Im	<code>\Re^a</code>	\Re
<code>\hbar</code>	\hbar	<code>\hslash</code>	\hslash	<code>\nexists</code>	\nexists
<code>\aleph</code>	\aleph	<code>\infty</code>	∞	<code>\angle</code>	\angle
<code>\neg</code>	\neg	<code>\nabla</code>	∇	<code>\mho</code>	\mho
<code>\triangle</code>	\triangle	<code>\backslash</code>	\backslash	<code>\prime</code>	\prime
<code>\Box</code>	\Box	<code>\Diamond</code>	\Diamond	<code>\blacktriangledown</code>	\blacktriangledown
<code>\backprime</code>	\backprime	<code>\square</code>	\square	<code>\blacklozenge</code>	\blacklozenge
<code>\bigstar</code>	\bigstar	<code>\triangledown</code>	\triangledown	<code>\lozenge</code>	\lozenge
<code>\emptyset</code>	\emptyset	<code>\varnothing</code>	\varnothing	<code>\blacksquare</code>	\blacksquare
<code>\blacktriangle</code>	\blacktriangle	<code>\measuredangle</code>	\measuredangle	<code>\sphericalangle</code>	\sphericalangle
<code>\complement</code>	\complement	<code>\clubsuit</code>	\clubsuit	<code>\spadesuit</code>	\spadesuit
<code>\heartsuit</code>	\heartsuit	<code>\diamondsuit</code>	\diamondsuit		

^a Se [Eksempel 3.34 på side 61](#) for forklaring på hvordan man laver disse om til noget mere relevant.

3. Introduktion til matematik

Sidebemærkning 3.10 (At generere egne symboler ved at kombinere andre.)

Hvis et symbol man søger ikke er til at finde i Pakin (2003), men faktisk kan laves ved at kombinere andre symboler, så kan man tage et kig bagest i Pakin (2003) hvor der bliver forklaret lidt om hvordan man kan sætte symboler sammen. Dette kan f.eks. anvendes til at lave et uafhængighedssymbol (ofte en streg med to lodrette streger).

Eksempel 3.17



```
\DeclareMathOperator*\rmodels{\perp\mskip-11mu\perp}
```

slut preamble

```
$A \rmodels B$
```

```
\[A \rmodels_{n=1}^k B_i \]
```

 $A \perp B$

$$A \underset{n=1}{\overset{k}{\perp}} B_i$$

Vi anvender her `\DeclareMathOperator*` for at få den til at opføre sig som `\sum` mht. grænserne.

Sidebemærkning 3.11. Det er nok de færreste som har anvendt » \Im « eller » \Re « som symboler. Man ville nok hellere anvende dem som imaginær- (\Im) og realdel (\Re) for komplekse tal. Se Eksempel 3.34 på side 61 for et eksempel på hvordan man kan opnå dette.

Sidebemærkning 3.12. Når man skal angive den afledte af en funktion anvender man normalt bare apostrof fra tastaturet i stedet for at anvende `\prime`.

Eksempel 3.18

```
$f'(x)$ $h'\prime(x)$.
```

 $f'(x) h'(x).$

Faktisk er apostroffen i matematik-mode omdøbt til at betyde » \prime «.

Pile

Se også Pakin (2003), tabel 68–73.

Et par pile som mangler fra fontene er negerede versioner af `\uparrow` (\Uparrow) og `\downarrow` (\Downarrow). De findes normalt ikke i symbolfontene.⁴ Dem har vi så lavet i `mathtools`-pakken ved at snyde lidt.

Eksempel 3.19



```
\usepackage{mathtools}
```

slut preamble

```
$ a \uparrow b \downarrow c $
```

 $a \Uparrow b \Downarrow c$

Skribe over eller under en pil

Man får tit brug for at kunne skrive noget ovenover eller under en horisontal pil. Til dette anvendes

```
\xrightarrow{\langle under \rangle}{\langle over \rangle}
```

```
\xleftarrow{\langle under \rangle}{\langle over \rangle}
```

Med den simple anvendelse.

⁴ Dette er ikke længere korrekt, thi pakken `MnSymbol` har faktisk disse med, desværre passer denne symbolsamling ikke ret godt sammen med andre tekstfonte end Minion.

Operatorsymboler

Tabel 3.7: Nogle forskellige pile-symboler

<code>\to</code>	\rightarrow	<code>\leftarrow</code>	\leftarrow	<code>\leftrightarrow</code>	\leftrightarrow
<code>\uparrow</code>	\uparrow	<code>\downarrow</code>	\downarrow	<code>\nearrow</code>	\nearrow
<code>\searrow</code>	\searrow	<code>\swarrow</code>	\swarrow	<code>\nwarrow</code>	\nwarrow
<code>\longrightarrow</code>	\longrightarrow	<code>\longleftarrow</code>	\longleftarrow	<code>\mapsto</code>	\mapsto
<code>\longmapsto</code>	\longmapsto	<code>\longleftrightarrow</code>	\longleftrightarrow	<code>\leadsto</code>	\leadsto
<code>\Rightarrow</code>	\Rightarrow	<code>\Leftarrow</code>	\Leftarrow	<code>\Leftrightarrow</code>	\Leftrightarrow
<code>\iff</code>	\Leftrightarrow	<code>\Downarrow</code>	\Downarrow	<code>\Uparrow</code>	\Uparrow
<code>\Updownarrow</code>	\Updownarrow	<code>\hookrightarrow</code>	\hookrightarrow	<code>\hookleftarrow</code>	\hookleftarrow
<code>\rightarrow</code>	\rightarrow	<code>\circlearrowleft</code>	\circlearrowleft	<code>\circlearrowright</code>	\circlearrowright
<code>\curvearrowleft</code>	\curvearrowleft	<code>\curvearrowright</code>	\curvearrowright	<code>\dashrightarrow</code>	\dashrightarrow
<code>\dashleftarrow</code>	\dashleftarrow	<code>\leftrightharpoons</code>	\leftrightharpoons	<code>\rightleftarrows</code>	\rightleftarrows
<code>\twoheadleftarrow</code>	\twoheadleftarrow	<code>\twoheadrightarrow</code>	\twoheadrightarrow		
<code>\nLeftarrow</code>	\nLeftarrow	<code>\nrightarrow</code>	\nrightarrow	<code>\nLeftrightarrow</code>	\nLeftrightarrow
<code>\nleftarrow</code>	\nleftarrow	<code>\nletrightarrow</code>	\nletrightarrow	<code>\nrightarrow</code>	\nrightarrow
<code>\leftrightharpoons</code>	\leftrightharpoons	<code>\rightleftharpoons</code>	\rightleftharpoons	<code>\leftharpoondown</code>	\leftharpoondown
<code>\leftharpoonup</code>	\leftharpoonup	<code>\rightharpoondown</code>	\rightharpoondown	<code>\rightharpoonup</code>	\rightharpoonup
Ekstra pile via stmaryrd pakken					
<code>\mapsfrom</code>	\mapsfrom	<code>\longmapsfrom</code>	\longmapsfrom	<code>\leftrightarrows</code>	\leftrightarrows
<code>\nrightarrow</code>	\nrightarrow	<code>\nnwarrow</code>	\nnwarrow	<code>\ssearrow</code>	\ssearrow
<code>\sswarrow</code>	\sswarrow	<code>\lightning</code>	\lightning		

```
\begin{align*}
& & X & \xrightarrow{f(x)} & Y \\
& & X & \rightarrow & Y \\
& & & f & \\
& & X & \xrightarrow[f]{f(x)} & Y \\
\end{align*}
```

$$\begin{array}{ccc}
 X & \xrightarrow{f(x)} & Y \\
 X & \rightarrow & Y \\
 & f & \\
 X & \xrightarrow[f]{f(x)} & Y
 \end{array}$$

Eksempel 3.20

Pakken `mathtools` giver desuden flere lignede strækbare pile-konstruktioner, se [Høgholm \(2008b\)](#).

Sidebemærkning 3.13. Mange preprint-forfattere kender ikke ovenstående konstruktion og forsøger sig derfor med `\overset` eller `\stackrel`, dette kan ikke betale sig og bliver nemt grimt. Anvend `\rightarrow` eller lignende i stedet.

3.2.2 Operatorsymboler

Et operatorsymbol eller et »stort« symbol er betegnelsen for symboler så som \sum , \cap , o.a. som findes i en fremhævet variant og en (mindre) tekstvariant. Disse vil ofte desuden placere eventuelle grænser over/under symbolet i den fremhævede form samt ved siden af symbolet i tekstvarianten.

Skulle man have brug for at kunne tvinge et stort symbol til at placere grænser enten over/under eller på siden, kan man anvende `\limits`-makroen eller den tilsvarende `\nolimits`.

`\limits`
`\nolimits`

3. Introduktion til matematik

Eksempel 3.21

```
\begin{equation*}
\int\limits_{\Omega} f(\omega)\wedge g(\omega)\,d\omega
\sum\limits_n^k f_k
\end{equation*}
```

$$\int_{\Omega} f(\omega) \wedge g(\omega) d\omega \sum_n^k f_k$$

Nogle af de mest almindelige store symboler kan ses i Tabel 3.8 (i deres fremhævede varianter).

Tabel 3.8: Nogle store symboler

<code>\sum</code>	Σ	<code>\prod</code>	\prod	<code>\coprod</code>	\coprod
<code>\bigvee</code>	\bigvee	<code>\bigwedge</code>	\bigwedge	<code>\bigsqcup</code>	\bigsqcup
<code>\bigsqcap</code>	\bigsqcap	<code>\bigcap</code>	\bigcap	<code>\bigcup</code>	\bigcup
<code>\bigotimes</code>	\bigotimes	<code>\bigoplus</code>	\bigoplus	<code>\int</code>	\int
<code>\iint</code>	\iint	<code>\iiint</code>	\iiint	<code>\iiiint</code>	\iiiint
<code>\iiiiint</code>	\iiiiint	<code>\idotsint</code>	\idotsint	<code>\oint</code>	\oint
Via stmaryrd-pakken					
<code>\bigbox</code>	\bigbox	<code>\biginterleave</code>	\biginterleave	<code>\bigparallel</code>	\bigparallel
Via esint-pakken eller kpfonts-pakken					
<code>\ointclockwise</code>	\ointclockwise	<code>\ointctrlockwise</code>	\ointctrlockwise	<code>\sqint</code>	\sqint
<code>\sqint</code>	\sqint	<code>\oint</code>	\oint	<code>\oiint</code>	\oiint

Man kan også finde andre i Pakin (2003) kapitel 3, bemærk at visse symboler kræver at man har installeret ekstra fonte. Her følger et simpelt eksempel.

Eksempel 3.22

```
\begin{equation*}
\sum_n \bigwedge_{k \geq n}
\biggl( \bigotimes_j \xi_{jkn} \biggr)
\end{equation*}
```

$$\sum_n \bigwedge_{k \geq n} \left(\bigotimes_j \xi_{jkn} \right)$$

Et symbol som mangler fra tabel 3.8 er en stor udgave af \times , altså en `\bigtimes`. Pakin (2003) nævner godt nok nogle fonte hvori man kan finde den, men man kan lige så godt lave den selv. Vi har tilføjet konstruktionen til mathtools (special tak til Enrico Gregorio). Hvilket så kommer til at se ud som følger.

Eksempel 3.23

```
\[ \bigtimes_{i=1}^n [a_i, b_i] +
\frac{\bigtimes_{i=1}^n A_i}{\Omega} .
\]
```

$$\bigtimes_{i=1}^n [a_i, b_i] + \frac{\bigtimes_{i=1}^n A_i}{\Omega} .$$

Tip 3.14 (Giv egne navne til specifikke symboler/konstruktioner).

Der er naturligvis ingen som kan huske alle disse symboler udenad, så dem slår man op når man har brug for det. Omvendt så har mange af symbolerne nogle underlige navne som måske ikke lige er relevant indenfor det felt man selv arbejder med. F.eks. anvendes symbolet `\simeq` (\simeq) til at betyde homotopi-ækvivalens indenfor topologi, og man anvender `\cong` (\cong) til at angive isomorfier. Dette er ikke videre nemt at forstå udfra kildekoden. Det kan derfor være en god ide at man selv laver nogle andre mere sigende navne for visse makroer eller konstruktioner.

Se Appendix A på side 351 for en gennemgang af hvordan man laver makroer.

Nogle eksempler på makroer af denne type, kan vi hente fra statistik. Her skriver man f.eks. $X \sim N(0,1)$ for at angive at den stokastiske variabel X er standard normalfordelt. Men det er lidt svært at se udfra `X\sim N(0,1)`, en bedre ide ville være at lave makroen `\DistAs` og så skrive `X\DistAs N(0,1)`, så får man i det mindste en ide om hvad der står.⁵ Det er en ret god ide at bruge navne som i sig selv giver mening, således at kildekoden faktisk begynder at give mening uden at man skal se slutproduktet. Flere eksempler fra statistik

```
\newcommand\DistAs{\sim}
\newcommand\DistEq{%
\stackrel{\scriptscriptstyle\smash{d}}{=}}
\l X\DistAs N(0,1), \quad
Y \DistEq Z . \l
```

$$X \sim N(0,1), \quad Y \stackrel{d}{=} Z.$$

Eksempel
3.24

Vi anvender `\smash` for at `\DistEq` ikke skal forstyrre linieafstanden hvis den anvendes i den almindelige tekst. Makroen `\stackrel` forklares på side 64 og `\scriptscriptstyle` er en måde hvorpå man kan påtvinge stilen/størrelsen fra dobbelt subscript.

3.2.3 Matematiske alfabetfonte

Med matematiske alfabetfonte menes her fontvarianter lige som *kursiv* bare i stedet for til brug i matematik. En oversigt kan findes i tabel 3.9 på næste side.

Se også Pakin (2003) tabel 135. Forskellen mellem den almindelige matematik mode (som jo er kursiv) og `\mathit` er afstanden mellem bogstaverne, se Eksempel 3.25.

`\$differ\$` vs. `\$ \mathit{differ} \$`

differ vs. *differ*

Eksempel
3.25

Man anvender ikke `\mathit` særligt ofte. Der hvor den kan anvendes er når man har én variabel som består af flere bogstaver, en sådan ville man traditionelt skrive upright, altså SDD i stedet for SDD , men insisterer man på kursiv, så giver `\mathit` et bedre resultat SDD .

Skrive almindelig tekst indeni i et matematisk udtryk

I matematik-mode ignoreres alle mellemrum og man kan ikke anvende f.eks. æ, ø eller å. Så hvordan tilføjer man noget tekst til en formel? Det ses af Tabel 3.9 på næste side at den anbefalede metode til indsættelse af tekst i matematik er makroen

`\text{...}`

`\text`

3. Introduktion til matematik

Tabel 3.9: Matematiske alfabetfonte

<code>\mathrm{ABCdef123}</code>	ABCdef123	
<code>\mathit{ABCdef123}</code>	<i>ABCdef123</i>	
<code>\mathsf{ABCdef123}</code>	ABCdef123	
<code>\mathbf{ABCefg\Gamma1}</code>	ABCefgΓ1	se den efterfølgende bemærkning
<code>\mathfrak{ABCdef123}</code>	℔℔℔℔ef123	
<code>\mathcal{ABC}</code>	<i>ABC</i>	kun store bogstaver
<code>\mathbb{ABC}</code>	ABC	kun store bogstaver, kommer via <code>amssymb</code> , men andre pakker kan ændre udseendet af disse, f.eks. <code>mathpazo</code> .
<code>\mathscr{ABC}</code>	<i>A B C</i>	via pakken <code>mathrsfs</code> , kun store bogstaver
<code>\mathbbm{ABC12}</code>	ABC12	via pakken <code>bbm</code> , kun store bogstaver samt 1 og 2, bemærk at denne font kan være fuzzy i PDF preview
<code>\mathds{ABC1}</code>	ABC1	via pakken <code>dsfont</code> , kun store bogstaver samt 1
<code>\bm{ABcd1\Gamma\alpha\Sum}</code>	ABcd1ΓαΣ	via pakken <code>bm</code> , anbefalet til fed matematik
<code>\text{almindelig tekst}</code>	almindelig tekst	fra <code>amstext</code> (loades automatisk af <code>amsmath</code>)

Inden i denne er man igen tilbage til almindelig tekst. Man bør lige bemærke at hvis argumentet til `\text` starter med et mellemrum, vil dette mellemrum faktisk blive anvendt (hvilket ofte er en fordel), se det følgende eksempel

Eksempel 3.26

```
\[ x^2 \geq 0 \text{ for alle } x \in \mathbb{R} \]. \]
```

$$x^2 \geq 0 \text{ for alle } x \in \mathbb{R}.$$

Bemærk hvordan vi i dette tilfælde skriver *helle* kommentaren inden i `\text{\{...\}}`, på denne måde er der en form for adskillelse mellem matematikken ($x^2 \geq 0$) og kommentaren (for alle $x \in \mathbb{R}$).

Personligt skriver jeg normalt [Eksempel 3.26](#) som

Eksempel 3.27

```
\[ x^2 \geq 0 \quad \text{for alle } x \in \mathbb{R} \]. \]
```

$$x^2 \geq 0 \quad \text{for alle } x \in \mathbb{R}.$$

for at få et lidt større mellemrum. Jeg anvender så `\quad` eller `\quad\quad` hele vejen gennem dokumentet for at få et mere ensartet udtryk.



Anvender man ord som indices til variable, bør man generelt undgå enhver brug af `\text` kommandoen. Man bør i stedet eksplicit angive at dette ord skal skrives *opretstående*.

Hvorfor denne advarsel? Fordi `\text{\{tekst\}}` holder øje med den font man anvender uden for matematikken, og tilpasser sig så efter den. Dette er illustreret i det følgende eksempel.

Danske vokaler i matematik

```
tekst \textit{tekst $xyz_
\text{tekst}$ tekst
```

tekst *tekst* xyz_{tekst} *tekst* tekst

Eksempel
3.28

Det anbefales i stedet at man sørger for at alle ord-indices skal være opretstående og i den almindelige tekstfont, i.e. via `\textup{⟨tekst⟩}`.

`\textup`

```
$R_{\textup{sø}}=25\,\textup{km}\quad
r_{\textup{min}}=2$
```

$R_{sø} = 25 \text{ km}$ $r_{\min} = 2$

Eksempel
3.29

Danske vokaler i matematik

Der er jævnligt brugere som klager over at de ikke kan skrive de danske vokaler i matematik. Kigger man på det de ønsker at skrive så er det oftest enten kommentarer, se kommandoen `\text` eller det er vedr. indices, f.eks. radius af en sø: $R_{sø}$.

Løsningen på dette er forklaret i det forrige afsnit (»*Skrive almindelig tekst indeni i et matematisk udtryk*«).

Nogle lettere bizarre mennesker anvender de danske vokaler som funktions- eller mængdenavne i matematik, så man kan godt hvis man selv ved hvordan man skal rode med tingene, hvilket vi ikke kommer nærmere ind på.

Se desuden afsnit 10.4 på side 268.

Fed skrift i matematik

Makroen `\mathbf` sætter godt nok fonten i fed men de er nu *opretstående* i stedet for den normale kursive. Desuden virker den kun på almindelige bogstaver, tal samt de store græske bogstaver. Løsningen på dette problem hedder `\bm` fra pakken `bm`, den virker på alle symboler, bogstaver, tal osv. Hvis et symbol ikke findes i en fed version, laver `\bm` selv en simuleret fed ved at tegne symbolet nogle gange lettere forskudt (metoden kaldes også *poor mans bold*).

`\mathbf`

`\bm`

```
\usepackage{amssymb,bm}
```

— slut preamble —

```

\mathbf{ABcd12\Lambda\Gamma\alpha\beta\oplus\sum}

```

```

\bm{ABcd12\Lambda\Gamma\alpha\beta\oplus\sum}

```

ABcd12ΛΓαβ⊕Σ
ABcd12ΛΓαβ⊕Σ

Eksempel
3.30



Forstørret man sumtegnet kan man godt se at den er fede udgave er simuleret.

Et interessant trick er også at man kan skifte al matematik til at være fed. Kan f.eks. være relevant at bygge ind i konfigureringen af en afsnitsoverskrift, således at teksten der har fed matematik, men at samme matematik ikke bliver fed i sidehovedet eller indholdsfortegnelsen. Kommandoen til denne aktivering hedder `\boldmath`.

```
$x^2$ \boldmath $x^2$
```

x^2 **x^2**

Eksempel
3.31

Makroen kan kun anvendes uden for matematik.

3.2.4 Navngivne funktioner og operatorer

Inden for matematikken har man en masse operatorer/funktion som er angives via navn, eksempelvis »sin« (sinus), »lim« (grænseovergang), »log« (logaritme) samt mange andre. \LaTeX stiller en del af disse til rådighed og `amsmath` giver os redskaber til selv at lave dem. Men først en formaning: Der er en grund til at man anvender f.eks. `\sin` i stedet for bare at skrive »sin«.

3. Introduktion til matematik

Eksempel 3.32

```
\begin{equation*}
A \sin B \quad A \operatorname{\sin} B
\quad A \sin B
\end{equation*}
```

$$A \sin B \quad A \operatorname{\sin} B \quad A \sin B$$

Eksempel 3.32 viser at det første eksempel er i hvert fald helt forkert, man kan ikke se meningsmæssig forskel på dette samt $A \cdot s \cdot i \cdot n \cdot B$, hvilket er grunden til at man *altid* skriver disse operatornavne med en opretstående font. Men det andet eksempel viser at den naive metode heller ikke dur – der mangler mellemrum, og det er så det man i et tredje punkt kan se at L^AT_EX har tilføjet via `\sin`. Med dette mellemrum behøver man heller ikke altid at anvende parenteser omkring argumentet til sinus.

Tabel 3.10: Prædefinerede navngivne operatører i L^AT_EX

<code>\cos</code>	cos	<code>\sin</code>	sin	<code>\tan</code>	tan	<code>\cot</code>	cot
<code>\arccos</code>	arccos	<code>\arcsin</code>	arcsin	<code>\arctan</code>	arctan	<code>\csc</code>	csc
<code>\cosh</code>	cosh	<code>\sinh</code>	sinh	<code>\tanh</code>	tanh	<code>\coth</code>	coth
<code>\deg</code>	deg	<code>\arg</code>	arg	<code>\dim</code>	dim	<code>\ker</code>	ker
<code>\log</code>	log	<code>\ln</code>	ln	<code>\lg</code>	lg	<code>\exp</code>	exp
<code>\hom</code>	hom	<code>\lim</code>	lim	<code>\sup</code>	sup	<code>\inf</code>	inf
<code>\liminf</code>	liminf	<code>\limsup</code>	limsup	<code>\max</code>	max	<code>\min</code>	min
<code>\sec</code>	sec	<code>\Pr</code>	Pr				

Visse af funktionsnavnene i Tabel 3.10 placerer sup- eller superscript under/over navnet når vi er i fremhævede formler, dette gælder f.eks. `\lim`, `\max`, `\min`, `\sup`, `\inf`.

Før nogen begynder at brokke sig over at deres yndlings operator ikke er med på listen kan vi berolige med at der er meget nemt selv at lave disse. Jævnfør diskussionen efterfølgende **Eksempel 3.32** er det følgende desuden den anbefalede måde at gøre lave navngivne funktioner på.

```
\DeclareMathOperator{\macronavn}{\langle tekst \rangle}
\DeclareMathOperator*{\macronavn}{\langle tekst \rangle}
```

Bemærk, at de kan kun anvendes i preamblen. Makroen `\DeclareMathOperator*` giver en operator som i fremhævede formler placerer sub- eller superscript under/over navnet. Herunder følger et eksempel som genererer sporet (`\tr`) samt limes exterior (`\limext`).

Eksempel 3.33



```
\DeclareMathOperator{\tr}{tr}
\DeclareMathOperator*{\limext}{\lim\,ext}
----- slut preamble -----
\l \limext_{x \in A} \tr T(x)=42 \l
```

$$\lim_{x \in A} \operatorname{tr} T(x) = 42$$

Man skal være opmærksom på at visse navne allerede er optaget af L^AT_EX, så man kan blive nødt til at finde på andre navne.

Er man *helt sikker* på at man ved, hvad et allerede eksisterende makronavn gør og man ved, man ikke får brug for det, kan man godt *genbruge* et makro navn. Et eksempel kan være makroerne `\Re` (\Re) og `\Im` (\Im), de fleste ville jo anvende dette til real- og imaginærdelen af komplekse tal. Her kan `\operatorname` gøre gavn, idet der findes *ingen* `\reDeclareMathOperator`-makro.

`\Re`
`\Im`
`\operatorname`

Navngivne mængder

```
\renewcommand{\Re}{\operatorname{Re}}
\renewcommand{\Im}{\operatorname{Im}}
```

— slut preamble —

```
Lad  $z=a+ib$  da er  $\Re\{z\}=a$  og  $\Im\{z\}=b$ .
```

Lad $z = a + ib$ da er $\operatorname{Re} z = a$ og $\operatorname{Im} z = b$.

Eksempel
3.34



Se desuden appendix A på side 351, vedrørende yderligere information om hvordan man laver sine egne simple makroer.



Advarsel 3.15 (Overskriv kun hvis du ved hvad originalen gør).

Et af de navne som er optaget er `\span`-makroen. Det er *meget vigtigt* at man *ikke* overskriver denne. Det er en meget vigtig byggesten i \TeX , og den anvendes internt i tabeller og visse matematikkonstruktioner.

3.2.5 Navngivne mængder

6

Det anbefales generelt at man sørger for at lave makroer til at angive alle de specielle mængder man anvender i sit dokument, f.eks. til de almindelige talmængder. Det er dog ikke altid nogen god ide at anvende meget korte navne til sine mængder, især ikke hvis det man skriver skal flettes sammen med noget andre har skrevet (thi det kunne være en anden forfatter har anvendt samme makro til at betyde noget helt andet).

I de seneste projekter har jeg f.eks. anvendt `\setZ` som et alias for `\mathbb{Z}` (de hele tal), og ikke `\Z` som mange måske ville anvende.

Er dokumentet bare til en selv, så kan man sagtens anvende meget korte navne, men er det noget som senere skal anvendes af andre så bør man overveje en bedre navngivning.

Pt. har vi ikke noget forslag til en god navngivning. Forslag modtages gerne.



Ud over navngivningen af makroer, så er der også en anden ting man skal vurdere vedr. sine mængder: »Luften omkring navnet«. Normalt behøver man ikke ekstra luft (som ved `\sin`), dvs. skal man f.eks. definere mængden `GL`, så kan man bare anvende

```
\newcommand\GL{\mathrm{GL}}
```

Eksempel
3.35

Men hvis makroen har flere betydninger, så bør man evt. anvende en anden luft omkring, eller anvende to forskellige makroer, så man har en til hver af de to betydninger. Et eksempel på dette er hvis et navn både er en mængde samt en operator, så kan man overveje at gå over `\DeclareMathOperator` vejen, således at man kan gange ting på operatoren, og stadigvæk se forskel på de enkelte dele.

3.2.6 ... - prikker er ikke bare prikker

Som det også er tilfældet i almindelig tekst skriver man *ikke* ... via ... (tre punktummer) i matematik. Den vigtigste makro til at lave »...« med er

```
\dots
```

```
\dots (...)
```

som har den fordel at den indenfor matematik-mode er *semi-intelligent*, på den måde at alt efter hvad der står *efter* »\dots«, justerer den sig selv så den står frit svævende eller solidt plantet på basislinien.

⁶ Fixme Note: skrives bedre, når der bliver tid

3. Introduktion til matematik

Eksempel 3.36

`$1,\dots,5$`, `$1<2<\dots<7$`,
`$x_1=x_2=\dots=x_n$`, `$1+2+\dots+6$`

$1, \dots, 5$, $1 < 2 < \dots < 7$, $x_1 = x_2 = \dots = x_n$,
 $1 + 2 + \dots + 6$

Så i en opremsning indsætter man bare `\dots` og lader den gøre arbejdet. Der er en enkelt undtagelse vedrørende produkter, her anvendes i stedet følgende eksempel:⁷

Eksempel 3.37

`$x_1\cdots x_n$` eller
`$x_1\cdot\ldots\cdot x_n$`.

$x_1 \cdots x_n$ eller $x_1 \cdot \dots \cdot x_n$.



`\cdots` (⋯)

Når prikkerne kommer til slut har `\dots` ikke noget at kigge efter makroen (en uendelig fortsættelse), den vil derfor altid stå på baseline. Her må man så selv sørge for at indsætte `\cdots` når det er nødvendigt. Den generelle regel er at der skal anvendes `\cdots` i næsten alle tilfælde undtagen når prikkerne afslutter en komma-separeret opremsning,⁸ dvs.

Eksempel 3.38

`x_1,x_2,\dots` og `\emph{ikke}`
`x_1,x_2,\cdots`!

x_1, x_2, \dots og *ikke* x_1, x_2, \dots !

Downes (2002) afsnit 4.6. nævner desuden visse andre dot-konstruktioner til anvendelse i slutningen af en uendelige opremsning. Tabel 3.11 er en oversigt over almindelige dots-konstruktioner i \LaTeX .

Tabel 3.11: Dots i \LaTeX

<code>\dots</code>	$x + \dots + y$	beskrevet ovenfor
<code>\cdot</code>	$x \cdot y$	en enkelt centreret prik
<code>\cdots</code>	$X_1 \cdots X_n$	flere centrerede prikker
<code>\ddots</code>	\ddots	diagonale prikker
<code>\vdots</code>	\vdots	vertikale prikker
<code>\cdot</code>	$x \cdot y$	en enkelt prik på baseline, som et punktum bare med en anden spacing, sammenlign med $x \cdot y$
<code>\ldots^a</code>	x, \dots, z	prikker på basislinien, kan også anvendes i den almindelige tekst
Kræver mathdots-pakken		
<code>\iddots^b</code>	$\cdot\cdot$	

NOTE: Se også Downes (2002) afsnit 4.6.

^a Da mange er kommet galt af sted ved at brug af `\ldots` anbefales det at man holder sig til `\dots` og kun bruger `\ldots` i specialtilfælde.

^b Pakken `mathdots` forbedrer også `\ddots` og `\vdots`.

3.2.7 Matematiske accenter

Det er vigtig at huske at accenterne man anvender i den normale tekst er ikke de samme som man anvender i matematik – faktisk er de ikke tilladt i matematik, man skal derfor anvende helt andre konstruktioner når det gælder matematik, her har accenterne normalt også andre fortolkninger. En oversigt kan ses i Tabel 3.12 på næste side.

⁷ Se f.eks. Chicago (2003).

⁸ Se Chicago (2003) eller Swanson (1999).

Tabel 3.12: Matematiske accenter

Kode	Eksempel
<code>\hat{\langle txt \rangle}</code>	\hat{a}
<code>\widehat{\langle txt \rangle}</code> ^a	\widehat{XXX}
<code>\tilde{\langle txt \rangle}</code>	\tilde{g}
<code>\widetilde{\langle txt \rangle}</code> ^a	\widetilde{YYY}
<code>\bar{\langle txt \rangle}</code>	\bar{b}
<code>\overline{\langle txt \rangle}</code>	$\overline{x+y}$
<code>\underline{\langle txt \rangle}</code> ^b	\underline{A}
<code>\dot{\langle txt \rangle}</code> ^c	\dot{x}
<code>\vec{\langle txt \rangle}</code>	\vec{x}
<code>\overrightarrow{\langle txt \rangle}</code> ^d	\overrightarrow{abc}
<code>\underrightarrow{\langle txt \rangle}</code> ^e	\underrightarrow{xyz}
<code>\overleftarrow{\langle txt \rangle}</code> ^f	\overleftarrow{xyz}
<code>\acute{\langle txt \rangle}</code>	\acute{a}
<code>\grave{\langle txt \rangle}</code>	\grave{a}
<code>\breve{\langle txt \rangle}</code>	\breve{a}
<code>\check{\langle txt \rangle}</code>	\check{a}
<code>\mathring{\langle txt \rangle}</code> ^g	\mathring{a}

^a Strækker sig til en vis grad efter teksten. Mængden af »strækbarhed« afhænger af fonten, men den går normalt ca. ved bredden af XXXXXX. Visse kommercielle fonte ser ud til at kunne klare mere.

Nogle anvender disse tegn til at markere en eller anden operation, dette giver problemer når det som der skal opereres på bliver meget bredt. Her bliver man enten nødt til f.eks. at skrive $(F(X)+G(y))^\sim$ eller endnu bedre, sørge for at forklare i teksten, at dette kan skrives på to måder, og så f.eks. skrive $\text{Navn}(F(A)+G(X))$ i stedet for at forsøge at skrive en tilde over. (Det er jo også det man gør med e^x og $\exp(x)$.)

Desuden vil de lange tilder eller hatte, forstyrre linieafstanden, hvis de anvendes uden for den fremhævede matematik.

^b Denne strækker sig efter teksten og kan desuden anvendes i den almindelige tekst.

^c Findes også i `\ddot`, `\ddd` og `\ddddot` varianter.

^d Der findes naturligvis også en `\leftoverarrow` med samme syntaks. Strækker sig efter bredden af teksten.

^e Tilsvarende for `\underleftarrow`.

^f Der findes også en `\underleftarrow`.

^g Denne skal *ikke* anvendes til at lave vokalen »å« i matematik mode.

Bemærk at det ikke er noget umiddelbart problem at lægge accenter på accenter, men husk at de små accenter centrerer over symbolet, hvilket kan se lettere underligt ud, e.g. $\tilde{\underline{A}}$ (`\tilde{\underline{A}}`). Problemet kan løses ved at ændre rækkefølgen, så man laver tilden først $\underline{\tilde{A}}$.

```
\begin{equation*}
\hat{\bar{x}}+\tilde{\underline{B}}=2
\end{equation*}
```

$$\hat{\bar{x}} + \tilde{\underline{B}} = 2$$

Eksempel
3.39

3. Introduktion til matematik

Sidebemærkning 3.16. Man bør også huske at passe lidt på når man vil sætte en accent på en variabel med index. Det ses tit at folk skriver » \hat{x}_j « hvor man egentlig mente » \hat{x}_j «.

Sidebemærkning 3.17. Man bedes desuden bemærke problemet mellem accenter og dobbelt sub- eller superscript, se Bemærkning 3.33 på side 69.

Sidebemærkning 3.18. Før man begynder f.eks. at tænke på at placere en accent ovenpå e.g. $e =$ bør man lige tage et kig i symbollisten, evt. i Pakin (2003).

Til tider kan det være en fordel at kunne f.eks. placere (d) oven på $e =$ for at symbolisere en speciel type lighed. Dette kan gøres via

`\stackrel` `\stackrel{\langle over \rangle}{\langle symbol \rangle}`

Selve $\langle over \rangle$ -delen bliver man ofte nødt til at pille ved, thi den nemt kan blive breddere end $\langle symbol \rangle$ -delen og derfor ødelæge justeringen i formlen. Kan fikses via `\mathclap` fra `mathtools`-pakken.

Eksempel 3.40



```
\usepackage{mathtools}
----- slut preamble -----
\l X \stackrel{\mathclap{
\scriptscriptstyle\smash{(d)}}}{=}
Y \l
```

$$X \stackrel{(d)}{=} Y$$

Generelt bør man ikke skrive noget som er alt for bredt over et sådant symbol. Før man begynder at lave nye symboler på denne måde bør man lige tjekke symbolisterne om der ikke er noget man kan anvende. Desuden bør man naturligvis definere sig en makro til at give sig dette nye symbol, således man ikke skriver denne smøre hver gang.

Sidebemærkning 3.19. Det er en god ide a kombinere definitionen med kommandoen `\smash`, således at højden af symbolet ikke påvirker linieafstanden. Dette kan dog give overlappingsproblemer hvis det som står på linien lige over rager lidt langt ned.

Anvend *ikke* `\stackrel` til at skrive over en almindelig pil, se i stedet afsnit 3.2.1 på side 54.

Sidebemærkning 3.20. Man ser til tider folk der skriver $\stackrel{(*)}{=}$ for at kunne give en kommentar til denne lighed senere. Dette anses for at være dårlig stil på skrift, og bør derfor kun anvendes ved en tavlegennemgang.

Skriv i stedet teksten om, gerne med forklarende tekst mellem linierne i en udregning, se *Kommentarer i alignment matematik* på side 93.

Hegn

Tip 3.21 (Linie over variabel).

Man vil nok opdage at \bar{V} (`\bar{V}`) ikke er videre pænt, og at \overline{V} (`\overline{V}`) forekommer lidt lang (tykkelsen kan vi ikke helt gøre noget ved). Via lidt magi og `mathtools`-pakken, kan vi lave en lidt bedre marko

```
% kræver mathtools
\DeclareRobustCommand\altoverline[2][0.8]{%
\mathmakebox[\widthof{#2}][c]{\overline{\mathmakebox[\widthof{#2}*\real{#1}][c]{#2}}}}
```

Dette giver så \overline{V} i stedet, hvilket er lidt bedre. Man kan gøre stregen kortere via

```
$\altoverline[0.65]{XYZ}$
```

$$\overline{XYZ}$$

Kodetip

3.1



Eksempel

3.41

Tip 3.22 (Linie under variabel).

Pakken `ushort` kan hjælpe hvis man synes at `\underline` gør det dårligt.

3.3 Hegn

Definition 3.23. Et *hegn* (eller *delimiter* eller *afgrænser*) er et symbol eller oftere et par af symboler med den egenskab at de fås i forskellige (indbyggede) størrelser samt kan strækkes hvis man har noget meget højt (eller dybt).

hegn

Et simpelt eksempel er naturligvis de almindelige parenteser `»()«` eller `»< >«`, men `»/«` er også et hegn (som kan skaleres). Se tabel 3.13.

Tabel 3.13: Indhegnings- og afgrænsersymboler

<code>()</code>	<code>()</code>	<code>[]</code>	<code>[]</code>	<code>\{\}</code>	<code>\{ }</code>
<code>\lceil \rceil</code>	<code>[]</code>	<code>\lfloor \rfloor</code>	<code>[]</code>	<code>\lgroup* \rgroup*</code>	<code>()</code>
<code>\langle \rangle</code>	<code>\langle \rangle</code>	<code>\llbracket[†]</code>	<code>\llbracket</code>	<code>\lvert \rvert</code>	<code> </code>
		<code>\rrbracket[†]</code>	<code>\llbracket</code>		
<code>\lVert \rVert</code>	<code> </code>				
<code>/</code>	<code>/</code>	<code>\backslash</code>	<code>\</code>	<code>\vert</code>	<code> </code>
<code>\Vert</code>	<code> </code>	<code>.</code> [‡]			

^{*} virker kun forstørret, i.e. kun hvis man sætter f.eks. `\bigl` eller `\bigr` foran.

[†] kræver pakken `stmaryrd`

[‡] tomt hegn, se afsnit *Om \left og \right* side 67.



Formaning 3.24 (Anvend den rigtige type hegn).

I stedet for `\vert` kan man også anvende `»|«`-tegnet og i stedet for `\Vert` kan man anvende `\|`. Men anvend dem (`»\vert«`, `»\Vert«`, `»|«` og `»\|«`) med varsomhed, thi

Eksempel 3.42

```
$|-x| \neq \vert -x\vert$ og
$\|-x\| \neq \lVert -x\rVert$
```

$$|-x| \neq |-x| \text{ og } \lVert -x \rVert \neq \lVert -x \rVert$$

giver forkert spacing.

Man kan så manuelt skalere sit hegn ved at foranstille en af makroerne

```
\bigl( ( \Bigl( ( \biggl( ( \Biggl( ( \Biggr) ) \biggr) ) \Bigl) ) \bigl) )
```

`\left, \right` beskrevet på næste side

`\bigm... \Biggm` beskrevet på side 68

Sidebemærkning 3.25. Der findes også `\big`-varianter uden `l` eller `r` til slut, disse kan også bruges, men det er en god ide lige at læse teksten side 410 for et eksempel på hvorfor vi har valgt at koncentrere os om `l`- og `r`-varianterne.

`\big`-makroerne (samt `\left` og `\right`) kan *kun* anvendes *foran* tegn eller symbolnavne som kvalificerer som hegn. F.eks. er symbolet `\mid` *ikke* en afgrænser, selv om man skulle tro det, og faktisk anvendes `\mid` ofte som mængdebygger og man bliver så nød til at anvende `\bigm\vert` for at få symbolet i den næste størrelse.

Eksempel 3.43

```
Lad $A=\{x\mid x\geq 0\}$ og
\l
B=\bigl\{x\in A \bigm|
x<\tfrac{1}{2}\big\r}.
\l
```

Lad $A = \{x \mid x \geq 0\}$ og

$$B = \left\{x \in A \mid x < \frac{1}{2}\right\}.$$

Sidebemærkning 3.26 (Mængdebyggermakro).

Her er en anden mængdebygger baseret på pakken `mathtools`

Eksempel 3.44

```
\usepackage{mathtools}
\DeclarePairedDelimiter\Set[2]%
{\lbrace}{\rbrace}{#1\,\delimsize\vert\, #2}
----- slut preamble -----
Lad $A=\Set{x}{x\geq 0}$ og
\l
B = \Set[\big]{ x\in A }{ x<\tfrac{1}{2} }
\l
```

Lad $A = \{x \mid x \geq 0\}$ og

$$B = \left\{x \in A \mid x < \frac{1}{2}\right\}$$

Sidebemærkning 3.27. Som en bizar bemærkning kan nævnes at symboler som `\downarrow`, `\uparrow`, `\Downarrow`, `\Uparrow`, `\updownarrow` og `\Updownarrow` faktisk også kan skaleres med nævnte konstruktioner.

Sidebemærkning 3.28. Hvilke størrelser skal man så anvende? Som man vil opdage, anbefales det at man undgår at anvende autoskallering (forklaret i afsnittet herefter) over det hele, det går nemt galt. Mit generelle råd er:

sørg for at anvende en størrelse som gør at læseren IKKE ER I TVIVL om hvad det er som INDHEGNES, men således at det heller IKKE ER HEGNET som tager fokus.

Husk i øvrigt det gode råd man ofte ser i vejledninger til journalartikler: Bland hegnene, altså i stedet for at skrive ((((i varierende størrelse), kunne man sagtens anvende {{{ (forudsat {} eller {} ikke har en speciel betydning indenfor det man skriver).

Om \left og \right

\left og \right anvendes *som par*, til at give et hegn som er skaleret efter det som de omslutter.

```
\begin{equation*}
\left(\frac{1}{x^2+1}+2\right)^2
\end{equation*}
```

$$\left(\frac{1}{x^2+1}+2\right)^2$$

Eksempel 3.45

Hvis man har en situation hvor man kun skal bruge den ene afgrænser i et par, skal man bare angive den anden som det tomme hegn ».« (punktum).

```
\begin{equation*}
\left.\sum_{i=1}^{\infty}\right\rangle\Rightarrow
\end{equation*}
```

$$\left.\sum_{i=1}^{\infty}\right\rangle\Rightarrow$$

Eksempel 3.46

Se desuden *cases*-environmentet, beskrevet på side på side 75.

cases



Formaning 3.29. Man skal dog afholde sig fra bare at tænke:

ok, jeg sørger da bare for at autoskalere alle mine hegn!

Det bliver nemlig hurtigt grimt:

```
\begin{equation*}
\left(\sum_{k_j} k_j^2\right) \quad \text{vs.} \quad \quad
\biggl(\sum_{k_j} k_j^2 \biggr).
\end{equation*}
```

$$\left(\sum_{k_j} k_j^2\right) \quad \text{vs.} \quad \biggl(\sum_{k_j} k_j^2 \biggr).$$

Eksempel 3.47

I eksemplet til højre har vi skaleret ()-erne pr. håndkraft således at de ikke længere er så dominerende. Hvilke \bigl/r-konstruktioner som skal anvendes afhænger af situationen. Ofte må man prøve sig frem, med en vis erfaring kan man nemt gætte de bedste størrelser. Til summer med grænser anvendes ofte \biggl/r-konstruktionen.



Formaning 3.30 (Ingen autoskalering udenfor fremhævet matematik).

I den almindelige tekst bør man holde sig fra \left og \right, samt alle hegn større end dem man kan få via \big-konstruktionen. Større hegn kommer nemt til at forstyrre linieafstanden.

3. Introduktion til matematik

Varianten `\middle`

Hvis man anvender en moderne L^AT_EX installation (hvilket de fleste gør), har man ud over `\left` og `\right` også en `\middle`-makro som anvendes til at skalere hegn mellem et `\left... \right`-par.

Eksempel 3.48

```
\[
\left\{ x \in \mathbb{R} \middle| \,
\frac{1}{x} > 5 \right\}
\]
```

$$\left\{ x \in \mathbb{R} \middle| \frac{1}{x} > 5 \right\}$$

i dette eksempel ville det nok være på sin plads manuelt at indsætte noget ekstra luft. Endnu bedre ville det være at lave en makro, og så gemme syntaksen inden i denne makro, se [Eksempel 3.44 på side 66](#).

Om *m*-varianterne af `\big... \Bigg`

Til tider vil man gerne forstørre delimiters som ikke er anvendt som et par, f.eks. i en mængdebygger konstruktion. Til dette anvender man `\bigm`, `\Bigm`, `\biggm` eller `\Biggm`, det specielle ved disse er at de efterlader automatisk lidt ekstra plads på begge sider af symbolet.

Eksempel 3.49

```
\begin{equation*}
\biggl\{ x \biggm\vert
\sum_{n=1}^{\infty} f(x,n) < \infty \biggr\}
\end{equation*}
```

$$\left\{ x \middle| \sum_{n=1}^{\infty} f(x, n) < \infty \right\}$$

Her kan man jo selv prøve at se hvordan det ser ud med en af de andre typer `\big`. Man kan også anvende dem til at give noget mere luft:

Eksempel 3.50

```
\begin{equation*}
\lim_{k \rightarrow \infty} \biggm[
\sum_{n=1}^k n^2 \Bigm /
\sum_{n=1}^k n^3 \biggm] < \infty
\end{equation*}
```

$$\lim_{k \rightarrow \infty} \left[\sum_{n=1}^k n^2 \Bigm / \sum_{n=1}^k n^3 \biggm \right] < \infty$$

3.4 Almindelige matematiske konstruktioner

Sidebemærkning 3.31. Læseren opfordres desuden til at læse appendix A på side 351 for at lære hvordan man selv laver simple makroer.

Sub- og superscript

Sub- hhv. superscript angives via tegnene `_{\langle argument \rangle}` og `^{\langle argument \rangle}`. Skulle `\langle argument \rangle` kun består af ét bogstav, tal eller et enkelt makronavn (uden argumenter) kan man se bort fra `{}`, men generelt anbefales det at man altid tager dem med. Visse kommende pakker (breqn) vil altid påkræve brugen af `{}`'erne. Fontstørrelsen bliver automatisk mindre når man anvender sub- eller superscript

Eksempel 3.51

```
\begin{equation*}
x^2 + y_i + z_j^{3+35b}.
\end{equation*}
```

$$x^2 + y_i + z_j^{3+35b}.$$

Vi vil se lidt mere på specielle konstruktioner med sub- og superscript i den øvede del. Man kan brug for at kunne skrive 1_6C , så tag et kig i afsnit [10.9 på side 281](#).

Brøker



Advarsel 3.32 (Dobbelt super- eller subscript).

Der gælder en speciel regel vedrørende sub- og superscript. Man må *ikke* lave dobbelte sub- eller superscripts, dvs. ingen $a^b_c^d$. Problemet er at man ikke kan se om man mener $a^{b_c^d}$ eller $a^{b_c^d}$ eller $a_{c^d}^b$. Derfor skal man altid sørge for at huske sine {}-er i en sådan situation.

Sidebemærkning 3.33. Der er en lille bizar hage ved sup- og superscript når man blander det sammen med tegn hvor man har placeret en accent på. Hvor \hat{x}_i^2 fungerer fint (giver x_i^{*2}) så gør \hat{x}_i^2 *ikke*!

Man kan læse en forklaring på <http://groups.google.it/group/comp.text.tex/msg/05a62d404223e13f>. Af underlige årsager bliver man altså nødt til at anvende \hat{x}_i^2 i stedet (det giver \hat{x}_i^{*2}), bemærk {}-parret indsat i starten.

Brøker

Brøker laves via $\frac{\langle tæller \rangle}{\langle nævner \rangle}$. Størrelsen på brøken afhænger af om brøken er skrevet i tekst- eller fremhævet matematiksammenhæng.

```
Hvis  $x > \frac{1}{2}$  gælder at
\l
\frac{1}{4}g(x)^2 - 35 < 0.
\l
```

Hvis $x > \frac{1}{2}$ gælder at

$$\frac{1}{4}g(x)^2 - 35 < 0.$$

Eksempel 3.52

Brøken kan tvinges over i sin teksttilstand via tfrac og i den fremhævede version via dfrac . Man anvender ofte tfrac i fremhævet matematik hvis der er tale om stambrøker og disse er det eneste som rager op på linien

```
\l
\frac{1}{2}\log(5) \quad \quad
\text{tfrac}{1}{2}\log(5).
\l
```

$$\frac{1}{2}\log(5) \quad \frac{1}{2}\log(5).$$

Eksempel 3.53

Der findes også en mindre kendt version der bruges til at lave uendelige (eller lignende) brøker: $\cfrac{\langle tæller \rangle}{\langle nævner \rangle}$

```
\begin{equation*}
\cfrac{1}{1+
\cfrac{1}{1+
\cfrac{1}{1+\cdots}}}
\quad \text{vs.} \quad
\text{dfrac}{1}{1+
\text{dfrac}{1}{1+
\text{dfrac}{1}{1+\cdots}}}
\end{equation*}
```

$$\frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \dots}}} \quad \text{vs.} \quad \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \dots}}}$$

Eksempel 3.54

Til tider har man brug for at lave en form for deling eller lignende for en brøk med en lang tæller. Dette kan opnås via pakken `mathtools`, se også afsnit 4.5. Makroen $\text{splitfrac}\langle \text{øvre} \rangle \langle \text{nedre} \rangle$ er hvad vi leder efter.⁹

⁹ Der findes også en splitdfrac udgave.

3. Introduktion til matematik

Eksempel 3.55



```
\usepackage{mathtools}
----- slut preamble -----
\begin{equation*}
a=\frac{
\splitfrac{aa+ab+ac+ad}%
{+ae+af+ag+ah}
}{2} = 5
\end{equation*}
```

$$a = \frac{aa + ab + ac + ad + ae + af + ag + ah}{2} = 5$$

Binomialkoefficienter

`\binom`
`\tbinom`
`\dbinom`

Disse laves via `\binom{⟨n⟩}{⟨k⟩}`, med varianterne `\tbinom` og `\dbinom`.

Eksempel 3.56

```

 $\binom{n}{k}$  angiver antallet af måder
hvorpå man kan placere $k$
bolde i $n$ kasser.
\begin{equation*}
\binom{n}{0} \equiv 1
\end{equation*}
```

$\binom{n}{k}$ angiver antallet af måder hvorpå man kan placere k bolde i n kasser.

$$\binom{n}{0} \equiv 1$$

Indrømmet denne tekst udgave af en binomialkoefficient er ikke særlig pæn. Her er en anden mulighed:

Eksempel 3.57

```
tekst  $\bigl(\begin{smallmatrix} \\ n \end{smallmatrix}\bigr)$  tekst
```

tekst $\binom{n}{k}$ tekst

Man kan så anvende denne konstruktion til at lave sin egen binomial-makro, i den forbindelse vil makroen `\mathchoice` nok være hjælpsom, se [Eksempel 3.23 på side 56](#) for et eksempl på brugen af `\mathchoice`.

Rodfunktionen

`\sqrt` Makronavnet er oplagt `\sqrt[⟨grad⟩]{⟨mat⟩}`

Eksempel 3.58

```
\begin{equation*}
\sqrt{\frac{n}{n^2+1}} - \sqrt[5]{2} = 11
\end{equation*}
```

$$\sqrt{\frac{n}{n^2+1}} - \sqrt[5]{2} = 11$$

Se desuden [Mittelbach og Goossens \(2004\)](#) side 506–507 som forklarer hvordan man kan justere ved placeringen af roden.

Modulus

Når man skal angive at noget er ækvivalent med noget andet modulo noget tredje, har man forskellige muligheder, se [Eksempel 3.59](#) (bemærk mellemrum de forskellige markroer indsætter).

`\mod`
`\bmod`
`\pmod`
`\pod`

Eksempel 3.59

```
\begin{align*}
& x \equiv y \pmod c \\
& x \equiv y \pmod c \\
& x \equiv y \pmod c \\
& x \equiv y \pmod c \\
& x \equiv y \pmod c
\end{align*}
```

$$x \equiv y \pmod c$$

$$m \bmod n$$

$$x \equiv y \pmod c$$

$$x \equiv y \pmod c$$

Afledt funktion

Afledt funktion

I matematik-mode er apostroffen blevet et alias for tegnet `\prime` skrevet som superscript, (i.e. i math-mode er »'« det samme som »^{\prime}«), herved kan man bare skrive

```
\[ f' \quad f'' \quad f''' \]
```

$$f' \quad f'' \quad f'''$$

Eksempel 3.60

Et af mine ynglingseksempler vedr. hjemmelavede makroer vedrører partielt afledte:

```
\newcommand\diff[3][\partial]%
{\frac{#1 #2}{#1 #3}}
```

— slut preamble —

```
\[ \diff{f}{x}\diff{d}{x}{t} \]
```

$$\frac{\partial f}{\partial x} \frac{dx}{dt}$$

Eksempel 3.61



Eksempel 3.61 viser styrken ved `\newcommand`, vi laver en makro, som tager tre argumenter, hvoraf det første er valgfrit med standardværdien `\partial`, de to obligatoriske argumenter er så funktionsnavnet og det man afleder efter, begge kan evt. være tomme.

Om matricer

Pakken `amsmath` giver os seks beslægtede matrix-konstruktioner:

```
\begin{align*}
\begin{matrix} a & b \\ c & d \end{matrix} & \quad & \begin{pmatrix} a & b \\ c & d \end{pmatrix} \\
\begin{pmatrix} a & b \\ c & d \end{pmatrix} & \quad & \begin{bmatrix} a & b \\ c & d \end{bmatrix} \\
\begin{Bmatrix} a & b \\ c & d \end{Bmatrix} & \quad & \begin{vmatrix} a & b \\ c & d \end{vmatrix} \\
\begin{Vmatrix} a & b \\ c & d \end{Vmatrix} & \quad & \begin{Vmatrix} a & b \\ c & d \end{Vmatrix}
\end{align*}
```

$$\begin{matrix} a & b \\ c & d \end{matrix} \quad \begin{pmatrix} a & b \\ c & d \end{pmatrix} \\ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad \begin{Bmatrix} a & b \\ c & d \end{Bmatrix} \\ \begin{vmatrix} a & b \\ c & d \end{vmatrix} \quad \begin{Vmatrix} a & b \\ c & d \end{Vmatrix}$$

Eksempel 3.62

De forskellige navne henviser altså til hvilket hegn der er anvendt. Alle indgangene i matricerne bliver automatisk centreret. Dette kan være et problem i visse tilfælde. Igen løser `mathtools` problemet med *-ede versioner af ovenstående environments. Her kan man så, med et valgfrit argument angive alignment i søjlerne via søjlerne (`l`, `c` (standardvalg) eller `r`). Faktisk kan man også kombinere dette med en `dcolumn`-søjlespecifier, se afsnit 7.4.1.

mathtools

dcolumn

```
\usepackage{mathtools}
```

— slut preamble —

```
\begin{equation*}
\begin{bmatrix} -1 & 2 \\ 0 & -5 \end{bmatrix} + A = 0
\end{equation*}
```

$$\begin{bmatrix} -1 & 2 \\ 0 & -5 \end{bmatrix} + A = 0$$

Eksempel 3.63



Skulle man have brug for en matrix i selve teksten, eller bare vil have noget med en mindre font end $\langle X \rangle$ *matrix* kan man anvende *smallmatrix*-environmentet. Her skal man selv huske at indsætte hegn.

smallmatrix

3. Introduktion til matematik

Eksempel 3.64

```

lad $A=\left(
\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$
så er
\begin{equation*}
A\bigl[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\bigr]
\biggr]
=\bigl[\begin{smallmatrix} a \\ c \end{smallmatrix}\bigr]
\end{equation*}

```

lad $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ så er

$$A \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} a \\ c \end{bmatrix}$$

Med mindre man altså har husket at opdatere sin L^AT_EX-installation, en opdatering af mathtools-pakken i 2011, gav os nye *-small-* udgaver af matricerne fra [Eksempel 3.62](#) på [forrige side](#).

Eksempel 3.65

```

\usepackage{mathtools}
----- slut preamble -----
\begin{align*}
\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}
&\quad
\begin{psmallmatrix} a & b \\ c & d \end{psmallmatrix}
&\quad
\begin{bsmallmatrix} a & b \\ c & d \end{bsmallmatrix}
&\quad
\begin{Bsmallmatrix} a & b \\ c & d \end{Bsmallmatrix}
&\quad
\begin{vsmallmatrix} a & b \\ c & d \end{vsmallmatrix}
&\quad
\begin{Vsmallmatrix} a & b \\ c & d \end{Vsmallmatrix}
\end{align*}

```

$$\begin{matrix}
 a & b & \begin{pmatrix} a & b \\ c & d \end{pmatrix} \\
 \begin{bmatrix} a & b \\ c & d \end{bmatrix} & \begin{Bmatrix} a & b \\ c & d \end{Bmatrix} \\
 \begin{vmatrix} a & b \\ c & d \end{vmatrix} & \begin{Vmatrix} a & b \\ c & d \end{Vmatrix}
 \end{matrix}$$

Sidebemærkning 3.34. Man bør ikke anvende de store matricer i selve teksten, men kun i fremhævede formler, idet de ellers forstyrrer linieafstanden.

Sidebemærkning 3.35 (Max antal (mulige) søjler i \mathcal{AMS} matricer).

Sammenligner man matrix-konstruktionerne ovenfor med med *array* (næste afsnit), så ser man at \mathcal{AMS} -konstruktionerne ikke siger noget om hvor mange søjler der er i matricen. De har i stedet et øvre antal som er predefineret. Dette er normalt ikke noget problem, men når man har over 10 søjler så får man en fejl. Dette løser man via

Eksempel 3.66

```
\setcounter{MaxMatrixCols}{20}
```

placeret i preamble efter man har indlæst amsmath-pakken.

Matrix konstruktioner via array

Man har også en tredje konstruktion man kan anvende til matricer og matrixagtige konstruktioner:

```

\begin{array}{\langle cols \rangle}
\langle celle 1 \rangle & \langle celle 2 \rangle & \dots & \langle celle n \rangle \\
\dots

```

Om matricer

```
\end{array}
```

Med `{<cols>}` skal man specificere justeringen af søjlerne, her anvendes bogstaverne »c«, »l« og »r« som hver især også betyder én søjle, ergo er »ccl« fire søjler. Man skal selv indsætte hegn.

```
\[
\left[
\begin{array}{ccc}
a & b & c \\
d & e & f
\end{array}
\right]
\]
```

$$\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}$$

Eksempel
3.67

Syntaksen er nøjagtigt den samme som for tabeller (se kapitel 7 på side 165).

Sidebemærkning 3.36. \mathcal{AMS} -matricerne er faktisk specielle versioner af `array`, konfigureret med `MaxMatrixCols` søjler.

Med denne konstruktion kan man sagtens lave streger i matricen.

```
\[
\left[
\begin{array}{@{} c | c @{}}
A & B \\
C & D
\end{array}
\right]
\]
```

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

Eksempel
3.68

Bemærk brugen af »@{}« for at fjerne den ekstra luft ved hegnene. Man kan desuden indlejre matricer.

```
\[ \left[
\begin{array}{ c | c }
\begin{matrix}
\begin{matrix}
0 & 1 \\
1 & 0
\end{matrix} \\
& \\
\begin{matrix}
\text{\huge $0$} \\
\end{matrix}
\end{matrix} \\
\hline
\begin{matrix}
\begin{matrix}
\text{\huge $0$} \\
\end{matrix} \\
& \\
\begin{matrix}
0 & c \\
d & 0
\end{matrix}
\end{matrix}
\end{array}
\right]
\]
```

$$\begin{bmatrix} \begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix} & \text{\huge 0} \\ \text{\huge 0} & \begin{matrix} 0 & c \\ d & 0 \end{matrix} \end{bmatrix}$$

Eksempel
3.69



Her bliver vi nødt til at anvende `\text{\huge 0}` for at kunne få nullet i en større størrelse, thi man må *ikke* bruge makroerne til skift af fontstørrelse indeni matematik-konstruktioner.

3. Introduktion til matematik

Et hack med array

Environmentet *array* fungerer på samme måde som environmentet *tabular* som normalt anvendes til tabeller (*tabular* beskrives senere, se kapitel 7 på side 165). Vi har allerede set at man kan indsætte linier en en *array*, men vi kan også *fuske* lidt med mellemrummet mellem søjlerne. Dette er dog kun noget som bør anvendes i meget højt specialiserede tilfælde. Eksemplet som følger er på opfordring af Morten Risager.

Eksempel 3.70

```
\[ \left[
\begin{array}%
{c @{\hspace{-15pt}} c
 @{\hspace{-15pt}} c c }
y_1\cdots y_{n-1} \\
& y_1\cdots y_{n-2} \\
& & \ddots \\
& & & 1
\end{array}
\right]
```

$$\left[\begin{array}{ccc} y_1 \cdots y_{n-1} & & \\ & y_1 \cdots y_{n-2} & \\ & & \ddots \\ & & & 1 \end{array} \right]$$

Ideen er at få en diagonalmatrix til at fremstå lidt mere *visuelt* diagonal. Det er ren kosmetik, så dette er *kun* noget man skal bruge i specialtilfælde.

Blkarray-pakken

blkarray Med *blkarray*-pakken kan man lave nogle lidt mere specielle matricer. Konstruktionen er specielt nyttig hvis man skal have nummereret søjlerne i en matrix. Vi vil her bare give et par eksempler og så ellers henvise til [Carlisle \(1999a\)](#). De forskellige environments kan sagtens anvendes både i og udenfor matematik-mode.

Eksempel 3.71

```
\usepackage{blkarray}
\begin{blockarray}{c c c c }
& & j & \\
\begin{block}{c [ c c c ]}
& * & & \vdots & * \\
i & \dots & ij & & * \\
& * & & * & * \\
\end{block}
\end{blockarray}
```

$$i \begin{array}{ccc} & j & \\ * & \vdots & * \\ \dots & ij & * \\ * & * & * \end{array}$$

Det tager en lille smule tid at forstå hvordan konstruktionen skal anvendes, men med lidt øvelse er den nem nok at forstå.

Matrixoperationer

gauss Skulle man have brug for f.eks. at illustrere matrixoperationer, så kan pakken *gauss* være en hjælp. Vi vil ikke gå i dybden med pakken, men i stedet henvise til [Kauers \(2002\)](#). Her er et eksempel:

Gaffelfunktioner

```
\usepackage{gauss,amsmath}
```

----- slut preamble -----

```
\begin{align*}
& \& \begin{matrix} \text{1} & \text{2} & \text{3} \\ \text{4} & \text{5} & \text{6} \\ \text{7} & \text{8} & \text{9} \end{matrix} \\
& \colops \\
& \swap{1}{2} \\
& \rowops \\
& \swap{0}{1} \\
& \mult{0}{\cdot 7} \\
& \add{5}{1}{2} \\
& \end{matrix} \\
& \\\[4mm] \\
& \& \quad = \\
& \begin{matrix} \text{28} & \text{42} & \text{35} \\ \text{1} & \text{3} & \text{2} \\ \text{12} & \text{24} & \text{18} \end{matrix} \\
& \end{matrix} \\
& \end{align*}
```

$$\begin{matrix} \left(\begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix} \right) \begin{matrix} \leftarrow | \cdot 7 \\ \leftarrow \left. \begin{matrix} \\ \end{matrix} \right\} 5 \\ \leftarrow + \end{matrix} \end{matrix} \\ \\ = \begin{pmatrix} 28 & 42 & 35 \\ 1 & 3 & 2 \\ 12 & 24 & 18 \end{pmatrix} \end{matrix}$$

Eksempel

3.72



Outputtet skal læses som at: Man har ombyttet søjle 1 og 2 (første søjle/række har nummer 0. Så har man byttet om på række 0 og 1, samt ganget den nye række 0 med 7. Så har man ganget række 1 med 5 og lagt den til række 2.

Gaffelfunktioner

For de, som ikke kender navnet, er der altså tale om konstruktioner lignende

```
\begin{equation*}
f(x)= \begin{cases}
1 & \text{\textit{for } } \$x \in \mathbb{Q} \\
0 & \text{\textit{ellers}}
\end{cases} \\
\end{equation*}
```

$$f(x) = \begin{cases} 1 & \text{for } x \in \mathbb{Q} \\ 0 & \text{ellers} \end{cases}$$

Eksempel

3.73

Environmentet *cases* vil altså lave en $n \times 2$ matrix-konstruktion med en stor »{« til venstre. Indgangene skrives automatisk i tekststyle, dvs.

cases

```
\begin{equation*}
f(x)= \begin{cases}
\sum_{n=1}^{\infty} n^x & \text{\textit{for } } x \in \mathbb{Q} \\
0 & \text{\textit{ellers}}
\end{cases} \\
\end{equation*}
```

$$f(x) = \begin{cases} \sum_{n=1}^{\infty} n^x & \text{for } x \in \mathbb{Q} \\ 0 & \text{ellers} \end{cases}$$

Eksempel

3.74

Ønsker man at have indgangene i fuldstørrelse (displaystyle) kan man anvende *dcases* fra pakken *mathtools*, eller indsætte `\displaystyle` pr. håndkraft.

dcases

mathtools

3. Introduktion til matematik

Eksempel 3.75



```
\usepackage{mathtools,amssymb}
```

----- slut preamble -----

```
\begin{equation*}
f(x)= \begin{dcases}
\sum_{n=1}^{\infty} n^x &
\text{for } x \in \mathbb{Q} \\
0 & \text{ellers}
\end{dcases}
\end{equation*}
```

$$f(x) = \begin{cases} \sum_{n=1}^{\infty} n^x & \text{for } x \in \mathbb{Q} \\ 0 & \text{ellers} \end{cases}$$

Pakken `mathtools` har også et par andre varianter af *cases* konstruktionen, f.eks. *cases** hvor højre søjle automatisk sættes i tekst-mode og tilsvarende for *dcases**. Fornyligt blev der desuden tilføjet versioner som har parenteser til højre i stedet for til venstre.

Nogle gange kan det være en fordel at angive formelnummer til hver indgang i en en gaffel. Dette kan vi gøre med en konstruktion fra pakken `empheq`.

Eksempel 3.76



```
\usepackage{empheq}
```

----- slut preamble -----

```
\begin{empheq}[left={f(x)=\empheqlbrace}]{align}
&x^2 && x>0 \\
&0 && \text{ellers}
\end{empheq}
```

$$f(x) = \begin{cases} x^2 & x > 0 & (1) \\ 0 & \text{ellers} & (2) \end{cases}$$

Fordi vi anvender *align* skal man lige huske at have styr på sine &'er. Ønsker man mere kontrol over luften, så anvend *alignat* i stedet, se afsnittet »*Alignat*« på side 92. Samtidigt man det være nyttigt at kombinere dette med *subequations*-konstruktionen, se »*Undernummerering af formelnummer*« på side 101.

Over/underbrace

`\overbrace`
`\underbrace`

Med makroerne `\overbrace{<txt>}` og `\underbrace{<txt>}` kan man lave en { som spænder over/under den givne tekst. Man kan så tilføje kommentarer som sub- eller superscript.

Eksempel 3.77

```
\begin{equation*}
\underbrace{\overbrace{\dbx[15mm]}^{2}}
+ \overbrace{\dbx[2cm]}^{4} = 6
\end{equation*}
```

Hvis teksten man spænder over ikke er ret bred kan det blive ret grimt. Pakken `mathtools` har også en `\overbracket` og en `\underbracket` med samme syntaks. Pakken giver desuden en bedre implementation af `\underbrace`, se Høgholm (2008b).

`\overbracket`
`\underbracket`

Grænser i flere lag

Grænser i flere lag kan opnås via

`\substack`

```
\substack{<linie 1> \& <linie 2> \dots}
```

Eksempel 3.78

```
\[
\sum_k \sum_{\substack{j \geq k \\ j \text{ lige}}}
\]
```

$$\sum_k \sum_{\substack{j \geq k \\ j \text{ lige}}}$$



Formaning 3.37. Dette er igen en god grund til *ikke* at anvende `\left... \right`-konstruktioner omkring de store operatorer.

```
\[
\sum_k \left[
\sum_{\substack{j \geq k \\ j \text{ lige}}}
\right]
\]
```

$$\sum_k \left[\sum_{\substack{j \geq k \\ j \text{ lige}}} \right]$$

Eksempel 3.79

Restingering/evaluering af funktioner

Til tider skal man restingere en funktion eller evt. evaluere en partielt afledet i et punkt. En syntaks man ofte anvender til dette er

```
\begin{equation*}
\frac{\partial f}{\partial x}
\Big|_{x=0}
\end{equation*}
```

$$\frac{\partial f}{\partial x} \Big|_{x=0}$$

Eksempel 3.80

Hvis man ikke skalerer `\Big|` så kan man godt risikere at det man evaluerer i kommer til at sidde lidt for højt på `\Big|`, sammenlign Eksemplerne 3.80 og 3.81.

```
\[ F|_{x=0} \]
```

$$F|_{x=0}$$

Eksempel 3.81

Når, det man placerer subscript på, har en 'normal' højde og der ikke er noget superscript, så vil L^AT_EX placere subscriptet højere end, hvis der var et superscript. Dette man man så afhjælpe ved at anvende et 'tomt' superscript, f.eks. med `\mathstrut`.¹⁰

`\mathstrut`

```
\[
F|_{x=0}^{\mathstrut} \quad F|_{x=0}^{\quad}
\quad \text{versus} \quad F|_{x=0}
\]
```

$$F|_{x=0}^{\mathstrut} \quad F|_{x=0}^{\quad} \quad \text{versus} \quad F|_{x=0}$$

Eksempel 3.82

3.5 Nogle almindelige hjemmelavede makroer

Sidebemærkning 3.38. I sit DK-TUG foredrag i februar 2010 (<http://www.tug.dk/foredrag/2010-02-02>) kom Rasmus Villemoes med en ret god pointe vedr. brugen af egne makroer. Nemlig at man skal passe på når man bruger den samme syntaks for to forskellige ting. Rasmus' eksempel var kardinaliteten af en mængde A . I nogle artikler skrives kardinaliteten som $|A|$, så man kunne nemt finde på at anvende `\abs{A}`. Men det ville være en dårlig ide, for hvad nu hvis man senere finder ud af at man bedre kan lide syntaksen $\#A$? Så skal man rette en masse pr. håndkraft og sidde og tolke på teksten i mens.

Løsningen er naturligvis at lave en separat makro til at betegne kardinaliteten af mængder, f.eks.

```
\newcommand\cardiNum[1]{\abs{#1}}
```

så kan man nemt skifte det senere.

Eksempel 3.83

¹⁰ Det er ikke så tydeligt med den font som anvendes i denne bog.

3. Introduktion til matematik

Diverse

Eksempel

3.84



```
\newcommand\morf[4][\to]{#2\colon #3#1 #4} % \morf{f}{A}{B} = f\colon A\to B
\newcommand\diff[3][\partial]{\frac{#1 #2}{#1 #3}}
\newcommand\inverse^{^{-1}} % hvis man tit skal skrive ^{-1}, ie $X\inverse$
\newcommand\isomorph{\cong} % \cong anvendes ofte til dette
\newcommand\homotopyeq{\simeq} % fra topologi
\newcommand\DistAs{\sim} % X \DistAs N(0,1)
\newcommand\ObsFromDist{\sim\sim} % observation fra variabel distribueret som...
\newcommand\DistEq{\stackrel{\smash{\scriptscriptstyle(d)}}{=}} % lighed i distribution
```

Sidebemærkning 3.39. Forslag modtages gerne.

Ting man finder senere i bogen

Af forskellige årsager er der en del ting som vi har valgt ikke at tage med i selve introduktionsdelen af bogen, dette gælder f.eks. layout konfigurering, tabeller, litteraturlister osv. Disse finder man i senere kapitler i bogen. Her er en hurtig oversigt:

- Yderligere informationer om matematiske konstruktioner samt om konstruktionen af Sætninger, Lemmaer etc. finder man i Kapitel 4 på side 83 samt i Kapitel 5 på side 115.
- Inklusion af ekstern grafik, dvs. tegninger, billeder etc, finder man i Kapitel 6 på side 135.
- Konstruktion og opstilling af tabeller, se Kapitel 7 på side 165
- Litteraturliste og citering af litteratur, se Kapitel 8 på side 197.
- Justering af layout, se Kapitel 14 på side 337 samt Kapitel 12 på side 301.

God fornøjelse.

Overbyggende L^AT_EX

Matematik på et højere niveau

I dette kapitel vil vi forklare lidt mere om de forskellige konstruktioner vedrørende matematik. Vi vil primært koncentrere os om fremhævet matematik, idet dette er hvor man har mest råderum.

I mange af eksemplerne vil vi anvende den hjemmelavede makro `\dbx` som giver resultatet \square i forskellige størrelser, thi det er udtrykket vi vil vise frem i dette tilfælde, ikke indholdet..

Vi begynder kapitlet med anbefalinger til hvor og hvordan man deler matematiske formler.

4.1 Matematisk typografi

Dette er ikke noget man lærer nogen steder, så her er et minikursus.

4.1.1 Hvor må man dele en formel?

Man har sikkert allerede opdaget at L^AT_EX automatisk vil lave linieombrydning visse steder i tekstbaserede formler (formler i $\$. . \$$) mens den overhovedet ikke laver nogen linieombrydning i fremhævet matematik. Der findes vejledninger i hvordan formler bør deles over linier, dog ikke så meget regler/retningslinier men mere i tråd med »sådan bør man gøre«. Følgende vejledning er baseret på anbefalingerne fundet i [Swanson \(1999, kapitel 3\)](#), [Chicago Manual of Style](#) (kapitel 14) og [Knuth \(1986\)](#).¹

Definition 4.1. For at kunne få terminologien på plads så lad os beskrive symboler så som: $+$, $-$, \cdot , \times , \pm , \circ , \otimes o.lign. som værende *operationer* og symboler i kategorien: $=$, \leq , \simeq , \subseteq , \in , \rightarrow , \Rightarrow etc. som værende *relationer*. Samtidigt betegnes parenteser og andre (par af) afgrænsere/delimiters som *hegn*. De store operatorsymboler \sum , \oplus etc. anses for at være i en klasse for sig, vi vil her betegne dem som *store operatorer*.

4.1.2 Formelombrydning af matematik i teksten

Med hensyn til liniebrydningen af matematik i teksten, er dette noget man først bør begynde at rette på, når man er helt færdig med at skrive sin tekst. Mens man skriver, eller hvis man senere skal justere marginerne, ved man jo på forhånd ikke hvor linieombrydningerne kommer til at være – og derfor er det spild af tid at rette den slags mens man skriver. Ret det i stedet til allersidst. Dog kan man sagtens ombryde fremhævede formler mens man skriver teksten.

Vejledningen følger herunder, visse kan måske lyde en smule sære, men husk at dette gælder *matematikken i teksten*, vejledningen for fremhævede udtryk kommer senere, i afsnit [4.1.3 på side 86](#).

Vi vil anvende tegnet »▲« til at markere de punkter hvor det er tilladt at dele.

¹ [Chicago Manual of Style](#) og [Knuth](#) nævner begge [Swanson](#).

Sidebemærkning 4.2. For god ordens skyld minder vi lige om at punkterne (a)–(f) herunder beskriver anbefalingerne om hvordan man som skribent bør dele sine formler »i teksten«. Der er altså *ikke* tale om hvordan L^AT_EX vil om muligt dele en formel i teksten. Dette ser vi på i afsnittet »*Hvor vil L^AT_EX automatisk dele matematiske udtryk i teksten?*« på side 85.

Bemærk også at der her er tale om formler i teksten, *ikke* om fremhævede formler, hvilket vi kommer til på side 86.

(a) Om nødvendigt må der deles *før* eller *efter* enhver relation eller operator (samt efter komma og semikolon) som *ikke* er omkranset af hegn.

$$\dots \text{ tekst } \dots \quad f(x, y) = (\square + \square) + \square \quad \dots \text{ tekst } \dots$$

Oftest er det i tekst-matematik bedre at dele efter relationer end før (thi så ved læseren, at der kommer noget mere på næste linie). For mere vedrørende indhegnede udtryk, se regel (f).

(b) En formel må deles ved ethvert (manuelt) indsat mellemrum større end *thin space* (`\,`), f.eks. indsat spacing med `_` og `\quad`. Et simpelt eksempel er

Eksempel 4.1

`$f(x_n)=1_ \text{for all } n \geq 2$,`

$$f(x_n) = 1 \quad \forall n \geq 2,$$

her må der gerne deles ved den indsatte spacing (`_`) foran \forall .

Hvis der er mulighed for valg, tager regel (b) præcedens over regel (a).

(c) Vedrørende store symboler (fraregnet integraltegn) bør linieombrydningen vente indtil første operation eller relation som forekommer *udenfor* hegn.

$$\dots \text{ tekst } \dots \quad \sum (\square + \square) + \square = \square \quad \dots \text{ tekst } \dots$$

Vedrørende integraler bør man vente med delingen til efter dx etc., dog bør $\int dx + dy$ ikke deles.

(d) Har man to tilstødende par af hegn, må man dele mellem disse forudsat at der efter delingen indsættes et passende gangetegn (\cdot eller \times).

$$\dots \text{ tekst } \dots \quad (\square + \square) (\square + \square) \quad \dots \text{ tekst } \dots$$

Hvorvidt gangetegnet skal stå i slutningen af linien eller i begyndelsen af den næste er op til skribenten. Se også afsnittet »*Automatisk indsættelse af gangetegn ved linieombrydning*« på side 86.

Swanson anbefaler at denne regel undertrykkes når det indhegnede udtryk har et stort symbol foran.

(e) Man bør aldrig dele et udtryk ved tegnet »/«.

(f) Vedrørende operatorer og komma / semikolon omkranset af hegn, skal man generelt være forsigtig med hvor man deler, hvis man bliver nødt til at dele. Det er altid bedst at lade være.

Fortsættes på næste side

Hvad gør man når man ikke kan få det til at passe?

Swanson (1999) anbefaler:

For operatorer omkranset af hegn må der deles før eller efter operatoren (eller efter komma/semikolon) hvis følgende er opfyldt:

- (i) Hvis der foran et endnu ikke-afsluttet starthegegn **ikke** findes et bogstav, tal, symbol, hegn eller en stor operator, og
- (ii) det samme er tilfældet for det efterfølgende sluthegegn.

Eksempler:

$$\text{tekst} \dots 5(\square + \square - \square) \quad \sum(\square + \square \oplus \square) \dots \text{tekst}$$

$$(\square + \square_{\blacktriangle} \square) \quad ((\square + \square)_{\blacktriangle} \square_{\blacktriangle} (4\square))$$

Jeg ville nok personligt ikke være så restriktiv, men i stedet lade min intuition råde. Generelt skal man dog være opmærksom på ikke at efterlade meget korte udtryk på en af de to linier. F.eks. ville jeg ikke tillade nogen af følgende delinger

$$\dots \text{tekst} \dots (a_{\blacktriangle} + b) \dots \text{tekst} \dots$$

da »(a« hhv. »b)« er alt for korte til at efterlade i hhv. slutningen eller starten af en linie. Tilsvarende ville jeg også dele ved komma hvis udtrykket alligevel var til det, se [Eksempel 4.2 på den følgende side](#).

Det hele kommer selvfølgelig an på situationen, Dog bør man f.eks. nok ikke dele $f(x, y) = \sqrt{x + y^2}$ ved kommaet, med mindre intet andet kan gøres.

Hvad gør man når man ikke kan få det til at passe?

Hvis man ikke kan få sin formel til at passe ordentligt ind, har man to muligheder:

- (1) omformuler noget af teksten således at linieombrydningen bliver anderledes, eller
- (2) lav udtrykket om fra at være inline (i teksten) til at være fremhævet.

Ved mindre udtryk vil (1) være at foretrække, mens det ved mere komplicerede udtryk i stedet altid vil være bedre at fremhæve formlen. Dette gælder også generelt, det er bedre for læsbarheden af det skrevne, hvis man fremhæver komplicerede formler i stedet for at skrive den i teksten.

Hvor vil L^AT_EX automatisk dele matematiske udtryk i teksten?

T_EX (og dermed L^AT_EX) har lov til at lave linieombrydning af tekstbaseret matematik *efter* relationer og operationer – hvor deling efter relationer har højere prioritet end deling efter operationer. L^AT_EX bryder *aldrig* automatisk *før* relationer eller operationer. På tilsvarende vis vil L^AT_EX *aldrig* ombryde efter komma eller semikolon.

Bemærk desuden at L^AT_EX *ikke* kigger på om en operation er indhegnet eller ej, så hvis den laver delinger som man ikke synes om, må man selv træde til for at hjælpe. Ønsker man at tillade at L^AT_EX ombryder udtrykket et bestemt sted kan man gøre opmærksom på dette via »\allowbreak«, som virker lidt på samme måde som \- gør ved orddeling af ord (bare uden bindestregen), den informerer altså L^AT_EX om at på dette sted må den også gerne lave ombrydning. Hvorvidt den så bliver anvendt kommer selvfølgelig an på situationen. \allowbreak anvendes ofte hvis man ønsker at tillade automatisk liniedeling ved et komma:

\allowbreak

4. Matematik på et højere niveau

Eksempel 4.2

givet at vi har en funktion $f(x_1, \dots, x_n, y_1, \dots, y_n)$ af $2n$ variable

givet at vi har en funktion $f(x_1, \dots, x_n, y_1, \dots, y_n)$ af $2n$ variable...

Tilsvarende kan man fratage delingsmuligheden ved at indsætte `\nobreak` på det sted hvor man *ikke* ønsker at L^AT_EX laver automatisk linieombrydning. Et eksempel kunne være udtrykket $n \rightarrow 0$ hvor L^AT_EX gerne må dele udtrykket efter \rightarrow , dette afværges via `\to\nobreak 0`.



Farlig (læs teksten) 4.3. I den lidt farligere ende finder man makroen `\linebreak`, som vil fremtvinge et lineskift. Forskellen mellem `\linebreak` og `\newline` er at førstnævnte vil strække afstanden mellem ordene således at vi fortsat har en lige højremargin. Makroen bør anvendes med den største forsigtighed, da resultatet nemt kan blive ret grimt.

Automatisk indsættelse af gangetegn ved linieombrydning

Vender vi tilbage til situationen hvor vi har et produkt som skal deles. Her er problemet netop det usynlige gange tegn som skal være synligt efter delingen. Dette kan L^AT_EX faktisk godt håndtere. Anvend makroen `*`:

Eksempel 4.3

```
\dbx[3cm]
$(a+b)\*(c+d)+(a+b)\*(c+d)$
```

$(a + b)(c + d) + (a + b) \times (c + d)$

Hvis man ønsker at lave om på `*` således at den anvender `\cdot` i stedet for `\times`, så kan dette gøres via

Eksempel 4.4

```
\renewcommand{\*}{\discretionary{\thinspace\hbox{\ensuremath{\cdot}}}{}}{\}}
```



4.1.3 Formelombrydning i fremhævede formler

Reglerne (a)–(f) på side 84 gælder for så vidt også for fremhævede matematikudtryk, dog med især en undtagelse. Man må *kun* dele *før* operationer eller relationer og *aldrig* efter! Desuden gælder naturligvis at hvis andet ikke er muligt er det tilladt at bryde reglerne (a)–(f). Man kan ikke nedskrive generelle regler for hvordan fremhævede matematiske udtryk skal deles, dette kommer an på den enkelte situation og valget af ombrydningspunkt kræver matematisk indsigt dog stadig med følgende in mente:

I fremhævet form deles altid før en relation eller operation, aldrig efter!

Opstilling af fremhævede (brudte)formler

Sidebemærkning 4.4. Dette er faktisk ikke helt korrekt at man ikke kan opskrive regler. Man kunne argumentere for at en del af beskrivelserne på de følgende sider kan danne grundlag for en automatiseret ombrydning af fremhævet matematik.

At have sådan en automatiseret opbrydning af fremhævet matematik vil også være en *kæmpe* fordel, man slipper for problemerne med at matematikken skal sættes anderledes op hvis man ændrer marginer. Journaler kan nemt sætte artiklen op til at fungere i to-søjle format, uden at pille ved matematikken.

Der findes en pakke som kan det meste af dette, men den indeholder desværre også en del bugs, bl.a. kan den ikke anvendes sammen med *memoir*-klassen.

Der er dog håb, idet Morten Høgholm (mathtools og empheq pakkerne (beskrives senere) samt medlem af L^AT_EX3-projektet) har påtaget sig at lave en ordenlig implementation af den pågældende pakke. Der kommer forhåbentlig en officiel udgivelse i anden halvdel af 2007. Indtil videre ser pakken ret fantastisk ud, men det er også ikke så lidt af en programmeringsmæssig bedrift faktisk at få dette til at virke.

Opstilling af fremhævede (brudte)formler

Det stridende punkt vedrørende fremhævede formler bliver nu ikke mere hvor må man dele – for det er næsten det samme som for matematik i tekst – det bliver mere et spørgsmål om hvordan man stiller tingene op. Dette vil vi give et bud på her. Konstruktionerne som er anvendt vil blive gennemgået i det efterfølgende afsnit. Den lodrette linie i visse af de efterfølgende eksempler, markerer hvor vi har valgt at placere en eventuel alignment. Forklaring til hvordan de enkelte eksempler er lavet findes i afsnit 4.2.5 på side 97.

Hvis en lang udregning deles (og der er plads) da justeres til venstre for relations tegn.

$$\begin{array}{l} \boxed{} = \boxed{} \\ \leq \boxed{} \\ = \boxed{}. \end{array}$$

Eksempel 4.5

Hvis *ikke* der er plads, alignes helt til venstre og alle efterfølgende linier i udregningen indenteres med 2 em (`\qqquad`).² Se desuden afsnittet *Justering af indryk ved brudte formler* på side 105 for en alternativ konstruktion til at opnå dette udseende.

$$\begin{array}{l} \boxed{} = \boxed{} \\ = \boxed{} \\ = \boxed{}. \end{array}$$

Eksempel 4.6

Variant (anvendes hvis det første udtryk er meget langt):

$$\begin{array}{l} \boxed{} \\ = \boxed{} \\ = \boxed{}. \end{array}$$

Eksempel 4.7

² Man kan selvfølgelig, hvis det er nødvendigt, anvende mindre indrykning, der bør dog være en indrykning for at fremhæve den første linie.

4. Matematik på et højere niveau

Hvis et udtryk ikke kan være på linien, deles før operationer og disse linier indenteres yderligere for at fremstå som en samlet enhed. Ofte vil \quad være en fin indrykning til højre for alignmentet (hvis man aligner til venstre for relationen). Her betyder den ekstra \quad-indrykning at linierne som starter med operationer rykkes ind *efter* relationen som man i forvejen alignede på.

Eksempel 4.8

$$\begin{aligned} &= \\ &+ \\ &- \\ &= \end{aligned}$$

Man kan naturligvis også justere til højre for relationerne, hvorved man undgår at skulle indrykke de brudte linier.

Eksempel 4.9

$$\begin{aligned} &= \\ &+ \\ &- \\ &= \end{aligned}$$

Her skal man dog igen være en smule opmærksom på hvad man gør. Se Tip 4.8 på side 91.

Er der tale om en enkelt formel som dog er meget lang kan Eksempel 4.6 på foregående side og 4.7 også skrives hvor man aligner ved operationer, men er der mere end en relation til stede bør det *altid* være disse man aligner på.

Eksempel 4.10

$$\begin{aligned} &= \\ &+ \\ &\times \end{aligned}$$

Variant:

Eksempel 4.11

$$\begin{aligned} &- \\ &\otimes \end{aligned}$$

Hvis det kun er første linie som er ret lang, kan man naturligvis indentere de efterfølgende linier endnu mere, for at få disse centreret lidt bedre under det brede udtryk. En tredje variant anvendes til tider (se diskussionen vedrørende *multiline*, side 94).

Eksempel 4.12

$$\begin{aligned} &= \\ &+ \end{aligned}$$

Hvis et indhegnet (med parenteser af forskellig art) udtryk deles over flere linier bør dette (hvis muligt) afspejles i indrykningen af de efterfølgende linier. Her vil \phantom være til stor hjælp, se afsnittet *Ser du spøgelse?*, side 99.

Eksempel 4.13

$$\begin{aligned} &= \\ &\times \\ &- \\ &= \end{aligned}$$

Tegnsætning i matematik

Er der tale om en *samling* af forskellige formler bør disse enten justeres på en passende relation:

$$\begin{array}{l|l} \boxed{} & = \boxed{} \\ \boxed{} & = \boxed{} \\ \boxed{} & = \boxed{}, \end{array}$$

Eksempel
4.14

eller centrereres over hinanden:

$$\begin{array}{l} \boxed{} = \boxed{} \\ \boxed{} \leq \boxed{} \\ \boxed{} \subseteq \boxed{}. \end{array}$$

Eksempel
4.15

Centrering kan også anvendes hvis man kun f.eks. har et enkelt »=« men hvor den ene side er meget lang. Her kan en indrykning på anden linie på kun 2em blive lidt for lidt.

$$\begin{array}{l} \boxed{} \\ = \boxed{} \end{array}$$

Eksempel
4.16

Dette kan alt efter materialet også varieres

$$\begin{array}{l|l} \boxed{} & = \boxed{}, \\ \boxed{} & = \boxed{}, \end{array} \qquad \begin{array}{l|l} \boxed{} & = \boxed{}, \\ \boxed{} & = \boxed{}, \end{array}$$

Eksempel
4.17

eller

$$\begin{array}{l|l} \boxed{} & = \boxed{}, \\ \boxed{} & = \boxed{}, \\ \boxed{} & = \boxed{}, \end{array} \qquad \begin{array}{l|l} \boxed{} & = \boxed{}, \\ \boxed{} & = \boxed{}. \end{array}$$

Eksempel
4.18

Se afsnit [4.2.5 på side 97](#) for en kort forklaring på hvordan disse eksempler er lavet, samt afsnittet om `\MoveEqLeft`-makroen (side [105](#)) som er den jeg personligt anvender aller mest.

4.1.4 Tegnsætning i matematik

Kommaterings-tegnsætningen af matematik i *teksten* foregår uden problemer som man er vant til. Dog med den regel at kommateringstegnet står *uden for matematik-mode*, altså »`\$x`», « (udenfor matematik-mode) og *ikke* »`\$x`», « (indeni matematik-mode).

Hvad angår matematik i fremhævet form er vejledningen tilsvarende:

Kommatér dine udtryk som var de skrevet i den almindelige tekst – dog nu med kommateringstegnet inkluderet i matematik mode.

Se [Eksempel 3.3 på side 47](#) samt [4.19](#) herunder.

Der gælder dog selvfølgelig visse undtagelser vedrørende kommatering af matematik i fremhævet form:

- (a) Det kan være nødvendigt at indsætte noget mellemrum før kommateringen hvis denne ellers ville forstyrre det matematiske udtryk. Ofte vil `_` være nok.

4. Matematik på et højere niveau

- (b) Hvis kommatring overhovedet ikke passer ind i det matematiske udtryk, skal man sørge for at afslutte sætningen før udtrykket og starte en ny sætning efter udtrykket. På denne måde kan det matematiske udtryk ses som en sætning i sig selv og behøver derfor ikke at blive kommateret.

Det næste eksempel viser både hvordan man normalt kommaterer i fremhævet matematik, samt hvad man gør ved en formel som ikke kan kommateres.

Eksempel 4.19



```
\usepackage[all,line,dvips]{xy}
```

----- slut preamble -----

```
For  $x \neq 0$  gælder at
```

```
\[ x^2 > 0. \]
```

```
\dots følgende diagram:
```

```
\[
```

```
\xymatrix@=5mm{
```

```
A \ar[r] \ar[d] & B
```

```
\ C \ar[ru] }
```

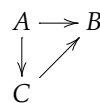
```
\]
```

```
Diagrammet viser\dots
```

For $x \neq 0$ gælder at

$$x^2 > 0.$$

...følgende diagram:



Diagrammet viser...

Sidebemærkning 4.5. Teknisk set bør kommatringstegnene som anvendes indenfor matematik-mode være de samme som anvendes i teksten, dvs. i samme font som i teksten. Der vil dog i de fleste tilfælde ikke være noget problem i bare at skrive » $[x=0 .]$ «. Det er først i det tilfælde at man anvender to forskellige fonte til teksten og matematikken at man begynder at kunne se en forskel mellem f.eks. de anvendte kommaer i teksten og i matematikken (tjek Palatino versus Euler-matematikfont).

4.2 Environments til fremhævet matematik

4.2.1 Alignment

alignment Som titlen antyder handler hele dette afsnit om begrebet *alignment*, altså om at stille formler op således at visse dele flugter under hinanden. Vi formoder at man anvender *amsmath* som minimum *amsmath* pakken. De fleste af eksemplerne 4.5–4.18 er lavet via en eller anden form for alignment.

Alignmentsøjle

align Vi har i kapitel 3 set hvordan man kan anvende *align* til at justere f.eks. =’erne i en udregning. Lad os nu definere begrebet *alignmentsøjle*.³ Et udtryk som

Eksempel 4.20

```
\begin{align*}
```

```
\dbxq &= \dbx\ \
```

```
\dbx &= \dbx
```

```
\end{align*}
```



alignmentsøjle

kan man opfatte som en tabel med to søjler. Den venstre søjle er det lysegrå område mens den højre består af det mørkegrå område. Et sådant sæt af en venstresøjle og samhörørende højresøjle definerer vi nu som en *alignmentsøjle*.

³ Dette er et hjemmestrikket udtryk som jeg har fundet på, det skulle gerne kunne hjælpe folk med at forstå, hvor de (potentielt mange) forskellige &-tegn skal placeres.

Align

Hvad nu hvis man ønsker flere alignmentsøjler? Det er vel oplagt, *adskil hver alignmentsøjle med et &*. Altså hvis man i *align* gerne have to søjler, anvendes

align

```
\begin{align*}
\delta x &= \delta x & \delta x &= \delta x \\
\delta x &= \delta x & \delta x &= \delta x
\end{align*}
```



Eksempel 4.21

Af dette kan man udlede at antallet af &-er i hver linie (række i tabellen) er

$$2 \times \text{antal alignmentsøjler} - 1$$

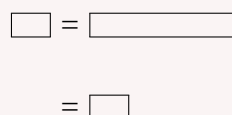
altså n &-er til adskillelsen i alignmentsøjlerne og $n - 1$ &-er til at skille søjlerne ad.

Sidebemærkning 4.6. Bemærk at i alle matematikkonstruktioner hvori $\backslash\backslash$ anvendes til at skifte linie/række, da kan man anvende følgende syntaks til at forøge linieafstanden, hvilket til tider kan være ret anvendeligt.

$\backslash\backslash[\langle \text{længde} \rangle]$

Et eksempel kan ses herunder:

```
\begin{align*}
\delta x &= \delta x[2cm] \backslash\backslash[5mm] \\
&= \delta x
\end{align*}
```



Eksempel 4.22

Align

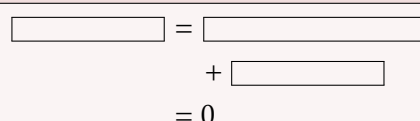
[Eksempel 4.21](#) viser at *align* sagtens kan finde ud af at have to (eller flere) alignmentsøjler, den finder selv ud af hvor mange søjler man ønsker osv. Man kan også se at *align* selv indsætter noget mellemrum mellem alignmentsøjlerne når der er flere end en søjle. Det er ikke altid man er tilfreds med denne spacing, se *alignat* på næste side for et alternativ.

align

Tip 4.7. I større fremhævede udtryk over flere linier kan det være en god ide at placere de $\backslash\backslash$ -er, som skifter til en ny linie i udtrykket, på en linie helt for sig selv. Så er det nemmere at skelne mellem de forskellige linier i udtrykket. Vi har ikke gjort det i bogen, da vi gerne vil spare lidt plads i outputtet.

Tip 4.8. Normalt anbefales det at man justerer til venstre for to de relationer man ønsker at aligne ved. Man kan dog sagtens justere til højre i stedet, bare man tager sine forholdsregler. Bemærk brugen af $\{ \}$ i følgende eksempel:

```
\begin{align*}
\delta x[2cm] &= \{ \} & \delta x[3cm] & \backslash\backslash \\
& & & + \delta x[2cm] & \backslash\backslash \\
&= \{ \} & & & 0
\end{align*}
```



Eksempel 4.23

Altså skal der $\{ \}$ ind mellem relationen (eller hvad det nu er) og &-tegnet, ellers opnår man ikke det rette mellemrum. For en nærmere forklaring se afsnit [E.3](#) på side 409.

4. Matematik på et højere niveau

Tricket kan anvendes til alle alignment konstruktioner fra \mathcal{AMS} .

Alignat

Når man gerne selv vil kontrollere hvilken spacing som kommer mellem alignmentsøjlerne kan man benytte **alignat**. I modsætning til **align** indsætter denne ikke selv ekstra mellemrum mellem søjlerne. Prisen er dog at man bliver nødt til at angive antallet af alignmentsøjler som argument til **alignat**. Syntaksen er:

```
\begin{alignat*}{\langle antal alignmentsøjler \rangle}
\langle almindelig kode som i align \rangle
\end{alignat*}
```

Eksempel 4.21 på foregående side nu med **alignat**.

Eksempel 4.24

```
\begin{alignat*}{2}
\dx &= \dx & \dx &= \dx \\
\dx &= \dx & \dx &= \dx
\end{alignat*}
```

$$\square = \square \quad \square = \square$$

$$\square = \square \quad \square = \square$$

Man ser at der nu ikke er nogen spacing mellem alignmentsøjlerne, den skal man selv indsætte hvis det er nødvendigt (bemærk at man behøver kun gøre det et sted).

Eksempel 4.25

```
\begin{alignat*}{2}
\dx &= \dx & \quad & \dx &= \dx \\
\dx &= \dx & & \dx &= \dx
\end{alignat*}
```

$$\square = \square \quad \square = \square$$

$$\square = \square \quad \square = \square$$

Man kan sagtens anvende konstruktionen uden at indsætte mere luft mellem søjlerne, dette kan være nyttig hvis man har flere relationer man gerne ville justere på.⁴

Eksempel 4.26

```
\begin{alignat*}{2}
\dx &= \dx[6mm] & \quad & \dx \\
\dx &= \dx & & \quad & \quad & \dx
\end{alignat*}
```

$$\square = \square = \square$$

$$\square = \square = \square$$

Man anvender til tider **alignat** til f.eks. at lave konstruktioner som den følgende:

Eksempel 4.27

```
\begin{alignat*}{3}
f(x) &= 0 & \quad & \quad & \text{hvis} \\
& \quad & \quad & f(x^2+1)+3x & \in A \\
\\
g(y) &= 1 & \quad & \quad & \text{hvis} \\
& \quad & \quad & y & \in \{1,2\}
\end{alignat*}
```

$$f(x) = 0 \quad \text{hvis} \quad f(x^2 + 1) + 3x \in A$$

$$g(y) = 1 \quad \text{hvis} \quad y \in \{1, 2\}$$

Her har vi alignet ved »=«, »hvis« og »∈«, plus indsat noget mellemrum via `\quad`.

Et yndet eksempel på brugen af **alignat** er inden for optimering eller heltalsprogrammering, hvor man ofte har en opstilling à la⁵

⁴ Bemærk hvordan en af kasserne i midten er større end den under.

⁵ U.bb. betyder vist noget med *under bi-betingelserne*, det er en standard forkortelse især anvendt indenfor matematisk økonomi og operationsanalyse.

Flalign

```
\begin{alignat}{2}
& \text{minim\`er}\quad& \dbx[2cm]\backslash
& \text{u.bb.} & & \dbx \\
& & & \dbx \\
& & & \dbx \\
& & & \dots
\end{alignat}
```

minimér	<input type="text"/>	(4.1)
u.bb.	<input type="text"/>	(4.2)
	<input type="text"/>	(4.3)
	<input type="text"/>	(4.4)
...		(4.5)

**Eksempel
4.28**

Bemærk brugen af & & & i de senere linier, pga. de tomme celler i første alignment søjle og fordi vi ikke har noget i venstre side af anden alignment søjle. Vi skal senere se nærmere på dette eksempel og komme med ideer til yderligere visuel optimering.

En anden anvendelse af *alignat* er til at lave konstruktioner med »↑« mellem. Se afsnittet *Ensbetyderpile mellem linierne i en beregning* på side 105.

Flalign

Environmentet *flalign* anvendes sjældent, men har en interessant feature. Når man har flere alignment søjler presses teksten helt ud til højre og venstre margin.

flalign

```
\begin{flalign*}
& \dbx & \dbx & \dbx \\
\end{flalign*}
```

<input type="text"/> = <input type="text"/>	<input type="text"/> = <input type="text"/>
---	---

**Eksempel
4.29**

Dette kan anvendes til notations misbrug:

```
\begin{flalign}
(\textup{A}) & \dbx & \dbx & \dbx \\
& & \delta=0 & \nonumber \\
\end{flalign}
```

(A)	<input type="text"/> = <input type="text"/>	(4.6)
	= 0	

**Eksempel
4.30**

Men man bør tænke sig godt om før man begynder på den slags.

En måske lidt mere brugbar anvendelse kan ses på følgende eksempel:

```
\begin{flalign*}
& \dbx[2cm] & \delta= & \dbx[2cm] & \dbx \\
\text{sam} & \dbx & \delta= & \dbx & \dbx \\
\end{flalign*}
```

	<input type="text"/>	=	<input type="text"/>
sam	<input type="text"/>	=	<input type="text"/>

**Eksempel
4.31**

Se desuden Voß (2008).

Kommentarer i alignment matematik

Med makroen `\intertext{⟨tekst⟩}` kan man afbryde alignment, indsætte noget tekst og genoptage alignment igen. Makroen må kun anvendes *efter* `\backslash` eller `\backslash*`⁶ og giver kun mening at bruge i forbindelse med alignment konstruktioner. Bemærk at kommentaren må gerne være en længere tekst – dog ikke alt for lang. Skulle man have brug for at starte et nyt afsnit inden i `\intertext` så skal man lige bruge `\indent` for at få den indenteret.

`\intertext`

⁶ `\backslash*` forklares senere, se afsnit 4.2.6 på side 98.

4. Matematik på et højere niveau

Eksempel 4.32

```

Noget tekst,
\begin{align*}
\quad \delta x[1cm] &= \delta x[3cm], \\
&\intertext{en kommentar}
\quad \delta x[5mm] &\rightarrow 0.
\end{align*}

```

Noget tekst,

$$\delta x[1cm] = \delta x[3cm],$$

en kommentar

$$\delta x[5mm] \rightarrow 0.$$

Hvis man synes `\intertext` indsætter for meget vertikal spacing omkring kommentaren kan man anvende `\shortintertext` fra pakken `mathtools` (sammenlign [Eksempel 4.32](#) og [4.33](#)). `\shortintertext` er mest beregnet til korte tekster.

`\shortintertext`
`mathtools`

Eksempel 4.33



```

\usepackage{mathtools}
----- slut preamble -----
Noget tekst,
\begin{align*}
\quad \delta x[1cm] &= \delta x[3cm], \\
&\shortintertext{og}
\quad \delta x[5mm] &\rightarrow 0.
\end{align*}

```

Noget tekst,

$$\delta x[1cm] = \delta x[3cm],$$

og

$$\delta x[5mm] \rightarrow 0.$$

Bemærk for øvrigt at disse to makroer *ikke* ignorerer eventuelle mellemrum i starten af argumentet.

Sidebemærkning 4.9. L^AT_EX må faktisk gerne dele et fremhævet matematisk udtryk før en `\intertext` eller en `\shortintertext`. Man kan altså anvende dette til at få opdelt et større udtryk som kommer lidt for tæt på et sideskift. Se også afsnittet »[Sideskift i fremhævet matematik](#)« på side 98.

4.2.2 Centreret flerlinjet matematik

Til tider har man ikke lyst til at aligne et par formler, men bare have dem samlet som en enhed. Til dette kan man i stedet anvende *gather*. Da der her ikke er tale om alignment, skal man *ikke* have `&`'er med, i stedet deler man bare linien med `\`.

`gather`
`\`

Eksempel 4.34

```

\begin{gather}
\delta x[2cm] \\
\delta x\{nonumber\} \\
\delta x[4cm]
\end{gather}

```

$$\delta x[2cm] \tag{4.7}$$
$$\delta x[4cm] \tag{4.8}$$

4.2.3 Multline

`multiline`
`gather`

Environmentet *multiline* har samme syntaks som *gather* men en lidt anderledes opførelse:

Eksempel 4.35

```

\begin{multiline}
\delta x[4cm] \\
\delta x[4cm]
\end{multiline}

```

$$\delta x[4cm]$$
$$\delta x[4cm] \tag{4.9}$$

multiline vil venstrejustere første linie, højrejustere sidste linie og centrere alle andre. Desuden vil formelnummeret kun være at finde efter sidste formellinie.⁷ Pakker man

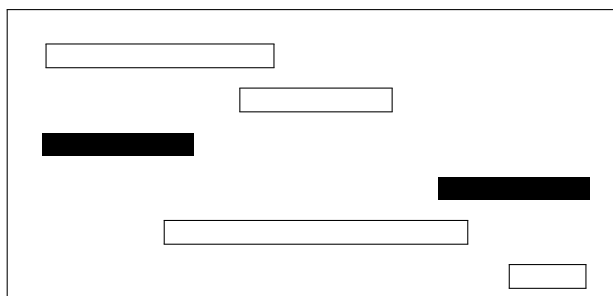
⁷ Med mindre man har formelnumrene i venstre side hvorved det så placeres ved første linie.

Varianter

en linie (undtagen `\`) ind i `\shoveleft` eller `\shoveright` kan man tvinge linien til venstre eller højre.

`\shoveleft`
`\shoveright`

```
\newcommand\sfr{%
  \rule[-0.6mm]{2cm}{3mm}}
\begin{multline*}
  \dbx[3cm] \\\
  \dbx[2cm]\\\
  \shoveleft{\sfr}\\\
  \shoveright{\sfr} \\\
  \dbx[4cm]\\\
  \dbx[1cm]
\end{multline*}
```



Eksempel
4.36

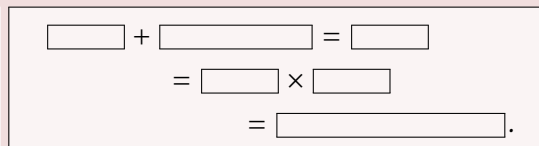
Man bør *ikke* anvende `\shoveleft` i første linie ej heller `\shoveright` i sidste linie, da det kan give underlige resultater.



Farlig (læs teksten) 4.10. Det anbefales heller ikke at anvende *multline* til at dele et udtryk med flere (≥ 2) relationer (f.eks. $>$ eller $=$). Her er det bedre at anvende *align* og i stedet aligne relationerne under hinanden.

Man bør også undgå at anvende *multline* ved alt for mange linier, samt sørge for at tilpasse længden på de enkelte udtryk når man har valgt at anvende *multline*. Sådan skal man altså *ikke* gøre:

```
\begin{multline*}
  \dbx[1cm]+\dbx[2cm] = \dbx[1cm]\\\
  =\dbx[1cm]\times\dbx[1cm]\\\
  =\dbx[3cm].
\end{multline*}
```



Eksempel
4.37

Det kommer til at se alt for rodet ud, især når tekstbredden er større end den vi har anvendt i [Eksempel 4.37](#).

Generelt vil jeg normalt *ikke* anbefale brugen af *multline*, jeg anvender normalt *align* og `\MoveEqLeft` i stedet.

4.2.4 Varianter

Alle `amsmath`'s environments til fremhævet matematik (undtagen *multline*⁸) har en speciel intern version som slet ikke giver formel numre og så *kun* kan anvendes inden i andre environments.

Split

Environmentet *split* har samme syntaks som *align*, men den understøtter kun *én* alignment søjle. Den anvendes ofte sammen med enten *equation* eller *gather* når man vil have en justeret udregning men kun ét formelnummer for hele udregningen.

split

⁸ Se dog Høgholm (2008b) (mathtools-pakken).

4. Matematik på et højere niveau

Eksempel 4.38

```
\begin{gather}
\begin{split}
\mathrm{d}x &= \mathrm{d}x \\
&= \mathrm{d}x \\
&= \mathrm{d}x \\
&= \mathrm{d}x
\end{split}
\end{gather}
```

$$\begin{aligned}
 &= \\
 &= \\
 &= \\
 &=
 \end{aligned}
 \tag{4.10}$$

Man bemærker at formelnummeret placeres vertikalt centreret i forhold til materialet, der kigges ikke efter linier men placeres vertikalt centreret efter hele formelen som helhed.

Aligned

aligned Omvendt svarer *aligned* til den fulde udgave af *align*, men unummereret til brug inden i andre environments.

Eksempel 4.39

```
\begin{gather}
\mathrm{d}x[2\mathrm{cm}] = \mathrm{d}x[2\mathrm{cm}], \\
\begin{aligned}
\mathrm{d}x &= \mathrm{d}x, & \mathrm{d}x &= \mathrm{d}x, \\
\mathrm{d}x &= \mathrm{d}x, & \mathrm{d}x &= \mathrm{d}x.
\end{aligned}
\end{gather}
```

$$\begin{aligned}
 &= , \\
 &= , \\
 &= .
 \end{aligned}
 \tag{4.11}$$

$$\begin{aligned}
 &= , & &= , \\
 &= , & &= .
 \end{aligned}
 \tag{4.12}$$

Man vil sikkert genkende denne opstilling fra [Eksempel 4.18 på side 89](#), som også kan laves via *alignedat*.

Noget andet man kan lave med *aligned* er:

Eksempel 4.40

```
\begin{equation*}
\left. \begin{aligned}
\mathrm{d}x &= \mathrm{d}x \\
&= \mathrm{d}x \\
&= \mathrm{d}x
\end{aligned} \right\} \Rightarrow \left\{ \begin{aligned}
&= \\
&= \\
&=
\end{aligned} \right. \\
\quad \rightarrow \\
\left\{ \begin{aligned}
\mathrm{d}x &= \mathrm{d}x \\
&= \mathrm{d}x
\end{aligned} \right.
\end{equation*}
```

$$\left. \begin{aligned}
 &= \\
 &= \\
 &=
 \end{aligned} \right\} \Rightarrow \left\{ \begin{aligned}
 &= \\
 &=
 \end{aligned} \right.$$

Environmentet kan faktisk også tage et argument som angiver om der skal justeres på første eller sidste række eller der som her bare skal vertikalt centreret (test f.eks. `\begin{aligned}[t]`).

Alignedat

alignedat Environmentet *alignedat* svarer til *aligned* som *alignat* svarer til *align*.

Eksempel 4.41

```
\begin{gather}
\mathrm{d}x[2\mathrm{cm}] = \mathrm{d}x[2\mathrm{cm}], \\
\begin{alignedat}{2}
\mathrm{d}x &= \mathrm{d}x, & \mathrm{d}x &= \mathrm{d}x, \\
\mathrm{d}x &= \mathrm{d}x, & \mathrm{d}x &= \mathrm{d}x.
\end{alignedat}
\end{gather}
```

$$\begin{aligned}
 &= , \\
 &= , & &= , \\
 &= , & &= .
 \end{aligned}
 \tag{4.13}$$

$$\begin{aligned}
 &= , & &= , \\
 &= , & &= .
 \end{aligned}
 \tag{4.14}$$

Gathered

gathered er bare *gather* pendanten til *aligned*, anvendes ofte til at samle formler under et enkelt formelnummer. *gathered*

```
\begin{equation}
\begin{gathered}
\text{\dbx[3cm]} = \text{\dbx[2cm]}, \\
\text{\dbx+}\text{\dbx}-\text{\dbx}\otimes\text{\dbx} \\
\text{\dbx}\times\text{\dbx}.
\end{gathered}
\end{equation}
```

$$\boxed{} = \boxed{}, \quad (4.15)$$

$$\boxed{} + \boxed{} - \boxed{} \otimes \boxed{} \times \boxed{}.$$

Eksempel
4.42

Man kan altså sige at *gathered* er en pendant til *split*.

Pakken *mathtools* har desuden en *lgathered* version som kan være ret interessant, se afsnit 4.5 på side 102. Samme pakke giver også et *multlined* environment, her bør man dog lige læse Høgholm (2008b) før brug. *mathtools*
lgathered
multlined

4.2.5 Eksemplerne 4.5 til 4.18 på side 87–89

For god ordens skyld er her hvordan eksemplerne 4.5 til 4.18 på side 87–89 er lavet, eller rettere med hvilken konstruktion.

Eksempel 4.5 på side 87.

Almindelig *align*-konstruktion med alignment til venstre for relationer.

Eksempel 4.6 på side 87.

Samme type med *align*, her er alignet yderst til venstre, og hver af de efterfølgende linier starter med $\&\!\!\quad$.

```
\begin{align*}
&\text{\dbx[5cm]} = \text{\dbx[4cm]} \\
&\&\!\!\quad = \text{\dbx[6cm]} \\
&\&\!\!\quad = \text{\dbx[5cm]}\!, .
\end{align*}
```

Eksempel
4.43

Det er dog meget nemmere at opnå via $\!\!\MoveEqLeft$, afsnittet *Justering af indtryk ved brudte formler* på side 105.

Eksempel 4.7 på side 87.

Samme som ovenfor, kun indholdet er anderledes.

Eksempel 4.8 på side 88.

Her er der som i 4.5 på side 87 alignet til venstre for relationerne. Linierne som starter med $\!+\!<$ og med $\!-\!<$ er så starter med $\&\!\!\quad$. Igen nemmere med $\!\!\MoveEqLeft$.

Eksempel 4.9 på side 88.

Her er alignet til *højre* for lighedstegnet, se Tip 4.8 på side 91.

Eksempel 4.10 på side 88.

Samme metode som i Eksempel 4.6 på side 87.

Eksempel 4.11 på side 88.

Samme metode som i Eksempel 4.7 på side 87.

Eksempel 4.12 på side 88.

Er lavet via *multline*.

Eksempel 4.13 på side 88.

En kombination af *align* og $\!\!\hphantom$. Se Eksempel 4.46 på side 99.

4. Matematik på et højere niveau

Eksempel 4.14 på side 89.

Igen *align*. Her er der ikke tale om en decideret udregning, bare en samling af formler.

Eksempel 4.15 på side 89.

En simpel konstruktion via *gather*.

Eksempel 4.17 på side 89.

Endnu en gang *align*, her har vi anvendt to alignmentsøjler for at kunne adskille formlerne og sætte dem visuelt pænt op.

Eksempel 4.18 på side 89.

Igen *gather*, her kombineret med varianten *alignedat*.

4.2.6 Sideskift i fremhævet matematik

Det at dele et flerliniet fremhævet matematisk udtryk over to sider, anses generelt som værende dårlig skik. Med de environments som er beskrevet her vil L^AT_EX også selv sørge for at holde dem samlet på én side, hvilket kan føre til sider hvor der er meget store mængder luft mellem de forskellige dele af teksten. Vi har dog allerede set at man sagtens kan få sideskift ind i billedet via `\intertext` eller `\shortintertext`. Her vil man nemlig få noget tekst (kommentaren) til at starte siden med.

Det er dog naturligvis muligt at få eksempelvis en *align* til at blive delt over to sider. Man bør dog kun gøre brug af det, hvis man anvender dobbeltsidet opsætning og delingen foregår fra en venstre side til en højre side, det går ud over overblikket hvis læseren skal bladre for at finde resten af udtrykket.

Man har to konstruktioner til sin rådighed:

`\displaybreak` **Pr. håndkraft:** Her indsætter man `\displaybreak[⟨num⟩]` før den `\` man vil tillade et sideskift ved. Makroen kan tage et valgfrit argument: `[⟨num⟩]`, hvor $⟨num⟩ \in \{0, \dots, 4\}$. `[0]` betyder: »Det er tilladt at bryde her«, men den opfordrer ikke til det. Jo større tal som anvendes jo mere opfordres sideombrydningen til at være lige her. `\displaybreak` er det samme som `\displaybreak[4]` og tvinger bruddet til at være lige her.

`\allowdisplaybreaks` **Automatisk valg:** Her placerer man `\allowdisplaybreaks` enten i preamblen (hvor efter det så gælder generelt for alle fremhævede formler⁹) eller indenfor en

Eksempel 4.44

```
\begingroup
\allowdisplaybreak
% matematisk konstruktion
\endgroup
```

eller en

Eksempel 4.45

```
{
\allowdisplaybreak
% matematisk konstruktion
}
```

konstruktion (hvor det så kun gælder her). L^AT_EX vil så selv finde ud af hvor den vil dele udtrykket. Er der en linie efter hvilken man helst *ikke* ønsker et sideskift kan man afslutte den med `*` (kan også anvendes generelt til at gennemtvinge et lineskift, men undertrykke muligheden for sideskift netop på dette sted) i stedet for `\`.

`\allowdisplaybreaks` tager også et valgfrit argument: 1–4, hvor 1 betyder: »Tillad sideskift i fremhævet matematik, men gør en indsats for at lade være«. Større tal betyder

⁹ Anbefales ikke.

Ser du spøgelse?

bare at den gør mindre og mindre for at undgå sideskift i matematik. `\allowdisplaybreaks` er det samme som `\allowdisplaybreaks[4]`.

Sidebemærkning 4.11. Af tekniske årsager kan man ikke dele materialet inden i en *split*-, *aligned*-, *alignedat*- eller *gathered*-konstruktion over flere sider.

4.3 Ser du spøgelse?

Antag at man på en linie ønsker at indsætte lige så meget blank plads som en given tekst er bred. Dette opnås med et spøgelse – eller rettere med

```
\phantom{<tekst>}
\hphantom{<tekst>}
\vphantom{<tekst>}
```

Første version sørger for at det reserverede mellemrum desuden passer i højden og i dybden (under baseline) svarende til den boks som skulle til for at omkrænse den angivne tekst. Ved `\hphantom` afsættes kun bredden af den angivne tekst, højden og dybden ignoreres. Tilsvarende har `\vphantom` ingen bredde, men kun højde og dybde svarende til førnævnte boks. Makroerne finder automatisk ud af om de er anvendt i eller udenfor matematikmode.

Vi skal her anvende den med hensyn til [Eksempel 4.13 på side 88](#). Man kan opnå disse indrykninger via:

```
\begin{align*}
\dx[1cm] &= \dx\bigl[
\dx[2cm] \ \
&\hphantom{=}=\dx\bigl[
+ \dx[1.5cm]\bigl] \ \
&=0.
\end{align*}
```

$$\boxed{} = \boxed{} \left[\boxed{} + \boxed{} \right] = 0.$$

Eksempel 4.46

Man emulerer altså bredden af den tekst som stod i linien ovenfor, regnet fra alignmentpunktet og frem. Bemærk brugen af `{}` i argumentet til `\hphantom`, uden den ville spacingen omkring lighedstegnet være for lille og derved ville man ikke få den rigtige bredde.¹⁰ Se desuden [Eksempel 4.60 på side 104](#).

4.4 Konfigurering af og leg med formelnummeret

Fjernelse af formelnummer

Næsten alle de formelnummer givende environments understøtter korrekt `\nonumber` (eller dets alias `\notag`). Undtagelserne er *equation* og *eqnarray* som til tider kan give bizzarre resultater i kombination med disse og brugen af `\label`, se [Madsen \(2006\)](#). `\nonumber` indsættes før en eventuel `\` og undertrykker derefter formelnummeret for denne linie.

`\nonumber`
`\notag`

Skribe noget andet i stedet for formelnummeret

I den helt anden grøft finder man `\tag{<tekst>}` og `\tag*{<tekst>}`, som i stedet kan

`\tag`
`\tag*`

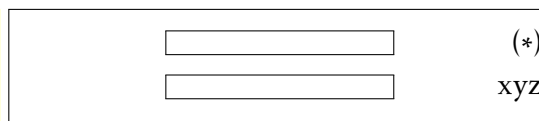
¹⁰ Man kunne også have anvendt `\mathrel{=}` i stedet for `»{=«`.

4. Matematik på et højere niveau

anvendes til at skrive noget helt andet end det normale formelnummer. `\tag` skriver sit argument (behandlet som almindelig tekst) mellem ()-erne¹¹, modsat skriver `\tag*` sit argument helt uden parenteser.

Eksempel 4.47

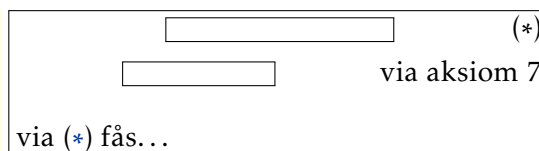
```
\begin{gather}
\dbx[3cm] \tag{*$*}\backslash
\dbx[3cm] \tag*{xyz}
\end{gather}
```



Bemærk brugen af `*$*` for at få `*`-en vertikalt centreret. Blander man `\label` ind i det, vil man faktisk kunne referere til et omdøbt formelnummer.

Eksempel 4.48

```
\begin{gather}
\dbx[3cm] \label{tag*}\tag{*$*}\backslash
\dbx[2cm] \tag*{via aksiom 7}
\end{gather}
via \eqref{tag*} fås\dots
```



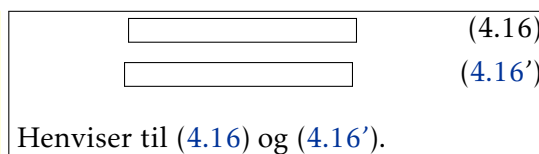
Sidebemærkning 4.12. Det skal bemærkes at dette med at skrive kommentarer som »via aksiom 7« i [Eksempel 4.48](#) anses for dårlig stil på skrift. Det forstyrrer forståelsen af formlen mere end det gavner.

Se eventuelt [Høgholm \(2008b\)](#) for flere ideer vedrørende konfigurering af formelnumre, se også afsnit 4.5, side 104.

Et tredje ting man kunne få brug for er at kunne lave en *mærket* version af en allerede nummereret formel. Dette er såre simpelt, anvend `\ref` og `\label`.

Eksempel 4.49

```
\begin{gather}
\dbx[3cm] \label{442}\backslash
\dbx[3cm] \tag{\ref{442}'}\label{442'}
\end{gather}
Henviser til \eqref{442} og \eqref{442'}.
```



Konfigurering af hvordan formler nummereres

I langt de fleste dokumentklasser er defaultmetoden til formelnummering at nummereringen sker fortløbende: (1), (2), etc. Er man i gang med et længere værk vil det ofte være en fordel at have såkaldt *subordineret* (eller *domineret*) nummering, i.e. f.eks. (`\section`).(`formel`)), hvor formelnummeret starter forfra i hver ny section. Med `amsmath` er dette meget nemt

```
\numberwithin        \numberwithin{\underordnet tæller}{\dominerende tæller}
```

I denne bog (som har kapitler) har vi anvendt

Eksempel 4.50

```
\numberwithin{equation}{chapter}
```

Det burde være oplagt hvad man skulle være hvis man anvender en klasse som kun har `\sections`. Det er også muligt at tage `equation` tælleren ud af denne underlæggelse, dvs. gå tilbage til (1), (2), ...

¹¹ Eller hvordan man nu har valgt at konfigurere dem. Se evt. `mathtools`-pakken.

Ét formelnummer til udregning

Sidebemærkning 4.13. Bemærk at hvis man har brug for at gå fra kapiteldominans, til afsnitsdominans og tilbage igen, da har man brug for at kunne få equation-tælleren ud fra afsnitsdominansen igen. Anvender man *memoir*-klassen, kan man tage et kig på `\counterwithout` makroen (også tilgængelig via pakken `chngctr`).

`\counterwithout`

Ét formelnummer til udregning

Her kombinerer man to environments, normalt *split* eller *aligned* (som jo ikke giver numre) sammen med *equation* eller *gather*. Se f.eks. [Eksempel 4.38 på side 96](#) eller [Eksempel 4.39 på side 96](#).

Undernummering af formelnummer

Til tider er det rart at lave en yderligere undernummering af formelnummeret via bogstaver, til dette anvendes *subequations*-environmentet¹² som bare pakkes udenom de formler (og tilhørende tekst) man ønsker skal være inden for denne undernummering. I det følgende eksempel bedes man bemærke placeringen og effekten af de forskellige `\label`'s.

subequations

```
\begin{subequations}
  \label{mainnummer}
  \begin{align}
    \dbx &= \dbx \label{suba} \\
    \dbx &= \dbx \label{subb}
  \end{align}
  Hovednummer: \eqref{mainnummer},
  undernumre: \eqref{suba} og \eqref{subb}.
\end{subequations}
```

$$\square = \square \quad (4.17a)$$

$$\square = \square \quad (4.17b)$$

Hovednummer: (4.17), undernumre: (4.17a) og (4.17b).

Eksempel 4.51



Man har altså mulighed for at referere til det overordnede nummer samt de enkelte elementer for dem selv. Alle nummererede formler inden for *subequations* vil så blive skrevet på denne måde.

Bemærk at det er, af tekniske årsager, ikke muligt f.eks. at ændre nummereringsmetoden midt i f.eks. en *align*.

Flytte lidt på formelnummeret

En af de gode ting ved \mathcal{AMS} -konstruktionerne er at de sørger for at holde øje med om de kommer for tæt på formelnummeret, så flytter de formelnummeret lidt ned (hvis man har placeret formelnumre i højre side).

Til tider kan placeringen dog se lidt grim ud. Man kan så i stedet skubbe lidt ved placeringen via makroen `\raisetag{<længde>}`. Anvend en negativ længde for at flytte nummeret længere ned.

`\raisetag`

```
\begin{align}
  \dbx[6cm] \\
  \dbx[6cm]
  \raisetag{-6pt}
\end{align}
```

$$\square \quad (4.18)$$

$$\square \quad (4.19)$$

Eksempel 4.52

¹² Del af `amsmath`.

4. Matematik på et højere niveau

Vis kun refererede formler

Se afsnittet »*Hvordan viser man kun refererede formler?*« på side 104.

4.5 \mathcal{AMS} udvidelsen mathtools

13

mathtools Pakken *mathtools* af Morten Høgholm en hel del ekstra ting i forbindelse med *amsmath*. Vi vil her præsentere nogle stykker og ellers henvide til [Høgholm \(2008b\)](#) for mere information (manualen er under alle omstændigheder værd at læse).¹⁴

Justering vedrørende grænser på operatorer

Til tider har forfattere for vane at skrive nogle grænser på store operatorer som er for brede eller som kan forstyrre alignment. Dette kan man så fikse med

`\smashoperator` `\smashoperator[⟨position⟩]{⟨operator incl. grænser⟩}`

hvor *⟨position⟩* er *r*, *l* eller *rl* (standard). Ideen med `\smashoperator` er at man får ignoreret bredden af øvre og nedre grænse på en stor operator. Positionerne angiver så om man ignorerer kun i venstre, højre eller begge sider.

Eksempel 4.53



```
\usepackage{mathtools}
----- slut preamble -----
\begin{equation*}
X\sum_{x\mid x^2+y^2=1}X \quad \quad
X\smashoperator{\sum_{x\mid x^2+y^2=1}}X \quad \quad
X=\smashoperator[r]{\sum_{n=1}^{12345678}}X
\end{equation*}
```

$$X \sum_{x|x^2+y^2=1} X \quad X \smashoperator{\sum_{x|x^2+y^2=1}} X \quad X = \smashoperator[r]{\sum_{n=1}^{12345678}} X$$

Eksempel 4.53 skal fortolkes som: Første sum er det man normalt får, bemærk at de to *X*'er ikke overlapper med grænsen. I den anden sum har vi anvendt `\smashoperator` som sørger for at bredden af grænsen ignoreres mht. placeringen af *X*'erne. Tredje sum viser så hvordan man kan lave ignoreringen kun til højre for summen.

Makroen er faktisk også ret smart til at aligne 'brede' summer.

Eksempel 4.54



```
\usepackage{mathtools}
----- slut preamble -----
\begin{align*}
&\&\smashoperator[l]{\sum_{x\mid x^2>3}}\dots\backslash
&\&\smashoperator[l]{\sum_{y\in B}} \dots\backslash
&\&\smashoperator[l]{\sum_{x_1+x_2+x_3=2}} \dots
\end{align*}
```

$$\begin{array}{c} \sum_{x|x^2>3} \dots \\ \sum_{y\in B} \dots \\ \sum_{x_1+x_2+x_3=2} \dots \end{array}$$

Det overlades til læseren at se hvad der ville ske hvis man fjernede `\smashoperator` fra Eksempel 4.54.

¹³ FixMe Note: husk lige at tjekke alle referencer til disse

¹⁴ Bemærk iøvrigt at *mathtools* automatisk loader *amsmath* hvis man ikke allerede havde gjort det i forvejen. Det betyder at man kan nøjes med at aktivere *mathtools* i stedet for både *mathtools* og *amsmath*.

Intelligente afgrænserfunktioner

Intelligente afgrænserfunktioner

Mange har prøvet at lave en makro til at lave f.eks. $|x|$ med, ofte via en konstruktion lignende

```
\newcommand{\abs}[1]{\lvert#1\rvert}
```

Men hvad så når man får brug for en som er lidt større. Så ændrer mange definitionen så `\left` og `\right` også indgår. Men det er jo heller ikke godt, se afsnit 3.3. Lav i stedet `\abs` og lignende med

```
\DeclarePairedDelimiter{\makronavn}{\left delim}{\right delim}
```

`\DeclarePaired-
Delimiter`

Man vil så anvende:

```
\DeclarePairedDelimiter{\abs}{\lvert}{\rvert}
```

**Eksempel
4.55**



Så får man automatisk også `\abs*`, som er versionen med `\left` og `\right`, mens `\abs` nu kan tage `\big`, `\Big`, `\bigg` eller `\Bigg` som valgfrit argument som så bliver anvendt til manuelt at skalere med.¹⁵ Se [Eksempel 4.56](#).

```
\usepackage{mathtools}
\DeclarePairedDelimiter{\abs}{\lvert}{\rvert}
\DeclarePairedDelimiter{\inner}{\langle}{\rangle}
```

**Eksempel
4.56**



— slut preamble —

```
\begin{equation*}
\abs{x}+\abs*{\frac{a}{b}}+\inner[\Big]{\sum_i f(x_i),\sum_j g(y_j)}=2
\end{equation*}
```

$$|x| + \left| \frac{a}{b} \right| + \left\langle \sum_i f(x_i), \sum_j g(y_j) \right\rangle = 2$$

Vedr. `\inner` kommandoen, så kan det være en fordel at man laver makroen til at kunne tage to argumenter i stedet for kun et, dette kan gøre makroen mere fleksibel da man kan gemme f.eks. komma mellem argumenterne inden i makroen, og så nemt senere skifte det til eksempelvis semi-kolon. I 2010 udvidede vi `mathtools` således at vi fik en generalisering af `\DeclarePairedDelimiter`. Med denne kan vi nu lave `\inner` via:

```
\DeclarePairedDelimiterX{\inner}[2]{\langle}{\rangle}{#1,#2}
\inner[\big]{a}{b}
```

**Eksempel
4.57**

Eller f.eks. den såkaldte *bra-ket*:

```
\DeclarePairedDelimiterX{\braket}[2]{\langle}{\rangle}{#1 \delimsize\vert #2}
```

**Eksempel
4.58**

`\delimsize` justeres så automatisk alt efter om det er `\braket*{a}{b}` eller `\braket[\big]{a}{b}` man anvender.

¹⁵ Internt i `\DeclarePairedDelimiter` bliver disse `\big`'er automatisk lavet om til `\bigl/r`-varianterne.

4. Matematik på et højere niveau

Hvordan viser man kun refererede formler?

Med `mathtools` selvfølgelig. Ideen er at man sørger for at nummerere *alle formler!* – og husker *kun* at anvende \mathcal{AMS} -environments til fremhævede formler.

Eksempel 4.59



```
\usepackage{mathtools}
\mathtoolsset{showonlyrefs,showmanualtags}

----- slut preamble -----

\begin{align}
a&=b \ \label{eq:1} \\
c&=d \ \label{eq:2}
\end{align}
Vi referer kun den sidste \eqref{eq:2}.
```

$$\begin{aligned}
 a &= b \\
 c &= d \tag{1}
 \end{aligned}$$

Vi referer kun den sidste (1).

Metoden kan også anvendes når man som forsker er ved at forberede en artikel og så af forlaget bliver bedt om kun at nummerere de formler som der faktisk henvises til i artiklen. I nyere versioner af `mathtools` finder man også `\noeqlref{<liste af labels>}` som kan anvendes til at aktivere formelnummeret til nogle formler man ikke direkte har henvist til.

`\noeqlref`

Sidebemærkning 4.14. Det har desværre vist sig at der er en lille hage ved brugen af denne feature. Hvis en formel kommer tæt på formelnummeret så vil `amsmath` normalt enten flytte formlen lidt eller flytte formelnummeret. Hvis man anvender `showonlyrefs` så virker denne feature desværre ikke mere! Vi har endnu ikke fundet en løsning på dette problem uden at skulle skrive store dele af `amsmath` om (hvilket vi pt. ikke har lyst til).

lgathered

Vi har allerede nævnet at [Eksempel 4.13 på side 88](#) kan fremstilles via `\phantom`. Her vil vi anvende *lgathered* til at lave den på en lidt anden måde.

`lgathered`

Eksempel 4.60



```
\usepackage{mathtools}

----- slut preamble -----

\begin{align*}
\ltx &= \ltx + \Bigl[ \!
\begin{lgathered}[t]
\ltx[2cm] + \ltx \ \
\ } + \ltx[2cm] \ \
\ } + \ltx[3cm] \ \Bigl]
\end{lgathered}
\ \
&= \ltx[2cm]
\end{align*}
```

$$\begin{aligned}
 \square &= \square + \left[\square + \square \right. \\
 &\quad + \square \\
 &\quad \left. + \square \right] \\
 &= \square
 \end{aligned}$$

Vi bliver nødt til at tage »]« med ind i konstruktionen ellers kommer den ikke på den korrekte linie, og {}-parret er for at få det korrekte mellemrum omkring +-erne.

Resultatet af [Eksempel 4.60](#) kan også fås via lidt andre metoder.

Overlappe alignment

Tidligere blev makroen `\mathclap` omtalt (se [Eksempel 3.40 på side 64](#)). Vi har også nogle som ignorerer mod højre eller venstre, forstået på den måde at indholdet skrives som normalt, men $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ afsætter ikke plads til det. Dette kan være anvendeligt i visse alignment situationer:

Ensbetyderpile mellem linierne i en beregning

$$\begin{array}{l} \boxed{} \\ \leq \boxed{} \\ = \boxed{} \end{array} = \boxed{} \quad \begin{array}{l} \forall \boxed{} \\ \forall \boxed{} \end{array}$$

Eksempel
4.61

Konstruktionen er lavet via

```
\begin{alignat*}{2}
\dbx[2cm] & = \mathrlap{\dbx[4cm]} & \\
& \leq \dbx[2cm] & \quad & \forall \dbx \\
& = \dbx & & \forall \dbx
\end{alignat*}
```

Eksempel
4.62

bemærk `\mathrlap`.

Ensbetyderpile mellem linierne i en beregning

En anden ting man kan bruge *alignat* til er når man gerne vil lave udregninger hvor der er ensbetydende- eller medførerpile mellem. Dette kan man lave på en smart måde via makroen `\ArrowBetweenLines` fra `mathtools`.¹⁶ Anvendelsen illustreres via følgende eksempel. Bemærk forskellen mellem brugen af `\ArrowBetweenLines` og `\ArrowBetweenLines*`.

```
\begin{alignat}{2}
& a=b \\
\ArrowBetweenLines
& b=c
\end{alignat}
\begin{alignat}{2}
c=1 \\
\ArrowBetweenLines*[\Downarrow]
a=1
\end{alignat}
```

$a = b$	(4.20)
\Updownarrow	
$b = c$	(4.21)
$c = 1$	(4.22)
\Downarrow	
$a = 1$	(4.23)

Eksempel
4.63

I første del af eksemplet skal man bemærke brugen af `&&` foran hver normal linie for at få alignment i anden alignmentsøjle, thi første søjle anvendes til pilene. I anden del er pilen i højre side og her behøver man ikke nogen ekstra `&&`. Bemærk at det kun er `\ArrowBetweenLines*` som giver pile i højre side. Observer desuden at konstruktionen ikke forstyrrer afstanden mellem linierne mere end højest nødvendigt.

Bemærk at `\ArrowBetweenLines*` bør *kun* anvendes i konstruktioner som har en enkelt alignment søjle. Versionen uden stjerne (som placerer pilen til venstre) kan anvendes ved et vilkårligt antal alignmentsøjler.

Justering af indryk ved brudte formler

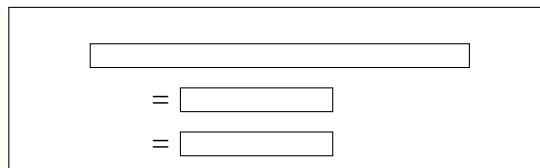
I visse af eksemplerne 4.5 til 4.18 på side 87–89 justerer man helt til venstre, men skal så have indenteret alle efterfølgende linier. Men det er jo træls i længden og alt for meget arbejde, thi man indser hurtigt at dette er det samme som at justere helt til højre og så trække første linie baglæns. Dette gør vi med `\MoveEqLeft` som så skal anvendes som *starten* på første linie *i stedet for* »&<«:

¹⁶ Koden var tidligere en del af bogen.

4. Matematik på et højere niveau

Eksempel 4.64

```
\begin{align*}
\MoveEqLeft \dbx[5cm] \\\
& = \dbx[2cm] \\\
& = \dbx[2cm]
\end{align*}
```

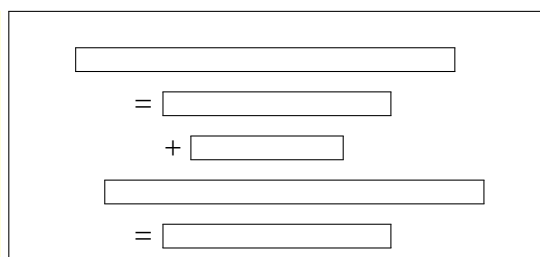


`\MoveEqLeft` vil ikke bare indsætte et »&« i starten af linien, men vil også trække linien mod venstre (der af navnet). Som standard hives linien 2 em baglæns i forhold til justeringspunktet (&). Denne værdi kan ændres via f.eks. `\MoveEqLeft[3.5]` til at trække linien 3.5 em baglæns.

`\MoveEqLeft` vil normalt give illusionen af at de efterfølgende linier er indenteret disse 2 em. Når man så vælger at justere til højre for en relation, så passer illusionen ikke mere. Her skal man huske at anvende [3] i stedet, sammenlign i det næste eksempel.

Eksempel 4.65

```
\begin{align*}
\MoveEqLeft[3] \dbx[5cm] \\\
= {} & \dbx[3cm] \\\
& + \dbx[2cm] \\\
\MoveEqLeft \dbx[5cm] \\\
= {} & \dbx[3cm]
\end{align*}
```



Personligt anvender jeg normalt `\MoveEqLeft` i de fleste konstruktioner jeg retter til for enten forskere eller studerende.

Andre

Se `mathtools`-manualen, [Høgholm \(2008b\)](#) for andre eksempler på hvad denne pakken kan gøre. Den kan f.eks. fikse udseendet af »:=« (således at »:=« bliver centreret omkring den matematiske centerlinie). Pakken indeholder også andre pile konstruktioner a la `\xrightarrow` samt konstruktioner til at forøge afstanden mellem alle linier i et matematik environment.

4.5.1 Pakken *empheq*

Man bør også gøre sig selv den ulejlighed at tage et kig på pakken `empheq` af samme forfatter, denne kan også lave nogle ting med matematik som ellers ikke er muligt, se [Høgholm \(2008a\)](#). Vi vil nøjes med et simpelt eksempel på hvad man kan med denne pakke:

Eksempel 4.66

```
\usepackage{empheq}
\newcommand*\widefbox[1]{\fbox{\hspace{1em}#1\hspace{1em}}}

----- slut preamble -----

\begin{empheq}[box=\widefbox]{align}
\dbx &= \dbx[3cm] \\\
E&=mc^2 + \int_a^a x \, dx
\end{empheq}
\begin{empheq}[left=\empheqlbrace \ ]{alignat=2}
\dbx &= \dbx, & \quad \dbx &= \dbx \\\
\dbx &= \dbx, & \quad \dbx &= \dbx \\\
\dbx &= \dbx, & \quad \dbx &= \dbx
\end{empheq}
```

Displaydelen af eksempel 4.66 findes på næste side

$\square = \square$	(1)
$E = mc^2 + \int_a^a x dx$	(2)
$\left\{ \begin{array}{ll} \square = \square & \square = \square \\ \square = \square & \square = \square \\ \square = \square & \square = \square \end{array} \right.$	(3)
$\left\{ \begin{array}{ll} \square = \square & \square = \square \\ \square = \square & \square = \square \\ \square = \square & \square = \square \end{array} \right.$	(4)
$\left\{ \begin{array}{ll} \square = \square & \square = \square \\ \square = \square & \square = \square \\ \square = \square & \square = \square \end{array} \right.$	(5)

Eksempel 4.66
fortsat

Empheq-pakken er den eneste pakke som er i stand til at kunne gøre dette.

4.6 Diagrammer i L^AT_EX

Indenfor visse grene af naturvidenskab har man brug for at kunne lave diverse (matematiske) diagrammer. Vi vil kigge på X_Y-pic, men man kan også sagtens anvendes METAPOST, eller måske mere relevant – TiKz.

4.6.1 Diagrammer med X_Y-pic og \xymatrix

Der er forskellige måder at gøre dette på men den mest almindelige i dag er at anvende X_Y-pic (i.e. pakken xy). Et diagram er i denne henseende en konstruktion a la

X_Y-pic
xy

<pre> \usepackage{amsmath,amssymb} \usepackage{all,line}{xy} ----- slut preamble ----- \begin{equation*} \xymatrix{A\times A \ar[r]\ar[d] & B \\ B \ar[d] \ll C \ar[r] & D\otimes D} \end{equation*} </pre>	
---	--

Eksempel 4.67

Den bedste kilde til information er [Rose \(1999\)](#), det følgende er små uddrag fra denne.

Den vigtigste syntaks man skal kende er

```

\xymatrix<options>{
<indhold hvor hver indgang i diagrammet er positioneret som en celle i en matrix>
}
\ar<options>[<sekvens>]

```

\xymatrix

\ar

Som man kan se ud fra [Eksempel 4.67](#) vil hver celle indeholde information om hvad der skal stå på denne plads samt information vedrørende eventuelle pile som udspringer her. Argumenterne som man kan angive mellem \xymatrix og den første { vil ikke blive behandlet her, læseren henvises i stedet til [Rose \(1999\)](#), afsnit 3.3, hvor man vil kunne lære at disse options kan anvendes til at forøge/formindske afstanden mellem søjlerne etc. Man kan desuden tage et kig i [Rose og Moore \(1999\)](#) som godt nok kan være en smule vanskelig at forstå, men eksempelvis forklarer hvordan man styrer tykkelsen af pilene.

4. Matematik på et højere niveau

Sidebemærkning 4.15. Som man kan se af [Eksempel 4.67 på forrige side](#) anvender vi visse options til xy-pakken. Man anbefales at anvende `all` og `line`. Hvis man ønsker det kan man desuden få pilehoveder som ligner dem fra Computer Modern fonten, via `cmtip`.

Anvender man desuden `dvips` som option (og anvender `latex + dvips` til output), så får man fine glatte pile. I mange år har der ikke været samme mulighed for glatte pile når man arbejdede med `pdflatex`. Dette blev ændret i 2010 da `xypdf` blev indlemmet i selve xy-pakken. Vi anbefaler derfor at indlæse xy via

Eksempel 4.68

```
\usepackage{ifpdf}% hvis man ikke anvender memoir-klassen
\ifpdf
  \usepackage[all,line]{xy}
\else
  \usepackage[all,line,dvips]{xy}
\fi
```

Vi vil nu i stedet se nærmere på hvordan man laver pile i \Xy-pic .

4.6.2 \Xy-pic pile syntaks

\var Pile i \Xy-pic laves med makroen `\var` som placeres i den matrix indgang hvorfra pilen skal gå. Syntaksen er

$$\text{\var}\langle options \rangle [\langle sekvens \rangle] \langle eventuel tekst \rangle$$

Placering

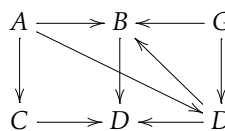
Lad os tage $[\langle sekvens \rangle]$ -delen først. Den består af en serie af bogstaverne `l`, `r`, `u`, og `d`, med åbenlyse referencer til fire engelske ord. Sekvensen skal så beskrive måden hvorpå man vil ville bevæge sig rundt i matricen for at komme derfra, hvor man er lige nu, til den indgang man gerne vil pege hen på.¹⁷ Eksempel (med almindelig matrix indgangsnotation, i.e. matrix indgang række 1, søjle 1 til række 1, søjle 4 ...)

$$\begin{array}{ll} \{11\} \rightarrow \{14\} \iff [rrr] & \{71\} \rightarrow \{63\} \iff [rru] \\ \{35\} \rightarrow \{21\} \iff [ullll] & \{84\} \rightarrow \{95\} \iff [rd] \end{array}$$

Ovenstående skulle gerne betyde at man nu kan forstå syntaksen i følgende eksempel.

Eksempel 4.69

```
\usepackage{amsmath,amssymb}
\usepackage[all,line,dvips]{xy}
----- slut preamble -----
\begin{equation*}
\xymatrix{A \ar[rrd]\ar[d]\ar[r] &
  B \ar[d] & G \ar[d]\ar[l]\
  C \ar[r] & D \ar[l]\ar[ul] D}
\end{equation*}
```



¹⁷ Parseren er intelligent nok til at kunne genkende ækvivalente stier, dvs. rækkefølgen er ikke noget problem.

Pilens udformning

Pilens udformning

Pilens udformning styres af argumentet stående mellem »\ar« og »[\langle sekvens \rangle]« og består af par bestående af en @ og visse argumenter. Her følger et par almindelige, flere kan ses i [Rose \(1999\)](#).

$\langle tom \rangle \Rightarrow A \longrightarrow B$	$@\{->\} \Rightarrow A - - \triangleright B$
$@2\{->\} \Rightarrow A \Longrightarrow B$	$@\{.\>\} \Rightarrow A \cdots \triangleright B$
$@\{ >\} \Rightarrow A \dashrightarrow B$	$@\{^{\{ }>\} \Rightarrow A \overset{\curvearrowright}{\longrightarrow} B$
$@2\{->\} \Rightarrow A \Longleftrightarrow B$	$@\{<->\} \Rightarrow A \longleftrightarrow B$
$@\{:>\} \Rightarrow A \dashrightarrow B$	$@\{\sim>\} \Rightarrow A \rightsquigarrow B$

Eksempel 4.70

Identitetspilen kan altså laves via

```
\usepackage{all,dvips,line}{xy}
----- slut preamble -----
$\xymatrix{A \var@2{-}[r] & \\ B \var@{->}[r] & C}$
```

$$A \Longleftrightarrow B - - \triangleright C$$

Eksempel 4.71



Man kan også lege med krummende pile, igen se [Rose \(1999\)](#), afsnit 1.6 for mere information.

```
\usepackage{all,dvips,line}{xy}
----- slut preamble -----
$\xymatrix{A \var/^{\@{->}}[r] & \\ B \var/_{\@{->}}[r] & C}$
```

$$A \overset{\curvearrowright}{\longrightarrow} B \quad C$$

Eksempel 4.72



Man kan også flytte lidt rundt på placeringen af pilene.

```
\usepackage{all,line,dvips}{xy}
----- slut preamble -----
$\xymatrix{A \var@<3pt>[r] & \\ B \var@<3pt>[l]}$
```

$$A \overset{\longleftarrow}{\longrightarrow} B$$

Eksempel 4.73



Labels på pile

Labels (dvs. tekstmarkeringer) på de forskellige pile angives som sub- eller superscript efter [\langle sekvens \rangle].

Sidebemærkning 4.16. Her er det let at blive forvirret, der er nemlig også en orientering på spil. Op og ned (i.e. om det skal være ^ eller _) afgøres ved at forestille sig at man har et højrehåndssystem liggende i papirets plan, den tredje akse skal så altid pege ind i papiret. Se [Eksempel 4.74](#).

```
\usepackage{all,dvips,line}{xy}
----- slut preamble -----
$\xymatrix{A & \\ B \var[l]^{\{g} \var[r]^{\{f} & C}$
```

$$A \overset{g}{\longleftarrow} B \overset{f}{\longrightarrow} C$$

Eksempel 4.74



Dvs. stort set hver gang en pil vender bagud skal man lige bytte om på op og ned. Det er heller ikke noget problem at skrive både over og under pilen på samme tid.

Labels bliver som standard placeret midt mellem centrene af de to matrixindgange pilen peger mellem, dette kan dog give problemer med indgange af uens størrelse.

4. Matematik på et højere niveau

Tip 4.17. Man kan placere labels centreret i forhold til selve pilen hvis man indsætter et »-« mellem »^« eller »_« og så selve labelteksten, se [Eksempel 4.75](#).

Eksempel 4.75



```
\usepackage[all,line,dvips]{xy}
----- slut preamble -----
$\xymatrix{
  A\times A\times A\ar[r]^f
& B \ar[r]^-g &
  C\times C\times C}$
```

$$A \times A \times A \xrightarrow{f} B \xrightarrow{g} C \times C \times C$$

Sidebemærkning 4.18. Bemærk at pilenes længde justeres *ikke* efter længden af labelen. Her bliver man nødt til manuelt at justere bredden mellem søjlerne, se [Rose \(1999\)](#).

4.6.3 Prækompilering af diagrammer

\CompileMatrices

Hvis man har rigtigt mange af disse Xy-pic diagrammer kan det godt være at oversættelsen af dokumentet bliver lidt langsom. Her an man udnytte \CompileMatrices, som placeres i preamble. Denne makro gør at alle diagrammerne nu bliver gemt, i deres færdigt oversatte form, i eksterne filer (med navnet *<filnavn>-<nummer>.xyc*) og hentes så ind i stedet for at oversætte Xy-matricen hver gang. Den eksterne fil fornyes kun hvis man har lavet ændringer i det tilhørende diagram.



Advarsel 4.19. Når man anvender \CompileMatrices kan parseren være temmelig følsom overfor indholdet af indgangene i matricen. Især hjemmelavede makroer kan få den til at bøvse. Men det fikses meget nemt. Man sætter et beskyttende {}-par omkring hver indgang i matricen, så fungerer det igen. Dvs. [Eksempel 4.75](#) ville så se sådan ud

Eksempel 4.76



```
\usepackage[all,line,dvips]{xy}
\CompileMatrices
----- slut preamble -----
$\xymatrix{
  {A \times A \times A} \ar[r]^f & {B} \ar[r]^-g &
  {C \times C \times C} }$
```

Man kan også få Xy-pic til f.eks. at anvende pilespidser som ligner mere Computer Modern pilene. Sammenlign med [Eksempel 4.75](#).

Eksempel 4.77



```
\usepackage[all,line,dvips,cmtip]{xy}
----- slut preamble -----
$\xymatrix{A\ar[r] & B}$
```

$$A \longrightarrow B$$

4.6.4 Xy-pic og Babel

\shorthandoff
\shorthandon

Vi har tidligere nævnt at babel-pakken vil i den danske opsætning anvende »"« til specielle genveje. Desværre anvender Xy-pic også »"« til noget specielt, så her bliver man nødt til at passe på. Ideen er at anvende babel's \shorthandoff før diagrammet og \shorthandon igen efter.

Xy-pic diagrammer og formelnumre

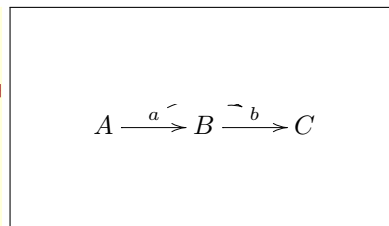
```

\usepackage[danish]{babel}
\usepackage[all,line,dvips]{xy}

----- slut preamble -----

\shorthandoff{"}
\l[\xymatrix{ A \ar[r] ^a & B \ar[r] ^b \\ & C \ar @/^/ "a";"b" } \]
\shorthandon{"}

```



Eksempel 4.78

Man kan også samle deaktiveringen i et environment som man så skal pakkes uden om matematikken.

```

\newenvironment{deactivate}{\shorthandoff{"}}{\shorthandon{"}}

```

Eksempel 4.79

Bemærk at ved andre sprogopsætninger kan det være nødvendigt at anvende andre tegn sammen med `\shorthandon` og `\shorthandoff`.

4.6.5 Xy-pic diagrammer og formelnumre

Hvis man ønsker at formelnummerere et diagram, så man kan jo pakke *equation* omkring. Problemet er så at nummeret ikke centrerer vertikalt i forhold til diagrammet. Man kan komme uden om dette på forskellig vis. En metode er at anvende `\vcenter` omkring hele diagrammet.

`\vcenter`

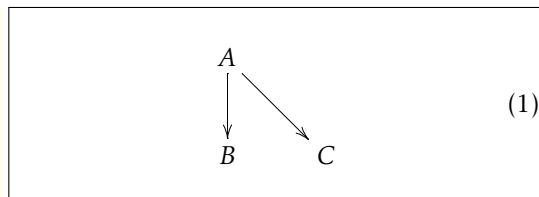
```

\usepackage[all,line,dvips]{xy}

----- slut preamble -----

\begin{equation}
\vcenter{\xymatrix{A \ar[rd] \ar[d] \\ B & C }}
\end{equation}

```



Eksempel 4.80

Hvis man har flere diagrammer på samme linie skal de have `\vcenter` omkring sig, hver for sig. Metoden her er mest anvendelig hvis man ikke har så mange nummererede diagrammer. Har man mange kan man anvende en anden konstruktion.¹⁸

```

\newcommand\cxymatrix[2][]{%
\begin{xy}*[c]\xybox{\xymatrix#1{#2}}\end{xy} }

```

Kodetip 4.1

Så vil `\cxymatrix` automatisk centrerer. Syntaksen bliver bare lidt anderledes. F.eks. skal man nu anvende `\cxymatrix[@C=2cm]{...}` i stedet for `\xymatrix@C=2cm{...}`.

4.6.6 Andre Xy-pic tips

I det næste eksempel viser vi hvordan man får indgange til at fylde mindre i et Xy-diagram.

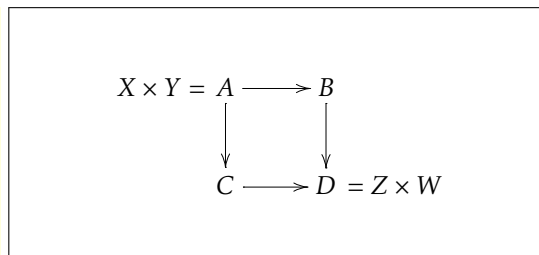
```

\usepackage[all,line,dvips]{xy}

----- slut preamble -----

\l[
\xymatrix{\llap{\$X\times Y = }{\$,} A}
\ar[r] \ar[d] & B \ar[d] \\
C \ar[r] & D \rlap{\$,} = Z\times W\$}
}
\l[

```



Eksempel 4.81

¹⁸ Efter ide af Emil Hedevang Lohse.

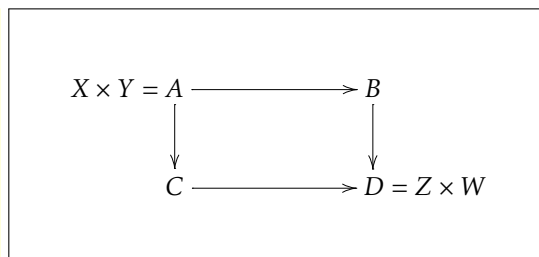
4. Matematik på et højere niveau

Det kan dog også gøres direkte med Xy -konstruktioner:

Eksempel 4.82



```
\usepackage{all,line,dvips}{xy}
----- slut preamble -----
\l
\xymatrix{ **[l] {X\times Y = A}
\ar[r] \ar[d] & B \ar[d] \l
C \ar[r] & **[r] {D = Z\times W}
}
\l
```

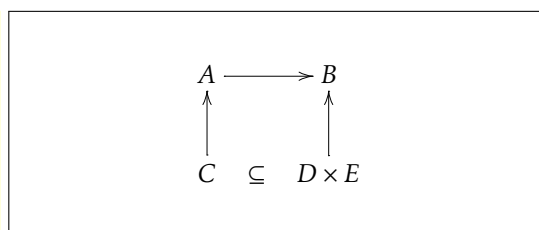


Antag at man i et diagram har at et af forbindelserne bare er en inklusion, så kan dettes jo skrives som $\xrightarrow{\subseteq}$. Men det ville måske være pænere hvis man kun skrev \subseteq i stedet for pilen. Se [Eksempel 4.83](#) samt den efterfølgende forklaring.

Eksempel 4.83



```
\usepackage{all,line,dvips}{xy}
----- slut preamble -----
\l
\xymatrix{ A \ar[r] & B \l
C \ar[u]
\ar@{}[r]|-\{\displaystyle\subseteq\}
& D\times E \ar[u]
} \l
```



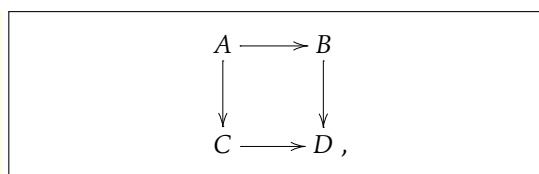
Forklaring: » $\ar@{}[r]$ « er den tomme pil, » $|$ « betyder at lablen skal skrives inden i pilen i stedet for over eller under. »-« gør, som tidligere nævnt, at lablen placeres midt mellem kanterne af de to kasser 'pilen' forbinder. \displaystyle gør at \subseteq kommer op i normalstørrelse, labels kommer jo normalt ud i \scriptstyle .

Tidligere har vi skrevet lidt om kommatering og matematik, hvor vi anvendte et diagram som eksempel. De firkantede diagrammer kan man sagtens kommatere, det er dog en god ide at gøre det på følgende vis

Eksempel 4.84



```
\usepackage{all,line,dvips}{xy}
----- slut preamble -----
\l
\xymatrix{ A \ar[r] \ar[d] & B \ar[d] \l
C \ar[r] & D \rlap{\ , }
} \l
```



hvor man naturligvis skal lægge mærke til $\rlap{\ , }$ som bevirker at kommaet ikke forstyrrer placeringen af diagram indgangen.

4.6.7 Alternativ til Xy -pic

Nogle vil måske synes at $tikz$ er mere fleksibel at arbejde med end xy -pakken. Her er et diagram lavet med $tikz$.¹⁹

¹⁹ Min matematiske tankegang er i hvertfald mere glad for denne trinvis opbygning end den til tider emget skumle syntaks som anvendes i Xy -Pic.

Alternativ til Xy-pic

```
\usepackage{tikz}
\usetikzlibrary{matrix}

----- slut preamble -----

\l
\begin{tikzpicture}
  \matrix (M)
  [matrix of math nodes, row sep=2em,column sep=2em ]
  {
    A & B & \& C & D & \& % sidste & \& er VIGTIG \\
  };
  \begin{scope}[execute at begin node=$\scriptstyle, execute at end node=$]
    \draw[->] (M-1-1) -- node[above] {\alpha} (M-1-2);
    \draw[->] (M-1-1) -- (M-2-1);
    \draw[->] (M-1-2) -- node[left] {\beta} (M-2-2);
    \path (M-2-1) -- node {\displaystyle\subseteq} (M-2-2);
  \end{scope}
\end{tikzpicture}
\l
```

Eksempel

4.85



$$\begin{array}{ccc} A & \xrightarrow{\alpha} & B \\ \downarrow & & \beta \downarrow \\ C & \subseteq & D \end{array}$$

For mere info tag et kig i [Tantau \(2007\)](#) eller smut forbi det glimrende galleri <http://www.texample.net/tikz/>.

Om sætningskonstruktioner

Indenfor matematik og lignende fag får man altid brug for sætningslignende konstruktioner – som også kan anvendes i sammenhænge hvor man måske ikke lige tænke på at anvende en sætning.

I dette kapitel vil vi gennemgå brugen af sætninger i L^AT_EX samt konfigureringen af disse.

Når man nu har en sætning vil man jo også gerne have et bevis,¹ dette er også noget vi vil kigge på.

Mantra 5.1. Det er spild af tid at begynde at kode egne sætningskonstruktioner helt fra bunden af. Lær i stedet at anvende og konfigurere de konstruktioner som allerede tilbydes.

En anden ting man desværre ser i visse journalskabeloner er *hjemmelavede* beviskonstruktioner. Ofte ser man noget lignende

```
\textit{Proof.} ... \halmos
```

hvor `\halmos` forsøger at placere en firkant. Dette bør man for alt i verden undgå. I det mindste bør man da anvende standardpakken `amsthm` samt det tilhørende *proof*-environment.

Eksempel
5.1

Efter min overbevisning er en god bevis/sætnings anvendelse når man sørger for at konstruktionen er tydeligt adskilt fra den omkringliggende tekst, f.eks. ved at sørge for, at der er lidt blank plads før og efter. Det kan desuden være en god ide at markere sætninger yderligere, f.eks. ved at skrive dem i kursiv eller på anden vis gøre dem nemmere for læseren at finde når man bladrer gennem teksten. Af samme grund bør man tydeligt markere hvornår et bevis et slut, f.eks. ved placering af \square ved den højre kant af teksten.

Sidebemærkning 5.2 (Hvornår bør man skrive »Bevis for ... «?).

Når der kun er nogle få liniers tekst mellem f.eks. teksten for et lemma samt beviset for selv samme, så er det nok at beviset starter med ordet »Bevis«.

Hvis tværtimod der er flere sætnings/lemma/... deklamationer eller beviser mellem sætningen og beviset så bør man naturligvis gøre opmærksom på hvilken sætning det er beviset hører til. Se f.eks. [Eksempel 5.22 på side 126](#).

¹ Det er der i hvert fald nogen som gerne vil have.

5. Om sætningskonstruktioner

5.1 Opret en ny sætningskonstruktion

Definition 5.3. En sætning i L^AT_EX er i bund og grund et environment med en overskrift og et nummer som tælles op hver gang environmentet anvendes. Sætningskonstruktionen indeholder normalt også ekstra luft før og efter.

Sidebemærkning 5.4. Det vi viser her er den klassiske deklaration af sætningsenvironmentens, der findes også andre interfaces.

`\newtheorem` En sætningskonstruktion defineres i L^AT_EX via makroen `\newtheorem` som kan anvendes i tre forskellige varianter.

```
\newtheorem{<env-navn>}{<thm-navn>}
\newtheorem{<env-navn>}{<thm-navn>}[<dominerende-tæller>]
\newtheorem{<env-navn>}[<fælles-tæller>]{<thm-navn>}
```

Fortløbende ikke-domineret og ikke fælles sætning

Første variant er den grundlæggende konstruktion, hvor `<env-navn>` er det navn man ønsker at bruge til sætningsenvironmentet (dette navn bør *ikke* indeholde danske bogstaver). Dette kunne f.eks. være `thm` for sætning

Eksempel 5.2

```
\newtheorem{thm}{Sætning}
```

hvilket så kan anvendes på følgende vis:

Eksempel 5.3



```
\newtheorem{thm}{Sætning}
----- slut preamble -----
\begin{thm}
Dette er en sætning.
\end{thm}
```

Sætning 1 *Dette er en sætning.*

Man observerer at overskriften skrives i fed og selve teksten kommer i kursiv. Afsnit 5.2.1 på side 118 samt 5.3.1 på side 121 forklarer to måder man kan ændre dette på.

Domineret sætning

Variant nummer to anvendes når man gerne vil have sætningsnummeret underlagt en anden tæller. Her angiver man så navnet på denne tæller som et ekstra argument (yderst til højre) i `[]`-parenteser. Navnet på tælleren vil oftest være enten »chapter« eller »section«. Vores eksempel fra før kunne så blive:

Eksempel 5.4



```
\newtheorem{thm}{Sætning}[section]
----- slut preamble -----
\begin{thm}
Dette er en sætning.
\end{thm}
```

Sætning 5.1 *Dette er en sætning.*

Sætningstælleren vil så automatisk blive *nulstillet* hver gang `section` ændres.

Domineret og underlagt sætning

Tredje variant anvendes når man gerne vil have flere sætningslignende environments til at anvende den samme tæller, dvs. således at man kan få »Sætning 1« efterfulgt af »Lemma 2«, efterfulgt af »Proposition 3« efterfulgt af »Sætning 4« etc.



Advarsel – teknisk forklaring 5.5. Når man anvender en af varianterne

```
\newtheorem{<env-navn>}{<thm-navn>}
\newtheorem{<env-navn>}{<thm-navn>}[<dominerende-tæller>]
```

vil der ud over environmentet `<env-navn>` også blive lavet en ny tæller med navnet »env-navn«, i vore to eksempler ovenfor er der altså lavet en tæller med navnet »thm«. Det er denne tæller som sørger for sætningsnummeret.

Hvis man nu vil lave et lemma-environment, som skal nummereres *sammen med* thm-environmentet, skal man altså anvende tredje variant med »thm« angivet som den fælles tæller.² Det er altså syntaksen

```
\newtheorem{<env-navn>}[<fælles-tæller>]{<thm-navn>}
```

man skal anvende. Eksempel:

<pre>\newtheorem{thm}{Sætning} \newtheorem{lemma}[thm]{Lemma} ----- slut preamble ----- \noindent Noget tekst. \begin{thm} En sætning.\end{thm} Noget mere tekst. \begin{lemma} Et lemma.\end{lemma} \begin{thm} En anden sætning.\end{thm}</pre>	<p>Noget tekst.</p> <p>Sætning 1 <i>En sætning.</i></p> <p>Noget mere tekst.</p> <p>Lemma 2 <i>Et lemma.</i></p> <p>Sætning 3 <i>En anden sætning.</i></p>
---	---

Eksempel 5.5



Til tider vil man gerne kunne give en sætning et ekstra navn, f.eks. hvis man anvender en særlig kendt sætning. Løsning: Angiv navnet som et ekstra argument til sætningsenvironmentet. Se desuden [Eksempel 5.25 på side 128](#).

<pre>\newtheorem{lemma}{Lemma} ----- slut preamble ----- \begin{lemma}[daleifs lemma] \#\\$\@***!!!? \end{lemma}</pre>	<p>Lemma 1 (daleifs lemma) #\\$***!!!?</p>
--	---

Eksempel 5.6



5.2 Opsætning af sætningskonstruktioner med amsthm-pakken

Den almindelige opsætning af `\newtheorem` (som set ovenfor) giver ikke brugeren nogen særlige muligheder for at ændre på udseendet. For at gøre dette må vi ty til ekstra pakker. De mest anvendte er `amsthm` og `ntheorem`.³

Selv om `amsthm` ikke er lige så fleksibel som `ntheorem` tager vi den med alligevel idet til artikler og lignende er `amsthm` et ganske fornuftigt valg da forlagene må forventes at kunne håndtere `amsthm` konstruktioner.

amsthm
ntheorem

² Bemærk at det giver ikke mening at blande syntaksen for variant 2 og 3.

³ Der er også andre pakker, men disse behandles ikke her.

5. Om sætningskonstruktioner

Tip 5.6. Med pakken `thmtools` kan man faktisk styre disse environments i ret stor detalje, pakken er et kig værd hvis man anvender `amsthm`.

5.2.1 Almindelig brug af `amsthm`

Pakken `amsthm` følger med resten af \mathcal{AMS} -systemet og giver adgang til to ekstra makroer

```
\theoremstyle
\newtheorem*{<env-navn>}{<thm-navn>}
\theoremstyle{<style>}
```

`\newtheorem*` laver et sætningsenvironment som ikke har noget nummer.

Eksempel 5.7



```
\usepackage{amsthm}
\newtheorem*{zorn}{Zorns lemma}
----- slut preamble -----
\begin{zorn}
Zorn er det bare\dots
\end{zorn}
```

Zorns lemma. *Zorn er det bare...*

Observer iøvrigt at med `amsthm` kommer der automatisk et ».« efter overskriften.

`\theoremstyle` understøtter tre forskellige værdier⁴

plain er standard opsætningen, dvs. overskrift og nummer i almindelig tekstfont med fed. Indholdet af sætningen er i kursiv.

definition som *plain* men her er indholdet skrevet med den almindelige tekst font i stedet for kursiv.

remark som *definition* men nu med headerteksten i kursiv i stedet for fed.

Sidebemærkning 5.7 (`\newtheorem` gemmer konfigurationen).

`\newtheorem` er lidt speciel i og med at når den anvendes så gemmer den en kopi af den nuværende sætningskonfiguration i det nye environment. Dvs. at anvender man `\theoremstyle` skal man ikke bekymre sig om at dette skulle påvirke de sætningskonstruktioner man allerede havde oprettet tidligere i dokumentet.

Dette bliver endnu mere relevant når vi i et senere afsnit når til `ntheorem` pakken.

Eksempel 5.8



```
\usepackage{amsthm}
\newtheorem{thm}{Sætning}
\theoremstyle{definition}
\newtheorem{defn}[thm]{Definition}
\theoremstyle{remark}
\newtheorem{remark}[thm]{Bemærkning}
----- slut preamble -----
\begin{thm} En sætning.\end{thm}
\begin{defn} En definition.\end{defn}
\begin{remark} En bemærkning.
\end{remark}
```

Sætning 1. *En sætning.*

Definition 2. En definition.

Bemærkning 3. En bemærkning.

For mere information vedrørende `amsthm` se [AMS \(2004\)](#), her vil man også finde information om hvordan man laver sine egne `amsthm` sætningsstile.

⁴ Bemærk at dette gælder kun for `amsthm`, thi `ntheorem` har også en `\theoremstyle`-makro som dog anvendes til noget helt andet.

Beviser med amsthm

5.2.2 Beviser med amsthm

Pakken `amsthm` stiller også et *proof*-environment til rådighed. Dette environment har den fordel at automatisk den placerer en slut-»□« til højre efter environmentet, men *altid på samme side*⁵ som afslutning af beviset. Bevisnavnet kan styres via `babel`.

`proof`

`babel`

<pre>\usepackage[danish]{babel} \usepackage{amsthm} ----- slut preamble ----- \begin{proof} Dette er et bevis \end{proof}</pre>	<p><i>Bevis.</i> Dette er et bevis □</p>
---	--

Eksempel

5.9



proof-environmentet kan tage et ekstra valgfrit argument som så erstatter »*Bevis*«-delen.

<pre>\usepackage[danish]{babel} \usepackage{amsthm} ----- slut preamble ----- \begin{proof}[Bevis for fundamentalsætning] \dots \end{proof}</pre>	<p><i>Bevis for fundamentalsætning. ...</i> □</p>
---	---

Eksempel

5.10



At »*Bevis*« skal være kursiv er desværre direkte hardwired⁶ ind i `amsthm`, men lidt snedighed kan altid hjælpe. Man skal omdefinere `\proofname` (men pga. `babel` skal det gøres *efter* `\begin{document}`), eller se afsnit 9.4 på side 229).

`\proofname`

<pre>\usepackage{amsthm} ----- slut preamble ----- \renewcommand{\proofname}% {\textnormal{\textbf{Bevis}}} \begin{proof} Dette er et bevis \end{proof}</pre>	<p>Bevis. Dette er et bevis □</p>
---	--

Eksempel

5.11



Man skal altid sørge for at der *ikke* er blanke linier før »`\end{proof}`«. En anden ting er at afslutter man sit bevis med andet end tekst bliver slutmarkøren ofte placeret lidt for langt nede. Man kan dog i stedet placere den pr. håndkraft via makroen `\qedhere`.⁷

`\qedhere`

⁵ Meget almindeligt problem ved hjemmelavede beviskonstruktioner, slutmarkøren kommer pludselig til at stå alene øverst på en side, eller den står ikke ude ved højre kant.

⁶ Dvs. det står direkte i kildekoden og er derfor ikke konfigurerbart.

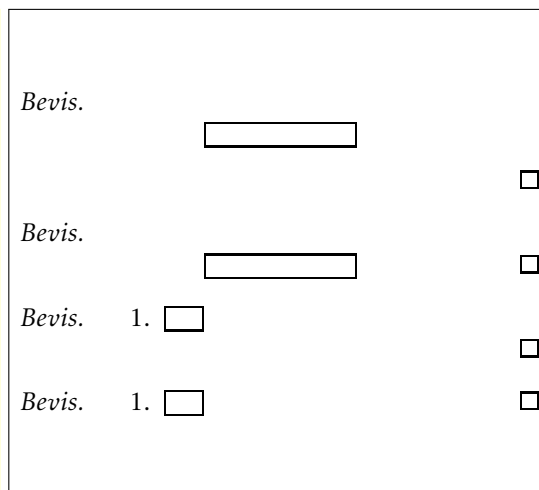
⁷ Gælder kun matematik konstruktioner understøtter af \mathcal{AMS} .

5. Om sætningskonstruktioner

Eksempel 5.12



```
\usepackage{amsthm}
----- slut preamble -----
\begin{proof}
\l[ \dbx[2cm] \l]
\end{proof}
\begin{proof}
\l[ \dbx[2cm] \qedhere \l]
\end{proof}
\begin{proof}
\begin{enumerate} \item \dbx
\end{enumerate}
\end{proof}
\begin{proof}
\begin{enumerate} \item \dbx \qedhere
\end{enumerate}
\end{proof}
```



ntheorem Pakken *ntheorem* er en del mere intelligent med hensyn til placeringen af slutmarkøren og kan lave tricket anvendt i [Eksempel 5.12](#) automatisk.

Sidebemærkning 5.8. Har man af en eller anden grund brug for at placere en »□« ved højrekant af papiret, men *ikke* er indenfor et bevis, så kan dette gøres med makroen `\qed`.

Anvender man *ntheorem*, skal man så huske at `\qed` makroen ikke er tilgængelig. Den kan dog nemt stjæles fra `amsthm.sty`.

Sidebemærkning 5.9. Pakken `amsthm` indlæses automatisk hvis man anvender en af $\mathcal{A}\mathcal{M}\mathcal{S}$ -klasserne `amsart` eller `amsbook`.

Dette betyder specielt at pakken *ntheorem* ikke kan anvendes sammen med disse klasser.

5.3 Opsætning af sætningskonfigurationer med *ntheorem*-pakken

ntheorem Pakken *ntheorem* er efter min mening den mest brugbare sætningsudvidelse til \LaTeX – den har en meget nem og simpel konfiguration og giver ekstra features som andre sætningspakker mangler. Manualen til *ntheorem* kan ses i [May \(2002\)](#).

Den almindelige brug af *ntheorem* er via

Eksempel 5.13

```
\usepackage[amsmath, thmmarks, ...]{ntheorem}
```

amsmath denne option sætter nogle ‘hooks’ ind i diverse `amsmath` environments for at kunne lave afslutningsmarkeringen, se nedenfor. Da denne option ændrer noget fra `amsmath` er det derfor nødvendigt at *ntheorem* loades *efter* `amsmath`!

thmmarks Slår muligheden for afslutningsmarkeringen til, beskrives i afsnit [5.3.3](#) på side 124.

For flere options til *ntheorem*, se [May \(2002\)](#).

5.3.1 Konfigurering

Syntaksen for `\newtheorem` med `ntheorem` er den samme som allerede nævnt i afsnittet om `\newtheorem` på side 116, desuden har `ntheorem` (ligesom `amsthm`) en `\newtheorem*` til ikke-nummererede konstruktioner. Den største forskel er `ntheorem`'s konfigurationsmuligheder.

Lad os se på konfigureringen.

`\theoremstyle{<style>}`

se beskrivelsen på den følgende side.

`\theorembodyfont{}`

ændrer fonten anvendt til selve teksten i sætningskonstruktionen. Vil man f.eks. sørge for at den næste sætningskonstruktion man definerer får den almindelige tekstfont kan man anvende `\theorembodyfont{\normalfont}`.

`\theoremheaderfont{}`

dette ændrer fonten anvendt til selve overskriften, nummeret samt den valgfrie ekstra tekst. Eksempel:

```
\theoremheaderfont{\normalfont\bfseries\large}
```

Bemærk at idet `\theoremheaderfont` arver fontegenskaber fra `\theorembodyfont` kan det være en god ide at starte `\theoremheaderfont` med `\normalfont` (og evt. `\normalsize`).

`\theoremseparator{<ting>}`

er det som adskiller overskriften fra selve teksten. `<ting>` kan være `»:«, ».»` etc. Det kan også være en spacing kommando hvis man gerne vil have noget mere luft. Eksempel `\theoremseparator{.}`.

Bemærk at `\theoremseparator` *ikke* skal anvendes til at lave sætninger hvor overskriften står på en linie for sig selv. Hvordan man opnår lineskift i starten af en sætningskonstruktion kommer vi til senere, se afsnit 5.3.2 på næste side.

`\theoremindent{<længde>}`

kan anvendes til at indentere *hele* konstruktionen, ikke bare første linie. Det mest normale er at denne er sat til nul, hvilket den er som standard.

`\theoremnumbering{<style>}`

angiver hvordan konstruktionsnummeret skal formateres, som værdier til `<style>` kan anvendes `»arabic«` i.e. almindelige tal (standarden), `»alph«`, `»Alph«` små eller store bogstaver, `»roman«`, `»Roman«` små eller store romertal, `»greek«`, `»Greek«` små eller store græske bogstaver.

`\theoremsymbol{<ting>}`

anvendes til at specificere det symbol (hvis man ønsker et) som vil anvendes som afslutningssymbol i den næste sætningskonstruktion som man vil definere. Brugen er beskrevet nærmere i afsnit 5.3.3 på side 124. Husk at hvis tingen man vil anvende er et matematiksymbol som f.eks. `\square` (\square) da skal man skrive det som `\ensuremath{\square}`.

`\theoremclass{<env-navn>}`

anvender man denne med et `<env-navn>` som allerede er defineret, så anvendes de konfigurationer som blev registreret via `\newtheorem{<env-navn>}{<navn>}`.

For bedre at få overblik over hvordan dette fungerer ser vi på et eksempel

5. Om sætningskonstruktioner

Eksempel

5.14



```
\usepackage{ntheorem}
\theoremheaderfont{\normalfont}
\theoremseparator{: \quad}
\theoremnumbering{greek}
\newtheorem{thm}{Sætning}
\theoremheaderfont{\normalfont\bfseries}
\theorembodyfont{\normalfont\sffamily}
\newtheorem{lemma}[thm]{Lemma}
----- slut preamble -----
\begin{thm} tekst \end{thm}
\begin{lemma} tekst \end{lemma}
```

Sætning α : tekst

Lemma β : tekst

Det vigtige at forstå her er at når `\newtheorem` bliver udført husker den automatisk den øjeblikkelige værdi af de forskellige `\theorem<X>`-makroer. Hvilket igen betyder at når / hvis vi så senere ændrer f.eks. `\theoremsymbol` da ændrer vi ikke ved de sætningskonstruktioner som vi allerede har lavet.

5.3.2 Theoremstyles

Pakken `ntheorem` har nogle helt anderledes theoremstyles end `amsthm`, hvor de i `amsthm` ændrede på nogle fonte, ændrer de nu i `ntheorem` ved måden hvorpå headeren opfører sig på. Vi har følgende theoremstyles, se også [Eksempel 5.15 på næste side](#). Ved beskrivelsen af hver theoremstyle finder man desuden et eksempel hvor resultatet.

plain er standardstilen. Her er headeren bare $\langle navn \rangle \langle nummer \rangle$ og teksten begynder på samme linie som headeren.

Sætning 1 *Tekst*

break som ved **plain**, men headeren er nu på en linie for sig selv og teksten begynder på næste linie.

Sætning 1
Tekst

change som **plain**, men med rækkefølgen af $\langle navn \rangle$ og $\langle nummer \rangle$ byttet om.

1 Sætning *Tekst*

changebreak svarende til **break**.

1 Sætning
Tekst

margin som **plain**, men her er nummeret skrevet i marginen før navnet.

1 Sætning *Tekst*

marginbreak svarende til **break**.

1 Sætning
Tekst

nonumberplain svarer til **plain**, men uden noget nummer. Vi skal senere se at denne kan anvendes til bevis-konstruktioner.

Bevis *Tekst*

Theoremstyles

nonumberbreak svarende til **break**, men uden nummer.

Bevis

Tekst

empty som **nonumberplain**, men her helt uden overskrift, i.e. kun det valgfrie argument til sætningen bliver skrevet som overskrift.

emptybreak som **empty**, men med lineskift efter overskriften.

```
\usepackage{ntheorem}
\newtheorem{thm}{Sætning}
\theoremstyle{break}
\newtheorem{lemma}[thm]{Lemma}
\theoremstyle{marginbreak}
\theorembodyfont{\normalfont}
\newtheorem{defn}[thm]{Definition}

----- slut preamble -----

\begin{thm} En almindelige sætning
\end{thm}
\begin{lemma} Her starter teksten
først på næste linie
\end{lemma}
\begin{defn} Her er nummeret før
navnet og vi starter på en ny linie
\end{defn}
```

Sætning 1 *En almindelige sætning*

Lemma 2

Her starter teksten først på næste linie

3 Definition

Her er nummeret før navnet og vi starter på en ny linie

Eksempel

5.15



Sidebemærkning 5.10 (Egne theorem styles).

Theorem styles til `ntheorem` er faktisk ikke så svære at lave som man skulle tro. Ofte er det nemmest at tage udgangspunkt i en af de eksisterende stile, disse findes i »`ntheorem.sty`«. Husk at laver man stilen *minstil*, og ønsker at anvende den sammen med `\newtheorem*` skal man huske også at lave *nonumberminstil* hvor man har fjernet henvisningen til sætningsnummeret. Som et eksempel finder man herunder et eksempel (anvendt til kasser som denne), det er en sammenblanding mellem *plain* og *break* således at når der kommer en ekstra titel på, så laves et automatisk lineskifte før teksten kommer.

```
\makeatletter
\newtheoremstyle{forframed}%
{\item[\hskip\labelsep \theorem@headerfont ##1\ ##2\theorem@separator]}%
{\item[\rlap{\vbox{\hbox{\hskip\labelsep \theorem@headerfont
##1\ ##2\ (##3)\theorem@separator}\hbox{\strut}}]}%
\newtheoremstyle{nonumberforframed}%
{\item[\hskip\labelsep \theorem@headerfont ##1\theorem@separator]}%
{\item[\rlap{\vbox{\hbox{\hskip\labelsep \theorem@headerfont
##1\ (##3)\theorem@separator}\hbox{\strut}}]}%
\makeatother
```

Her er »`##1`« en henvisning til den almindelige overskrift, »`##2`« en henvisning til den formaterede sætningstæller, og »`##3`« en henvisning til den ekstra overskrift. Som man kan se er andet argument til `\newtheoremstyle` koden til overskrifter uden ekstra tekst.

Eksempel

5.16

5. Om sætningskonstruktioner

Citering i sætningsoverskrift

Normalt kan man godt citere en artikel som en del af det valgfri argument til sætningen, dvs.

Eksempel 5.17 `\begin{theorem}[\cite{vigtigartikel}]`

Men ønsker man at tilføje noget i citeringen, i.e. `\cite[noget tekst]{vigtigartikel}` så går det galt pga. firkantede paranteser indeni firkantede parenteser. Brug i stedet:

Eksempel 5.18 `\begin{theorem}[\cite[noget tekst]{vigtigartikel}]`

5.3.3 Slutmarkører

Når `ntheorem` er aktiveret med optionen `[thmmarks, ...]` har man nu muligheden for at få placeret et symbol i slutningen af ens sætningskonstruktion (eller et bevis). I modsætning til ***proof*** fra `amsthm` vil placeringen her være noget mere intelligent, f.eks. kan den selv sørge for at flytte placeringen op i slutningen af sidste punkt af en liste, hvis teksten slutter med en liste. Den kan også placere symbolet pænt hvis teksten slutter med noget fremhævet matematik.

Sidebemærkning 5.11. Det skal lige indskydes at det anses som dårlig stil at afslutte et bevis med en fremhævet formel, men det afholder altså ikke folk fra at gøre det alligevel.

Sidebemærkning 5.12. Denne bedre placering af slutmarkør kommer dog med en pris. Pakken `ntheorem` anvender `aux`-filen til at beregne sig frem til hvor slutsymbolet skal placeres. Dette kan så i værste fald desværre betyde at man kan blive nødt til at kompilere sit dokument op til *fem* gange før den endelige placering er fastlagt.

Sidebemærkning 5.13 (Blanke linier til slut).

Desuden er det meget vigtigt at linien lige over f.eks. `\end{thm}` eller snarere `\end{proof}` ikke må være blank thi så forsvinder slutmarkøren.

Man bør generelt heller ikke anvende to slutmarkørskrivende environments indeni hinandens således at man ender med situationen

```
\end{env2}
\end{env2}
```

Her kan slutsymbolet for det ene environment forsvinde. Det er muligt at få dette til at fungere, men det kræver en smule ekstra håndarbejde som vi dog udelader her.

De fleste anvender i forvejen pakken `amsmath` så man skal huske at loade `ntheorem` *efter* `amsmath` og huske at lade `amsmath` være option til `ntheorem`, i.e.

```
\usepackage{amsmath}
\usepackage[amsmath, thmmarks]{ntheorem}
```

Beviser med ntheorem

Man kan så frit vælge om man vil have et slutsymbol til sin konstruktion eller ej. Hvis man ikke vil have det, skal man bare anvende `\theoremsymbol{}` (hvilket er standardværdien til at starte med).

`\theoremsymbol`

```
\usepackage{amsmath,amssymb}
\usepackage[amsmath,thmmarks]{ntheorem}
\theoremsymbol{\rule{4pt}{4pt}}
\newtheorem{thm}{Sætning}
\theoremstyle{break}
\theoremsymbol{%
  \ensuremath{\scriptstyle\blacktriangle}}
\newtheorem{lemma}[thm]{Lemma}

----- slut preamble -----

\begin{thm} tekst \end{thm}
\begin{lemma} \begin{enumerate}
\item tekst \item tekst
\end{enumerate} \end{lemma}
```

Sætning 1 tekst ■

Lemma 2

1. tekst

2. tekst ▲

Eksempel

5.19



I [May \(2002\)](#) kan man læse mere om opsætningen af `ntheorem`, bl.a. om hvordan man manuelt kan sætte/fjerne slutmarkøren i hvis man pludseligt stod og havde brug for dette.

5.3.4 Beviser med ntheorem

Med `ntheorem` er det meningen man også anvender en sætningskonstruktion til manuelt at lave et *proof*-environment med, `ntheorem`'s indbyggede support for slutsymboler lægger jo også op til dette. Det eneste man skal gøre op med sig selv er headerfont, bodyfont, slutsymbol og om man vil have `nonumberplain` eller `nonumberbreak` som `theoremstyle`. Følgende er meget almindelig.

proof

```
\usepackage{amsmath,amssymb}
\usepackage[amsmath,thmmarks]{ntheorem}
\theoremstyle{nonumberplain}
\theoremheaderfont{%
  \normalfont\bfseries}
\theorembodyfont{\normalfont}
\theoremsymbol{\ensuremath{\square}}
\theoremseparator{.}
\newtheorem{proof}{Bevis}

----- slut preamble -----

\begin{proof}
Dette er et bevis med noget
matematik
\begin{equation*}
a^2+b-1=2.
\end{equation*}
\end{proof}
```

Bevis. Dette er et bevis med noget matematik

$$a^2 + b - 1 = 2.$$

□

Eksempel

5.20



Den ekstra overskrift som kan gives til en sætningskonstruktion er lidt anderledes ved `ntheorem`:

5. Om sætningskonstruktioner

Eksempel

5.21



```
\usepackage[thmmarks]{ntheorem}
\theoremstyle{nonumberplain}
\theoremheaderfont{\normalfont\bfseries}
\theorembodyfont{\normalfont}
\theoremsymbol{\rule{5pt}{5pt}}
\theoremseparator{.}
\newtheorem{proof}{Bevis}

----- slut preamble -----

\begin{proof}[for mit lemma]
tekst
\end{proof}
```

Bevis (for mit lemma). tekst ■

Antag vi er i gang med at bevise en hovedsætning. Vi har lige færdiggjort 35 lemmaer (det er en meget svær sætning) og nu vil vi gerne lave beviset for sætningen. Dette skriver man gerne som: »*Bevis for Sætning...*«. Dette kan vi fiks ved at lave et ekstra *proof*-environment: *proofof*. Bemærk brugen af stilen **empty**.

Eksempel

5.22



```
\usepackage[thmmarks]{ntheorem}
\theoremstyle{empty}
\theoremheaderfont{\normalfont\itshape}
\theorembodyfont{\normalfont}
\theoremsymbol{\rule{5pt}{5pt}}
\theoremseparator{.}
\newtheorem{proofof}{}

----- slut preamble -----

\begin{proofof}[Bevis for daleifs
hovedsætning.]% anvend en reference her
Simpelt, ved brug af lemma 13, 17, 23, 29 og 34.
\end{proofof}
```

Bevis for daleifs hovedsætning. Simpelt, ved brug af lemma 13, 17, 23, 29 og 34. ■

Tip 5.14 (Matematik til slut).

Nogle puritanere mener at det er dårlig stil at slutte et bevis med noget fremhævet matematik, man bør i stedet skrive »*Hvilket skulle vises*« eller lignende.

Men det er ikke alle som går op i den slags. Anvender man `ntheorem` så piller den ved f.eks. *equation* eller *align* og vil så forsøge at placere slutmarkøren i området omkring formelnummeret. Hvis materialet rager et stykke under basislinjen, er dette ikke en optimal placering, som man kan se på dette eksempel

```
\begin{proof}
\dots
\begin{equation*}
\frac{1}{\frac{1}{2}}
} = 2.
\end{equation*}
\end{proof}
```

Bevis. ... $\frac{1}{\frac{1}{2}} = 2.$ □

Eksempel 5.23



Dette kan man fikse ved at indlæse `empheq`-pakken med den rigtige option, også skrive tingene lidt om.

```
\usepackage{amsmath,amssymb}
\usepackage[ntheorem]{empheq}
\usepackage[amsmath,thmmarks]{ntheorem}
\theoremstyle{nonumberplain}
\theoremheaderfont{\itshape}
\theorembodyfont{\normalfont}
\theoremseparator{. \ }
\theoremsymbol{\enspace\ensuremath\square}
\newtheorem{proof}{Bevis}
```

— slut preamble —

```
\begin{proof}
\dots
\begin{empheq}{equation*}
\frac{1}{\frac{1}{2}} } = 2.
\end{empheq}
\end{proof}
```

Bevis. ... $\frac{1}{\frac{1}{2}} = 2.$ □

Eksempel 5.24



Ser man efter vil man se at placeringen af markøren passer med bunden af brøken.

5.4 Andet godt med sætningskonstruktioner

Vi vil her gå ud fra at man altid anvender `ntheorem`. Som tidligere nævnt skal man ikke kun tænke på sætningskonstruktioner som noget vedrørende sætninger og beviser. Det er jo *bare* en overskrift med et nummer som tæller op for hver gang man anvender `environmentet`. Dvs. man kan uden problemer anvende det til at lave en liste af opgaver, spørgsmål etc. De bliver jo så automatisk fint afgrænset fra resten af teksten.

Som studerende kan man også anvende en lignende konstruktion til at pakke uden om sine besvarelser (hvis man f.eks. laver sine afleveringer i $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$). Her kan `\theoremstyle{empty}` med held anvendes.

5. Om sætningskonstruktioner

Eksempel

5.25



```
\usepackage{amsmath}
\usepackage[amsmath, thmmarks]{ntheorem}
\theoremstyle{empty}
\theoremheaderfont{\normalfont\bfseries}
\theorembodyfont{\normalfont}
\theoremsymbol{\ \textnormal{q.e.d.}}
\newtheorem{opgave}{Opgave}

----- slut preamble -----

\begin{opgave}[Opgave 2.3.]
en besvarelse.
\end{opgave}
```

Opgave 2.3. en besvarelse. q.e.d.

Det man skal bemærke er at man nu selv kan skrive overskriften til sætningen, hvilket jo kan være nødvendigt idet de stillede opgaver jo sjældent kommer i rækkefølge.

5.4.1 Sætningskonstruktioner med ramme eller farvet baggrund

Dette kan ntheorem faktisk godt selv gøre. Men jeg er ikke fan af resultatet:

Eksempel

5.26



```
\usepackage{xcolor, framed}
\usepackage[framed]{ntheorem}
\newframedtheorem{thm}{Sætning}
\def\theoremframecommand{\colorbox{gray}}
\newshadedtheorem{lemma}{Lemma}

----- slut preamble -----

tekst tekst tekst
\begin{thm}
tekst tekst tekst
\end{thm}
tekst tekst tekst
\begin{lemma}
tekst tekst tekst
\end{lemma}
tekst tekst tekst
```

tekst tekst tekst

Sætning 1 tekst tekst tekst

tekst tekst tekst

Lemma 1 tekst tekst tekst

tekst tekst tekst

Efter min mening er der alt for meget spildt plads her. Der er luft fra teksten ned til rammen eller den farvede baggrund, og fra kanten af denne og ned til teksten inden i sætningen.

Den bedste løsning på dette er at anvende en bedre konstruktion til at lave rammer og farvet baggrund med, og så på en eller anden måde sammenflette dette med sætningskonstruktionerne.



Der er kommet en ny løsning på dette problem via nogle nye features i mdframed-pakken. Man skal mindst anvende version 0.9e for at få adgang til løsningen vi beskriver om lidt. Dette, kan man sørge for via

Eksempel

5.27

```
\usepackage{mdframed}[2011/09/11]
```

Så får man en fejl hvis pakken er ældre end den angivne dato.

Vi får brug for 2 + 1 ting fra mdfamed: `\newmdtheoremenv` samt `\mdfdefinestyle`. Den første makro har syntaksen

```
\newmdtheoremenv[<mdframed options>]{%
  {<env-navn>}[<fælles tæller>]}{<thm-navn>}[<dominerende tæller>}
```

– dvs. samme syntaks som `\newtheorem`, nu kan man bare tildele den mdfamed-konfiguration

Sætningskonstruktioner med ramme eller farvet baggrund

Den næste makro

```
\mdfdefinestyle{<navn>}{<mdframed options>}
```

er en metode til at give et navn til en fælles mdframed-konfiguration. Så kan man nemt tildele denne til hver sætningskonstruktion, uden at forstyrre andre mdframed-konstruktioner andre steder i dokumentet.

Den tredje ting vi skal bruge er relevant hvis vi anvender ntheorem-pakken, så kan vi nemlig anvende

```
\usepackage[ntheorem]{mdframed}[2011/09/11]
```

Eksempel
5.28

og den ekstra luft i [Eksempel 5.26 på modstående side](#) er væk.

En rigtig interessant fordel ved mdframed-pakken, er at når en kasse deles over et sideskift, så vil en ramme fremstå som *åben*, det er med andre ord en fin løsning til at fortælle læseren at kassen fortsætter på næste side.

Lad os så se på et eksempel.

```
\usepackage[ntheorem]{mdframed}
\usepackage{ntheorem,xcolor}
\definecolor{shadecolor}{gray}{0.9}
\definecolor{rulecolor}{gray}{0.4}
\mdfdefinestyle{thm}{ % saving some space
  skipabove=1em plus 0.4em minus 0.6em,
  skipbelow=0.5em plus 0.4em minus 0.2em,
  leftmargin=-7pt, rightmargin=-7pt, innerleftmargin=6pt,
  innerrightmargin=6pt, innertopmargin=3pt, innerbottommargin=3pt,
  linewidth=1pt, linecolor=rulecolor, backgroundcolor=shadecolor,
  splittopskip=1.2em minus 0.2em,
  splitbottomskip=0.5em plus 0.2em minus 0.1em,
}
\newmdtheoremenv[style=thm]{thm}{Sætning}
\theoremstyle{nonumberplain}
\newmdtheoremenv[style=thm,backgroundcolor=white]{thmstar}{Sætning}
```

Eksempel
5.29



— slut preamble —

```
\begin{thm}
  En nummereret sætning.
\end{thm}
tekst tekst tekst
\begin{thmstar}
  En ikke-nummereret sætning.
\end{thmstar}
```

Sætning 1 *En nummereret sætning.*

tekst tekst tekst

Sætning *En ikke-nummereret sætning.*

Nærmere detaljer om konfigurationen kan ses i [Daniel og Schubert \(2010\)](#). Her har vi sørget for at rammen rager ud i marginen, men sådan at teksten fortsat følger resten.⁸

⁸ Det kan være en smule tricky at styre luften over og under.

5. Om sætningskonstruktioner



I denne bog anvendes en lidt anden konstruktion til at lave de farvede kasser. Denne konstruktion har nogle features som jeg har foreslået indbygget i `mdframed`, nemlig at man skal kunne tilføje ekstra kode indeni selve sætningskonstruktionen.

Hvorfor nu det?

I bogens tilfælde: (a) Fordi vi gerne vil have skiftet baggrundsfarven for kildekoden i eksemplerne til noget som passer bedre (gøres via `\colorlet`), samt (b) makroen der anvendes til at markere pakkenavne etc. kan også skrive ude i marginen, men man skal bruge en speciel metode for at kunne gøre det inde fra en kasse. Så her sørger vi for at ændre denne makro.

For kompletthedens skyld har vi inkluderet en makro som netop kan det samme som `\newmdtheoremenv`.

Sidebemærkning 5.15. Koden i Kodetip 5.1 er *ikke* inkluderet i `dfltxbcodetip`, pakken `loader` heller ikke selv `mdframed`, `ntheorem` eller `xcolor`

Kodetip

5.1



```
\usepackage{mdframed,xcolor}
\definecolor{shadecolor}{gray}{0.9}
\definecolor{rulecolor}{gray}{0.4}
\mdfdefinestyle{thm}{ % saving some space
  skipabove=1em plus 0.4em minus 0.6em,
  skipbelow=0.5em plus 0.4em minus 0.2em,
  leftmargin=-7pt, rightmargin=-7pt, innerleftmargin=6pt,
  innerrightmargin=6pt, innertopmargin=3pt, innerbottommargin=3pt,
  linewidth=1pt, linecolor=rulecolor, backgroundcolor=shadecolor,
  splittopskip=1.2em minus 0.2em,
  splitbottomskip=0.5em plus 0.2em minus 0.1em,
}
```

Vi sørger for at kasserne går ud i marginen, således at teksten indeni fortsat flugter med resten af teksten. De forskellige settings er forklaret i [Daniel og Schubert \(2010\)](#).

Så til den lidt mere 'sjove' del. I Kodetip 5.2, skal man bemærke brugen af `\DeclareDocumentCommand`. Dette er en meget nyttig makro fra `xparse`-pakken, som gør en i stand til at lave makroer med meget specielle blandinger af obligatoriske og valgfri argumenter. Sammenligner man koden nedenfor med koden til den gamle konstruktion vil man se at den nye er meget nemmere at forstå.

Konstruktioner som starter med en liste

```
\usepackage{xparse,etoolbox}
\usepackage[amsmath,thmmarks]{ntheorem}
\DeclareDocumentCommand\NewShadedTheorem{ o m o m o }{
  % #1 = options for mdframed
  % #2 = env name
  % #3 (optional) = common counter name
  % #4 = header text
  % #5 (optional) dominant counter name
  \IfNoValueTF{#5}{% not dominated
    \IfNoValueTF{#3}{% do not use common counter
      \newcounter{#2}
      \newtheorem{#2inner}[#2]{#4}
    }{% use common counter
      \newtheorem{#2inner}[#3]{#4}
    }{% use dominated
      \newcounter{#2}[#5]
      \newtheorem{#2inner}[#2]{#4}
    }
  }
  % create the environment
  \newenvironment{#2}{%
    % add extra code here
    \IfNoValueTF{#1}{\mdfsetup{#1}}{% this is local
      \begin{mdframed}
        \ifdeflength{\theorempreskipamount}{%
          \setlength\theorempreskipamount{0pt}%
          \setlength\theorempostskipamount{0pt}}{}
        \begin{#2inner}}{\end{#2inner}\end{mdframed}}
  }
}
```

Kodetip

5.2



Så er det bare at anvende

```
\NewShadedTheorem[style=thm]{thm}{Sætning}
\theoremstyle{nonumberplain}
\NewShadedTheorem[style=thm,
  backgroundcolor=white]{thmstar}{Sætning}
\begin{thm}
  En nummereret sætning.
\end{thm}
tekst tekst tekst
\begin{thmstar}
  En ikke-nummereret sætning.
\end{thmstar}
```

Sætning 1 *En nummereret sætning.*

tekst tekst tekst

Sætning *En ikke-nummereret sætning.*

Eksempel

5.30



Konstruktionen burde også virke fint sammen med `amsthm`, evt. skal man så lige smide et par ekstra nulstillinger af længder ind i konstruktionen.

Sidebemærkning 5.16. Man bedes bemærke at `\NewShadedTheorem` ikke giver en stjernet environment som `ntheorems` udgave af `\newtheorem` normalt gør (denne stjernede version avnendes i forbindelse med sætningslister, hvilket ikke er noget vi vil gå i detaljer med her).

Har man brug for det så er det ikke noget problem at udvide `\NewShadedTheorem` til også at inkludere denne.

5.4.2 Konstruktioner som starter med en liste

Når en sætning starter med en liste (og man ikke har anvendt `\theoremstyle{break}`) da vil første punkt på listen befinde sig på samme linie som overskriften, hvilket ikke er pænt, se for [Eksempel 5.32 på næste side](#). Har man mange af disse kan man overveje

5. Om sætningskonstruktioner

at lave en variant af ens sætningsenvironment, som er lavet med `\theoremstyle{break}`.
Eksempel

Eksempel 5.31

```
\usepackage[amsmath, thmmarks]{ntheorem}
\newtheorem{thm}{Sætning}
\theoremstyle{break}
\newtheorem{thmbreak}[thm]{Sætning}
\theoremstyle{plain}
...
```

Hvis man kun har nogle få, kan man anvende følgende lille makro:

Kodetip 5.3



```
\makeatletter
\newcommand\InsertTheoremBreak{%
  \@ifstar{\item[\vbox{\null}]}{%
    \begingroup % keep changes local
    \setlength\itemsep{0pt}%
    \setlength\parsep{0pt}%
    \item[\vbox{\null}]%
    \endgroup%
  }}
\makeatother
```

`\InsertTheoremBreak` har visse begrænsninger (i underlige kombinationer med `\displaystyle`) men det har `\theoremstyle{break}` også, så det er en nem lille brugbar makro. Nedenfor finder man et eksempel (vi har gemt preambleen for at spare plads).

Eksempel 5.32



```
\begin{thm}
  \begin{enumerate}
    \item nummer et
    \item nummer to
    \item \dots
  \end{enumerate}
\end{thm}
\begin{thm}
  \InsertTheoremBreak
  \begin{enumerate}
    \item nummer et
    \item nummer to
    \item \dots
  \end{enumerate}
\end{thm}
```

Sætning 1 1. nummer et
 2. nummer to
 3. ...

Sætning 2
 1. nummer et
 2. nummer to
 3. ...

Sidebemærkning 5.17. Det skal dog lige nævnes at konstruktionen ovenfor desværre er lidt skrøbelig hvis den kommer tæt på et sideskift. Her kan man altså risikere at sætningsoverskriften kommer på en siden mens resten kommer på næste side.

En anden måde at komme om ved dette er f.eks. at have flere forskellige thm environments. F.eks. defineret med og uden `ntheorem's break` stil.

5.4.3 Mere om optimeringseksemplet

alignat

I [Eksempel 4.28 på side 93](#) viste vi en anvendelse af *alignat* til en traditionel optimeringsopgave. Opbygningen i dette eksempel er den især studerende fra Matematik-Økonomi (operationsanalyse) anvender, sandsynligvis fordi det er sådan det er skrevet i deres bøger. Personligt synes jeg udtrykket er lidt mangelfuldt især hvis hele specialet er bygget op omkring analysen og løsningen af dette (eller flere) problem(er), i

Mere om optimeringseksemplet

dette tilfælde burde udtrykket (efter min mening) fremhæves noget mere. Mit forslag forklaring følger i [Eksempel 5.33](#).

<pre>\usepackage{amsmath} \usepackage[amsmath]{ntheorem} \theoremseparator{.} \newtheorem{problem}{Problem} ----- slut preamble ----- \begin{problem}\label{problem1} Minimer følgende udtryk \begin{equation}\label{eq:prob1} f(x,y,x,w)+\Omega(\xi) \end{equation} under hensyntagen til \begin{subequations} \label{eq:prob1-side} \begin{align} &\delta x[2cm]\label{eq:prob1a}\\ &\delta x[2cm]\label{eq:prob1b}\\ &\delta x[2cm]\label{eq:prob1c}\\ &\delta x[2cm]\label{eq:prob1d} \end{align} \end{subequations} \end{problem} Vi kan så henvise til Problem~\ref{problem1} som vores overskyggende hovedproblem. Til \eqref{eq:prob1} som udtrykket vi skal undersøge, til \eqref{eq:prob1-side} for det overordnede nummer for sidebetingelserne og til en enkel betingelse~\eqref{eq:prob1c}.</pre>	<p>Problem 1. <i>Minimer følgende udtryk</i></p> $f(x,y,x,w) + \Omega(\xi) \quad (1)$ <p><i>under hensyntagen til</i></p> <table style="margin-left: auto; margin-right: auto;"><tr><td style="border: 1px solid black; width: 100px; height: 15px;"></td><td style="text-align: right;">(2a)</td></tr><tr><td style="border: 1px solid black; width: 100px; height: 15px;"></td><td style="text-align: right;">(2b)</td></tr><tr><td style="border: 1px solid black; width: 100px; height: 15px;"></td><td style="text-align: right;">(2c)</td></tr><tr><td style="border: 1px solid black; width: 100px; height: 15px;"></td><td style="text-align: right;">(2d)</td></tr></table> <p>Vi kan så henvise til Problem 1 som vores overskyggende hovedproblem. Til (1) som udtrykket vi skal undersøge, til (2) for det overordnede nummer for sidebetingelserne og til en enkel betingelse (2c).</p>		(2a)		(2b)		(2c)		(2d)
	(2a)								
	(2b)								
	(2c)								
	(2d)								

Eksempel 5.33



Af [Eksempel 5.33](#) skulle man gerne observere:

- (i) Man har mulighed for at give hele problemet en overskrift og et nummer således man kan henvise til hele problemet.
- (ii) Man har mulighed for at henvise til det udtryk man skal optimere.
- (iii) Man har mulighed for at henvise til sidebetingelserne samlet eller hver af betingelserne for sig.

Denne adskillelse er (af tekniske årsager) ikke mulig når man samler det hele med *alignat* eller lignende. Jeg synes at den frihed vi her har demonstreret giver helt andre muligheder for opbygningen af specialet end man normalt har, med notationen fra [Eksempel 4.28 på side 93](#).

Om grafik, flydende objekter og billedtekst

Nye L^AT_EX-brugere har i starten ofte problemer med inklusion af ekstern grafik og med begrebet *floats*, især fordi der er begrænsninger på hvilke typer af grafik man kan anvende og fordi floats er meget anderledes i forhold til det man måske er vant til. Dette kapitel skulle gerne afhjælpe noget af denne forvirring.

Bare for ordens skyld gør vi opmærksom på at vi af pladshensyn ikke kommer videre meget ind på hvordan man kan lave grafik direkte i L^AT_EX, eller med de eksterne grafiske redskaber som ligger nær L^AT_EX,¹ vi vil dog skrive lidt.



Formaning 6.1 (Om fremvisning af DVI-formatet).

DVI-formatet er nok ikke kendt af så mange nye brugere – de fleste anvender PDF. DVI har dog været standard output formatet for L^AT_EX siden firserne, pdfL^AT_EX kom først frem i slutningen af halvfemserne.

Vælger man at fremvise en DVI-fil, skal man huske på at inkluderet ekstern grafik, sandsynligvis ikke vil fremvises korrekt i en DVI-fremviser. Dette er helt normalt. Er der noget specielt man skal tjekke, så oversæt videre til et format bedre egnet til gennemsyn, f.eks. PDF.

Det samme gør sig gældende hvis man anvender effekter så som skalering (f.eks. via `\scalebox`) eller rotering (`\rotatebox`).



Formaning 6.2 (Filnavne).

Det er generelt en *meget* dårlig ide at have mellemrum i filnavne eller biblioteksnavne. Erstat mellemrummet med en »_« i stedet. Der er mange programmer som har problemer med mellemrum i filnavne. dvips har i hvert fald tidligere haft problemer med det (ved ikke om man har fået det rettet endnu). Men generelt anbefales at man *aldrig* anvender mellemrum i filnavne.

Man bør heller ikke anvende special-tegn eller æøå i filnavne, så fungerer tignene bedst på tværs af operativ systemer. Faktisk kan det være en god ide altid at anvende små bogstaver til navne til grafik-filer, thi Windows er ligeglad, men det er Mac eller Linux ikke.

6.1 Inklusion af ekstern grafik

Som allerede nævnt er grafik det mest forvirrende punkt når man anvender L^AT_EX. Det er forskelligt hvad de forskellige L^AT_EX-versioner kan håndtere.

¹ Der er et par stykker som f.eks. anvender L^AT_EX til at formatere labels.

6. Om grafik, flydende objekter og billedtekst



Advarsel – teknisk forklaring 6.3. Det er i og for sig *ikke* L^AT_EX selv som inkluderer et stykke eksternt grafik. (pdf)L^AT_EX er ligeglad, den skal bare vide hvor stort et område den skal give plads til. Der efterlades en markør i filen og en senere driver sørger så for selve indlejringen i outputtet. I visse tilfælde sker dette først ved en konvertering til et andet output-format, f.eks. dvips fra DVI til PostScript eller dvipdfmx fra DVI til PDF. I andre tilfælde er driveren bygget helt ind i selve programmet, dette sker f.eks. med pdfL^AT_EX.



Vi anbefaler generelt brugen af pdfL^AT_EX, i denne sammenhæng fordi det er den L^AT_EX som understøtter flest grafik typer (se den senere oversigt), samt fordi de fleste kan forholde sig til PDF-formatet.

Under alle omstændigheder skal man bruge pakken `graphicx` for at kunne inkludere eksternt grafik.² Man vil finde information vedrørende `graphicx` i [Carlisle \(1999b\)](#) samt [Reckdahl \(2005\)](#). Loades via:

```
\usepackage[options]{graphicx}
```

Tip 6.4 (Vedr. options til graphicx).

options anvendes til forskellige ting.

Drivere. Disse vælger den normalt selv, vælger den selv en specifik driver når den genkender pdfL^AT_EX. Ved almindelig L^AT_EX antager `graphicx` automatisk at man bagefter ønsker at anvende dvips. Men her er der også muligheden dvipdfmx! I det (og kun dette) tilfælde *skal* man angive driveren »dvipdfmx«, ellers skal man ikke angive driver.

Undlad at inkludere noget. Ved visse typer af billeder (f.eks. hvis de er meget store) eller ved langsomme computere, kan det være en fordel at oversætte uden faktisk at inkludere billedet i outputtet. Her anvendes option »draft«, enten til `graphicx` eller til klassen selv. Så erstattes alle inkluderede billeder med en kasse af samme størrelse.

Oversæt fil hvor man ikke har billederne. Angiv option »demo«. Så erstattes alle billeder af en sort kasse af en fast størrelse. Dette påvirker naturligvis sideombrydningerne, men filen kan nu oversættes.

De primære drivere (sådan ca.) og hvilke grafikformater de understøtter er:

pdftex – »driveren« associeret med pdf`latex`, den sørger altså for at inkluderingen af figuren sker direkte sammen med oversættelsen. Denne driver understøtter: `JPG`, `PNG`, `PDF`, `MPS`³ og `JBIG2`.

Vi kommer senere ind på at det kan være en fordel at angive filen som skal inkluderes (syntaks kommer vi til om lidt) *uden* endelse. pdfL^AT_EX leder så efter filen ved at prøve en serie af endelser, den *første* som passer vælges. Standard listen svarer til at man havde skrevet følgende i preamble

² Visse 'ældre' brugere, samt gamle skabeloner anvender pakker så som `epsfig` eller `psfig`, begge pakker er forældede, og man bør i stedet gå over til at anvendes `graphicx`.

³ Output fra METAPOST.

Syntaks for inkludering af ekstern grafik

```
\DeclareGraphicsExtensions{%  
  .png, .pdf, .jpg, .mps, .jpeg, .jbig2, .jb2, %  
  .PNG, .PDF, .JPG, .JPEG, .JBIG2, .JB2%  
}%
```

Eksempel
6.1

Denne kan man så skrive om hvis man ønsker en anden rækkefølge.

`dvips` – kan kun klare EPS-formatet. En EPS-fil er en delmængde af POSTSCRIPT hvor det bl.a. er påkrævet at der er en

```
%%BoundingBox <tal> <tal> <tal> <tal>
```

linie i filen. Tallene angiver koordinaterne til hhv. nederste venstre- og øverste højre hjørne af figuren. L^AT_EX læser denne linie for at beregne hvor stort et område den skal reservere til filen.

Faktisk antager `dvips`-driveren at *hvad som helst* skal anses som en EPS-fil, så endelsen er i dette tilfælde ligegyldig.

`dvipdfmx` – en efterbehandler til almindelig L^AT_EX på linie med `dvips` men bare fra DVI direkte til PDF. Denne understøtter faktisk alle formaterne fra `dvips` og `pdftex` tilsammen.⁴ Dog med en lille hage:

For at kunne inkludere JPG, PNG eller PDF skal L^AT_EX vide hvilken BoundingBox (dvs. størrelse) figuren har og da L^AT_EX ikke kan læse disse tre formater skal brugeren hjælpe. Man kører det lille program `ebb` på hver figur. Dette skaber for filnavn.xxx filen filnavn.bb hvilket så anvendes som BoundingBox. For mere info se [Wicks \(1999\)](#).

ebb

Sidebemærkning 6.5. XeL^AT_EX (som vi ikke behandler nærmere), kan vist håndtere en sammensurium af de formater som er nævnt ovenfor.

Sidebemærkning 6.6 (Vejen til PDF når der er fejl).

Ovenfor kan vi se at der er mindst tre forskellige måder at komme frem til en PDF-fil på: `pdflatex`, `latex + dvips + ps2pdf` eller `latex + dvipdfmx`.

Når man får grafikrelaterede fejl kan det derfor være ret vigtigt at have styr på hvilken af de tre veje ens editor har valgt at lægge bag en L^AT_EX→PDF-knap.



Når vi nu anbefaler at man arbejder med PDF som slutresultat, så kan se være en fordel at anvende en anden fremviser end Adobe Reader, se afsnit C.3 på side 393.

6.1.1 Syntaks for inkludering af ekstern grafik

Selve inkluderingen af figuren laves via:

```
\includegraphics[<options>]{<filnavn>}
```

```
\includegraphics[width=.5cm]{fig.2}
```



Eksempel
6.2

`<options>` er en kommasepareret liste af nøgleord eller »key=value«-par:

⁴ Dog ikke klar over om JBIG2 understøttes.

6. Om grafik, flydende objekter og billedtekst

`angle = <grader>`

rotering gennem `<grader>` mod uret. Omdrejningspunktet vil normalt være nederste venstre hjørne af figurboksen, kan ændres via `origin`.

Rækkefølgen af denne og de andre options er *ikke* ligegyldig, se Bemærkning 6.8 på næste side.

`origin = <x>`

hvor `<x>` kan være »c« som giver rotation omkring centrum af figuren. Se Carlisle (1999b) afsnit 4.2 eller Reckdahl (2005) afsnit 8.3 for flere muligheder til `origin`.

`width = <længde>`

specificerer bredden af figuren.

`height = <længde>`

specificerer højden (over baseline) af figuren.

`totalheight = <længde>`

specificerer den samlede højde og dybde af figuren. Mest anvendelig hvis man roterer figuren og denne rotation ikke er delelig med 90°.

Sidebemærkning 6.7. Når man angiver højde eller bredden kan det være en god ide at man angiver den i termer af tekstbredden eller -højden eller nuværende liniebredde. Dvs. via

```
\textwidth \textheight \linewidth
```

Så kunne man f.eks. anvende

```
\includegraphics[width=0.7\textwidth]{fig.2}
```

og dermed skalerer figuren med dokumentet hvis man senere beslutter sig for at ændre marginerne.

Eksempel 6.3

`scale = <tal>`

skalér figuren med faktoren `<tal>` på begge ledter.

`keepaspectratio = <true|false>`

at angive den uden nogen værdi er det samme som `true`.

Er både `width` og `height` angivet kan figuren komme til at se ret underlig ud.

Eksempel 6.4

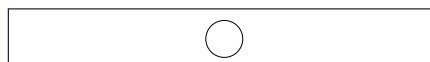
```
\includegraphics%  
[width=3cm,height=5mm]{fig.2}
```



Med `keepaspectratio` vil figuren blive skaleret (med samme faktor på begge ledter) til den maksimale størrelse der kan være uden at overskride den angivne højre eller bredde.

Eksempel 6.5

```
\includegraphics[width=3cm,  
height=5mm,keepaspectratio]{fig.2}
```



`clip = <boolean>`

Klipper figuren til `BoundingBox`'en. Anvendelig hvis dele af figuren faktisk rager udenfor `BoundingBox`'en. Dette ses til tider ved plots fra visse typer programmer.

Yderligere konfiguration vedrørende grafik

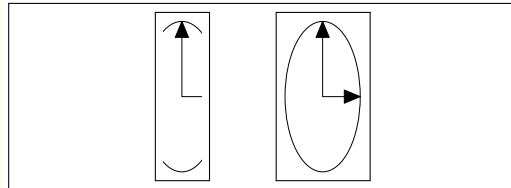
`viewport = <llftx llfty urtx urty>`

(nederre venstre (x, y) og øverste højre (x, y)) anvendes til at ændre den del man ser af figuren. Koordinaterne angivet her er regnet ud fra nederste venstre hjørne af figuren og man arbejder i tal angivet uden enhed (anvendes der ingen enhed, antages bp (big points), hvorom der gælder at 72 bp er lig 1 in). Kan være smart at kombinere med `clip`.

`trim = <left bottom right top>`

i samme enheder som ovenfor, her vil man nu skære denne mængde væk fra billedet. Dvs. »`trim=10 0 10 0`« vil skære 10bp væk fra billedet i venstre og højre side. Bør igen kombineres med `clip`, se [Eksempel 6.6](#). Bemærk at man kan godt anvende enheder i argumentet til trim, i.e. »`trim=0 1em 2cm 1mm`«.

```
\fbox{\includegraphics
[height=2cm,trim=15 0 15
0,clip]{fig.1}}
\qqquad
\fbox{\includegraphics
[height=2cm]{fig.1}}
```



Eksempel
6.6

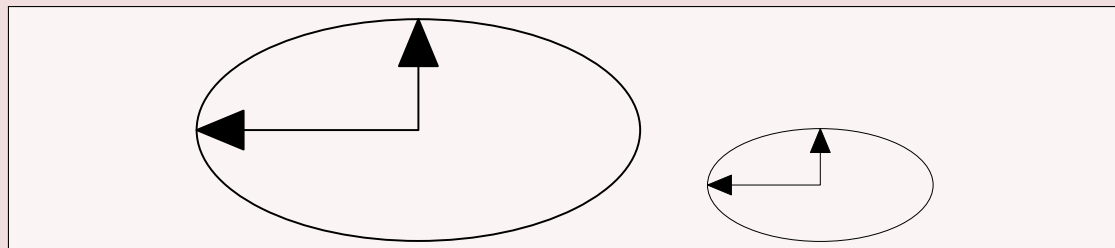
`page = <num>`

denne option virker *kun* når man arbejder under pdfL^AT_EX, og kan så anvendes til at angive en specifik side fra en ekstern PDF-fil.

Sidebemærkning 6.8 (Rækkefølgen af options er *ikke* ligegyldig!).

Det er værd at bemærke at `\includegraphics` fortolker options fra venstre mod højre, i.e. `angle=90,width=2cm` er *ikke* det samme som `width=2cm,angle=90`. I det første tilfælde bliver den resulterende *bredde* 2 cm, mens det i det andet bliver en resulterende *højde* på 2 cm.

```
\includegraphics[width=3cm,angle=90]{fig.1} \qqquad
\includegraphics[angle=90,width=3cm]{fig.1}
```



Eksempel
6.7

6.1.2 Yderligere konfiguration vedrørende grafik

En kildefil – flere outputformater

Det kan være rart at kunne køre både latex med dvips samt pdflatex på den samme kildefil. Det gør vi blandt andet med denne bog, thi det er hurtigere at fremvise POSTSCRIPT end PDF. Men har man både en EPS og en PDF version af den samme figur kan systemet jo ikke finde ud af det når man angiver filendelsen som del af filnavnet. Løsning: Lad være med at angive fil endelsen! Ergo kan man i stedet for

```
\includegraphics{filnavn.eps}
```

6. Om grafik, flydende objekter og billedtekst

anvende

```
\includegraphics{filnavn}
```

forudsat man så har både en .eps og en .pdf udgave (eller et af de andre formater som pdfL^AT_EX understøtter).

Sidebemærkning 6.9. Der er dog iflg. [Reckdahl \(2005\)](#) en mindre catch. Når man fjerner endelsen (inklusive ».«) må der i den resterende del af filnavnet *ikke* indgå punktum.

grffile Dette kan dog omgås via pakken `grffile` som udvider den måde filnavne parses på. Den kan endda tillade filnavne med mellemrum, se [Oberdiek \(2009\)](#).⁵

Sidebemærkning 6.10 (METAPOST og filendelser).

I og med METAPOST figurer ofte har ».1«, ».2«,... som endelse, bør man her *ikke fjerne endelsen!* Se iøvrigt [Eksempel 6.8](#).

METAPOST inklusion med pdf_latex

Når man arbejder med METAPOST vil de resulterende figurer ofte have filendelser af typen .1, .2, ...⁶ Men pdf_latex forventer jo endelsen .mps for METAPOST-filer. Man løser dette problem ved at forklare pdf_latex at de filendelser den ikke genkender skal den formode er METAPOST.

```
\DeclareGraphicsRule{*}{mps}{*}{}
```

Desværre er verden ikke så simpel. For med latex mener driveren jo at alt er EPS-formatet, og latex vil så brokke sig over at den ikke kender noget til mps. Løsningen er alt i alt at vi skal tjekke om vi skal lave PDF, i så fald inkluderer vi linien ovenfor, ellers lader vi værre. Anvend pakken `ifpdf` (direkte indbygget i *memoir*-klassen):

Eksempel 6.8

```
\usepackage{ifpdf}
\ifpdf
  \DeclareGraphicsRule{*}{mps}{*}{}
\fi
```

At hente figurer fra andre biblioteker

Hvis nu man f.eks. er statistiker⁷ og gerne vil inkludere en hulens masse SAS-figurer, kan det være en fordel at placere disse i et bibliotek for sig selv. Man kan så enten anvende relativ- eller fuld-sti når man specificerer filnavnet, *eller* man kan forklare L^AT_EX, hvor der skal ledes efter figurerne ud over nuværende bibliotek (og resten af L^AT_EX-installationen).

```
\graphicspath{{biblioteksstier med {} omkring}}
```

⁵ Det er dog bedre stil at undgå mellemrum i filnavne, erstat dem med bindestreg eller underscore.

⁶ Det er muligt at ændre dette, se METAPOST-manualen.

⁷ No pun intended.

Inkludere dele af (pdf-) dokumenter

Eksempel (her er det underbiblioteker af nuværende)

```
\graphicspath{{figurer/}{plots/}}
```

Eksempel
6.9

På Windows skal man anvende den samme syntaks, dvs. man skal *ikke* anvende syntaksen »c:\foo\bar« på Windows, anvend syntaksen med de fremadrettede stråstreger.

Skal man nu fremvise fig.eps fra ./figurer kan man nøjes med \includegraphics{fig} uden at specificere hvilket bibliotek fig.eps befinder sig i (altså uden at skulle angive \includegraphics{figurer/fig}). Man skal selvfølgelig passe på med at have flere filer med det samme navn liggende i de forskellige biblioteker, L^AT_EX vil tage den første den finder.

Sidebemærkning 6.11. Anvendelsen af \graphicspath har også den helt store fordel at sender man filerne til en anden skribent, behøver denne ikke lave samme biblioteksstruktur som man selv har, han/hun kan vælge at placere alle figurerne i samme bibliotek som .tex-filen, og det vil stadigvæk kompilere uden man skal ændre noget i .tex-filen.

6.1.3 Inkludere dele af (pdf-) dokumenter

Af tekniske årsager kan man ikke inkludere en serie af sider fra et PostScript dokument, eller for den sagsskyld, specificere hvilken side fra en PostScript fil, man ønsker at inkludere.

Men det er muligt at inkludere dele af eller hele PDF-dokumenter ved brug af pdfL^AT_EX. Anvend pakken pdfpages og inkludér dokumentet med

pdfpages

```
\includepdf[options]{pdf-fil}
```

Eksempelvis vil

```
\includepdf[pages=-]{fil.pdf}
```

Eksempel
6.10

inkludere *alle* sider i fil.pdf, mens pages={2-5,23-34} vil inkludere et udvalg af sider. Der er mange andre options til at styre inkluderingen, bl.a. skalering af siderne, rotation, flere sider på en side, se [Matthias \(2004\)](#) for yderligere information.

Sidebemærkning 6.12. pdfpages virker *kun* sammen med pdf_latex!

Der findes intet reelt PostScript alternativ til pdfpages. Virkningen kan emuleres via eso-pic-pakken, en masse enkeltsidede PostScript-filer, samt noget knofedt.

Sidebemærkning 6.13. pdfpages kan rigtig mange ting, så som at printe flere sider på en side. Men den kan også skrive ting ovenpå det som den har inkluderet. Man kan anvende en speciel pagestyle på de inkluderede sider samt evt. tilføje ting til masterdokumentets indholdsfortegnelse. Det sidste kigger vi nærmere på i afsnittet »[Et par ideer vedr. ph.d.-afhandlinger](#)« på side 323

Sidebemærkning 6.14 (Flette PDF-filer).

En anden ting `pdfpages` også kan bruges til er hvis man har en masse enkelte PDF-filer som skal printes. Det er også mere venligt overfor andre brugere af en given printer.

Et simpelt dokument til at flette PDF-filer med kan se ud som følger

Eksempel

6.11



```
\documentclass[a4paper]{article}
\usepackage{pdfpages}
\begin{document}
\includepdf[pages=-]{fil1.pdf}
\includepdf[pages=-]{fil2.pdf}
\includepdf[pages=-]{fil3.pdf}
\end{document}
```

6.2 Flydende objekter

Indtil videre har de figurer vi har indsat bare blevet placeret på netop det sted vi har sat dem ind. Hvis der ikke er plads til dem på siden bliver de bare flyttet om på næste side, efterladende et stort hul i teksten. Dette er normalt *ikke* det vi ønsker, desuden ville vi gerne kunne henvise til vore figurer samt eventuelt skrive en lille tilhørende tekst (caption). Til at løse dette anvender man begrebet *floats*!

floats

En float er en boks (som højest kan fylde *en* side!) hvori man har sin figur(e) eller tabel(ler) (samt en eventuel tilhørende tekst) og denne boks vil så blive placeret af L^AT_EX, på det sted den synes der er bedst plads – boksen siges at flyde nogenlunde frit. Fordelen ved dette er naturligvis, at man ikke får huller i teksten, papiret bliver bedre udnyttet. Hvis den flydende figur ikke kan placeres i nærheden af hvor man har indsat den, vil figuren blive placeret et andet sted. Teksten som kommer efter floatkoden som indsatte figuren fortsætter med det samme i stedet for at vente til figuren er sat ind.

Ulemperne er at man lige skal vænne sig til float-konceptet, man skal lære at skrive sin tekst på en anden måde, dvs. ikke længere skrive: »På figuren herover ses...«, thi man ved ikke hvor figuren indsættes. I stedet skal man henvise til figuren via en krydsreference. Pakken `varioref` kan være en stor hjælp, se afsnit 11.4 på side 298, den kan bl.a. sørge for at der automatisk kommer et sidetal med i referencen hvis det er nødvendigt (referencen herover til afsnit 11.4 er lavet via `varioref`).

varioref

En anden hage er at det er ret nemt at komme til at spærre indsættelsen af figurerne, således at en stor float kan blokere for en masse små som er indsat efter (floats indsættes i teksten i den rækkefølge man finder dem i kildekoden). Dette betyder at man skal være en smule omhyggelig med hvordan man konfigurerer sit flydende objekt. Man skal desuden huske at en flydende figur eller tabel må ikke fylde mere end én side (vi kommer senere ind på hvad man gør med tabeller som fylder mere end én side).

Alt dette skal vi se nærmere på i det følgende.

6.2.1 Syntaks

figure
table

Normalt har L^AT_EX to environments som flyder: *figure* og *table*.⁸ Eneste forskel på de to er, at anvender man en billedtekst (caption) til dem vil den ene skrive »Figur« og den anden »Tabel« (og de vil anvende to forskellige tællere til det eventuelle nummer). Vi vil anvende *figure* her, al syntaks og alle features fås også med *table*. Syntaks:

⁸ Man kan sagtens selv lave flere, men det forklarer vi ikke i denne bog.

Syntaks

```
\begin{figure}[\langle loc \rangle]
\centering
\langle figurkode \rangle
\caption{\langle tekst \rangle}
\label{\langle key \rangle}
\end{figure}
```

Vi starter med nogle bemærkninger til syntaksen

Sidebemærkning 6.15.

- (i) For god ordens skyld starter vi med at minde om at `\label` skal komme *efter* `\caption` og at det er `\caption` som giver figuren sin nummering (hvis man ønsker en).
- (ii) Man kan naturligvis også placere `\caption` over figuren (ses sjældent), men anvendes ofte ved tabeller. Her kan det blive aktuelt at skulle omdefinere spacingen omkring `\caption`, se afsnit *Mellemrum før og efter billedtekst* på side 152.
- (iii) Generelt gælder at `\caption` kun kan anvendes indeni floats, og dog, `\caption` kan emuleres, se afsnit *En figur og en tabel i samme float* på side 153.
- (iv) Der findes også en *-t udgave af de to float environments. Denne variant anvendes når man arbejder i een dokumentopsætning med to søjler og vil så give en float som kommer i toppen af papiret og som spænder over begge søjler.
- (v) `\langle loc \rangle`-argumentet anvendes for at hjælpe L^AT_EX med at bestemme hvor figuren skal placeres. Argumentet er en tekststreng lavet ud fra tegnene !, h, t, b og p. Rækkefølgen af disse er ligegyldig. Angiver man ikke noget valgfrit argument [`\langle loc \rangle`] så vil »tbp« blive anvendt i de fleste klasser (bemærk at »h« mangler). Se Tip 6.16 for hvordan man ændrer dette og dermed sparer sig selv for en masse skriveri.

Tip 6.16. Hvis man bliver træt af manuelt at skulle skrive »[htbp]« ved alle floats, så kan man faktisk gøre det automatisk. Den følgende makro findes allerede i *memoir*, vi anvender derfor `\providecommand` for kun at tilføje makroen hvis den ikke findes i forvejen.

```
\makeatletter
\providecommand*\setfloatlocations[2]{\@namedef{fps@#1}{#2}}
\makeatother
\setfloatlocations{figure}{htbp}
\setfloatlocations{table}{htbp}
```

Så behøver man kun angive argument til f.eks. *figure*-environmentet hvis man vil have noget *andet* end »[htbp]«, f.eks. hvis man vil tvinge en float til ikke at blive en helside float.

Det kan evt. anbefales at man helt fjerner »b« i større dokumenter, da det ikke altid er lige pænt med floats nederst på siden.

Man kan finde yderligere information vedrørende de størrelser som styrer placeringen af floats i [Reckdahl \(2005\)](#), afsnit 17.

Da der er rigtigt mange som har problemer med at forstå placeringen af flydende objekter, eller som har problemer med at de flyder for langt væk, så vil vi gå i lidt mere teknisk detalje vedr. hvordan L^AT_EX placerer flydende objekter.

Kodetip

6.1



6. Om grafik, flydende objekter og billedtekst

6.2.2 Algoritmen bag placeringen af flydende objekter

Først er der nogle interne parametre.

`topnumber`

(tæller), antallet af floats som må være øverst på en side. Standard er 2 (*memoir*: 3).

`bottomnumber`

(tæller), tilsvarende for floats i bunden af siden. Standard er 1 (*memoir*: 2).

`totalnumber`

(tæller), totale antal flydende enheder som må være på én side. Standard er 3 (*memoir*: 3).

`\topfraction`

makro hvis værdi angiver hvor stor en brøkdel af siden må *maksimalt* udfyldes af en float placeret i toppen af siden. Standarden er 0.7 (*memoir*: 0.85). Makroen ændres med `\renewcommand`.

`\bottomfraction`

tilsvarende for floats placeret i bunden af siden. Standard er 0.3 (*memoir*: 0.5).

`\textfraction`

er en makro hvis værdi angiver den *minimale* brøkdel af en almindelig side der skal være 'tekst'. Standard er 0.2 (*memoir*: 0.1), dvs. i *memoir*-klassen skal der på en side med floats og tekst (ergo ikke en ren float side) være mindst 10% tekst.

`\floatpagefraction`

makro hvis værdi angiver minimumsbrøkdelen af en floatside som skal være udfyldt. Standarden er 0.5 (*memoir*: 0.7).

`\floatsep`

elastiklængde⁹ som styrer afstanden mellem top floats placeret i toppen af siden. Standarden afhænger af dokumentklassen og kan også afhænge af fontstørrelsen.

`\textfloat`

elastiklængde som angiver den vertikale afstand afsat mellem floatområdet i toppen af siden, eller bunden af siden, og den efterfølgende/ovenstående tekst.

`\intextfloatsep`

elastiklængde som specificerer det mellemrum som automatisk indsættes over og under en float som er placeret i teksten.

Dette var nogle af de størrelser som styrer hvordan floats kan placeres på en given side. Lad os så beskrive algoritmen. Vi husker altså at vi har parametrene $!$, h , t , b samt p , og at rækkefølgen er ligegyldig.

- (a) Hvis $!$ er givet og vi har en af h , t , b så ser vi bort fra de tilhørende `\dots fraction` værdier i placeringen af det flydende objekt. Dvs. man kan have en bottom float som er større end det `\bottomfraction` tillader. Dette kan ses om en initialisering af algoritmen.
- (b) Hvis »h« er givet forsøges at placere figuren her (omkring). Hvis dette ikke lykkes og der ikke er givet andre tegn, skiftes fra »h« til »t« (til placering øverst på siden) og placeringen af dette objekt vil nu kun blive betragtet som en top-float, »h« bliver *ikke* betragtet igen.

⁹ Dvs. en længde som har visse muligheder for at strække sig eller trække sig sammen. Afstanden over og under afsnitoverskrifter er et andet eksempel på elastiklængder.

Her til og ikke længere

Derfor, nej, »h«, betyder ikke placér figuren lige her! Det gør »!h« heller ikke!

- (c) Hvis »t« er givet, forsøg at placere figuren øverst på denne side.
- (d) Hvis »b« er givet, forsøg at placere figuren i bunden af denne side.
- (e) Hvis »p« er givet forsøges at placere figuren på en floatside.
- (f) Hvis figuren endnu ikke er placeret gentages punkt (c) og (d) i begyndelsen af hver efterfølgende side, efterfulgt af en gentagelse af (e) ved slutningen af hver efterfølgende side.

Se desuden afsnit 6.2.4 om hvordan man får en float til slet ikke at flyde (hvilket dog sjældent anbefales!).

Vi kan nu se hvorfor »[h]« nemt kan være en dødssynd. Antag vores figur samt billedtekst er større end `\topfraction` og at der ikke er plads til denne float i teksten lige her. Så laves floatens status om til »t«, men figuren er for stor til at figuren kan placeres i toppen, så algoritmen fejler (der er ikke flere muligheder) og udskyder placeringen til siden efter. Her fejler placeringen igen og sådan fortsætter det. Dette objekt ligger nu og spærrer for alle efterfølgende floats som ikke kan placeres før denne er placeret. Figuren placeres nu først når den bliver tvunget til det, i.e. når den f.eks. når et nyt kapitel (som internt anvender `\clearpage` eller `\cleardoublepage`), møder en `\FloatBarrier` eller når dokumentet er slut.

Har man en float som opfører sig på denne måde bør man naturligvis give den nogle flere muligheder for placering, dvs. flere bogstaver til `<loc>`-listen. Eller man kan forsøge med at tilføje et »!«.

6.2.3 Her til og ikke længere

Da L^AT_EX indsætter figurerne i den rækkefølge de er blevet indsat i teksten kan man komme ud for forskellige problemer.

- Hvis der er en stor figur som endnu ikke er blevet placeret spærrer denne for alle de efterfølgende figurer som jo først kan indsættes når den store er placeret.
- Jf. foregående kan man jo sagtens komme ud for at en figur flyder over i det næste afsnit, hvilket til tider ikke lige er det man ønsker.

Løsningen på dette problem er at man skal direkte forlange at L^AT_EX »dumper« alle figurerne inden den går videre.

`\clearpage`

sørger for at alle figurerne indsættes her og nu inden vi går videre. `\clearpage` vil så skifte til en ny side *efter* den sidste float er blevet indsat.

`\cleardoublepage`

som `\clearpage` men her skifter vi til den næste højre (ulige)side efter sidste float.

`\FloatBarrier`

fra pakken `placeins` virker lige som `\clearpage` men den starter ikke en ny side efter sidste float, hvilket kan være en fordel.

`\FloatBlock`

i *memoir*, dette er bare en direkte kopi af `\FloatBarrier`, men kræver ikke `placeins`.

6. Om grafik, flydende objekter og billedtekst

`\FloatBarrier` kan nemt anvendes således at floats ikke kan flyde ind i andre afsnit, placér følgende i preamble:

Eksempel 6.12

```
\let\oldsection=\section % gemmer den gamle definition
\renewcommand\section{\FloatBarrier\oldsection}
```

Faktisk kan dette desuden opnås via en option til `placeins`-pakken. Makroen `\clear-doublepage` er normalt indbygget i `\chapter`-kommandoen.

Bruger man *memoir* kan [Eksempel 6.12](#) laves endnu nemmere

Eksempel 6.13

```
\setFloatBlockFor{section}
```

Et tilsvarende problem opstår idet at en float kan optræde øverst på den side hvorpå figuren er indsat. Dette kan være et problem hvis man har en situation a la:

Eksempel 6.14

```
bla bla

\section{Overskrift}

tekst
% float indsat her
```

Hvis denne afsnitsoverskrift kommer midt på siden og floaten kommer over den, så kunne det se ud som om figuren hører til i det forrige afsnit. Dette kan man undgå ved at anvende pakken `flafter` som sørger for at en float altid indsættes i dokumentet *efter* det sted den blev indsat i koden.

flafter

Brugen af `flafter` kan dog være et lidt drastisk skridt. Hvis det kun er nogle få steder det er et problem, så kan man anvende

```
\suppressfloats \suppressfloats[⟨loc*⟩]
```

hvor `⟨loc*⟩` er »t« eller »b«. Placeres denne på en side så betyder det at der på denne siden ikke må komme nogen floats i det angivne område. Helt uden argument betyder det at der slet ikke må komme nogen floats på den nuværende side.

6.2.4 Ikke-flydende floats

Det er en meget almindelig misforståelse at [h] skulle betyde »her og kun her«, vedrørende placeringen af figuren. Som vi kan se af afsnit [6.2.1 på side 142](#) er dette ikke tilfældet. Hvis man virkelig ønsker at figuren bare *skal* være her, er ligeglåd med huller i teksten og har brug for at kunne lave en caption, så kan man anvende pakken `float`, den giver endnu en mulighed til `⟨loc⟩`: [H], som *ikke* må blandes med de andre værdier som kan anvendes i `⟨loc⟩`. Ergo vil

float

Eksempel 6.15

```
\begin{figure}[H]
\centering
% figur kode
\caption{tekst}
\label{key}
\end{figure}
```

give en float som *ikke* flyder. Der er dog risiko for at endnu ikke fastlagte floats fra tidligere sider så flyder forbi og ødelægger nummereringen.

afterpage

Dette kan man dog lave lidt om på. Pakken `afterpage` giver makroen

```
\afterpage{⟨tekst/kode⟩}
```

Flere figurer på én gang

som indsætter $\langle\text{tekst/kode}\rangle$ som det første på næste side (heraf navnet). Ofte vil man jo bare have at figuren skal komme i nærheden af hvor man har indsat figuren. Så kunne man jo anvende

```
\afterpage{%
  \FloatBarrier % dump alle ikke placerede floats
  \begin{figure}[H]
  ...
  \end{figure}
}
```

Eksempel 6.16

$\backslash\text{afterpage}$ kan desuden anvendes til at dumpe alle endnu ikke placerede floats *efter* den i gangværende side. Dette har naturligvis den fordel, i sammenligning med $\backslash\text{FloatBarrier}$ eller $\backslash\text{clearpage}$, at man får fyldt den nuværende side med tekst inden siderne med floats kommer. Teksten fortætter så efter sidste floats. Dette opnås via:

```
\afterpage{\clearpage}
```

Eksempel 6.17

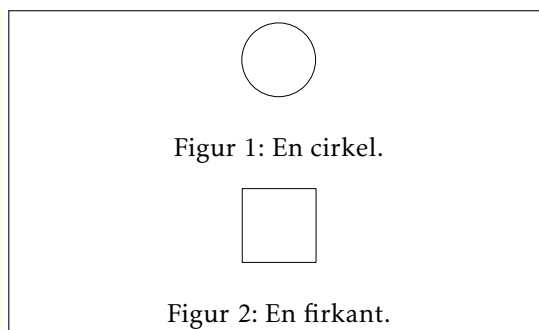
6.3 Flere figurer på én gang

Lad os kigge lidt på forskellige konstruktioner til at anvende mere end én figur ad gangen.

Flere figurer i samme float

For det første kan man uden problemer inkludere flere figurer i den samme float-konstruktion samt have flere $\backslash\text{caption}$'s.

```
\begin{figure}[htbp]
\centering
\includegraphics[width=1cm]{fig.2}
\caption{En cirkel.}
\vspace*{3mm}
\includegraphics[width=1cm]{fig.3}
\caption{En firkant.}
\end{figure}
```



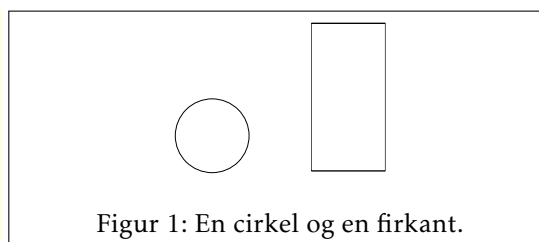
Eksempel 6.18

De to figurer i [Eksempel 6.18](#) flyder nu sammen i den samme float.

To figurer – en billedtekst

Flere figurer ved siden af hinanden kan laves nemt via [Eksempel 6.19](#).

```
\begin{figure}[htbp]
\centering
\includegraphics[width=1cm]{fig.2}
\quad
\includegraphics
  [width=1cm,height=2cm]{fig.3}
\caption{En cirkel og en firkant.}
\end{figure}
```



Eksempel 6.19

6. Om grafik, flydende objekter og billedtekst

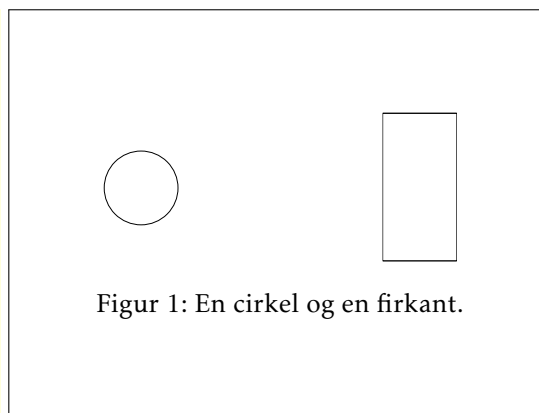
Her vil de to figurer begge stå på baseline. Skulle man ønske at de skulle aligne anderledes kan man forsøge med *minipage* eller `\parbox`, se [Eksempel 6.20](#).

`minipage`
`\parbox`

Eksempel 6.20



```
\begin{figure}[htbp]
\centering
\begin{minipage}[c]{0.47\textwidth}
\centering
\includegraphics[width=1cm]{fig.2}
\end{minipage}%
\hfill
\begin{minipage}[c]{0.47\textwidth}
\centering
\includegraphics
[width=1cm,height=2cm]{fig.3}
\end{minipage}
\caption{En cirkel og en firkant.}
\end{figure}
```



Argumentet `[c]` holder de to `minipage`s vertikalt centrerede i forhold til hinanden. Andre muligheder er `[t]` (justering på første basislinie) og `[b]` (justering på sidste basislinie).

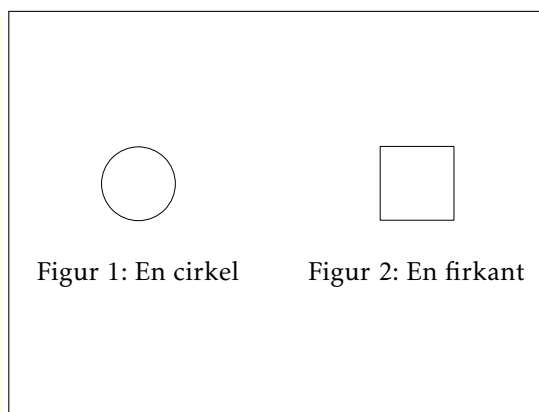
To figurer ved siden af hinanden med hver deres billedtekst

Ønsker man i stedet at have to figurer ved siden af hinanden, med hver sin billedtekst:

Eksempel 6.21



```
\begin{figure}[htbp]
\centering
\begin{minipage}[c]{0.47\textwidth}
\centering
\includegraphics[width=1cm]{fig.2}
\caption{En cirkel}
\end{minipage}%
\hfill
\begin{minipage}[c]{0.47\textwidth}
\centering
\includegraphics[width=1cm]{fig.3}
\caption{En firkant}
\end{minipage}
\end{figure}
```

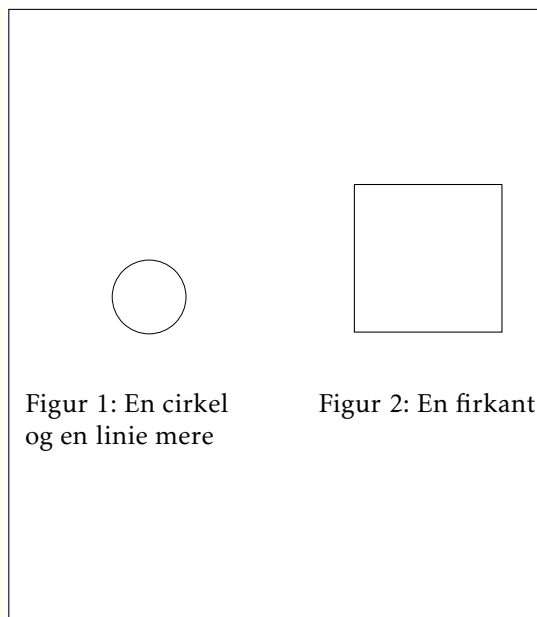


To figurer ved siden af hinanden – justering af billedtekst placering

Er de to captions eller figurer ikke lige høje kan man blive nødt til at blive lidt kreativ med f.eks. brugen af *minipage*. Et eksempel kan ses i [Eksempel 6.22 på næste side](#). Ideen er at man anvender en 2×2 struktur. Først to *minipage*'s til figurerne, her kan man selv bestemme hvordan den skal vertikalt justeres overfor hinanden. Dernæst på den følgende linie igen to *minipage*'s, denne gang justeret således at den første linie i begge kasser kommer til at stå på samme linie.

Tekst ved siden af figur – justering i toppen

```
\begin{figure}[htbp] \centering
\begin{minipage}[b]{0.47\textwidth}
\centering
\includegraphics[width=1cm]{fig.2}
\end{minipage}%
\hfill
\begin{minipage}[b]{0.47\textwidth}
\centering
\includegraphics[height=2cm]{fig.3}
\end{minipage} \\ %herefter captions
\begin{minipage}[t]{0.47\textwidth}
\caption{En cirkel\newline
og en linie mere}
\label{cap1}
\end{minipage}%
\hfill
\begin{minipage}[t]{0.47\textwidth}
\caption{En firkant}
\label{cap2}
\end{minipage}
\end{figure}
```



Eksempel
6.22



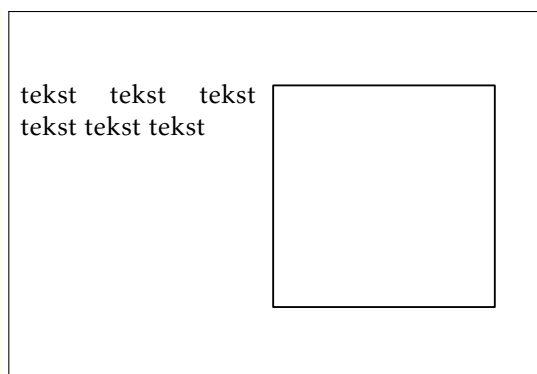
Bemærk hvordan de to figurer står på samme basislinie og at første linie i captions automatisk er justeret til samme baseline. Man skal især notere sig brugen af [b] og [t].

Har man en smal figur og en stor caption kunne man jo placere captionen ved siden af figuren via en konstruktion som i [Eksempel 6.21 på modstående side](#).

Tekst ved siden af figur – justering i toppen

Til tider vil man gerne have en figur på den ene side og en længere tekst til den på den anden. Det er nemt nok med to *minipage*-er hvis figuren og teksten bare skal være centreret vertikalt overfor hinanden. Hvis man ønsker at første linie af teksten skal passe med toppen af figuren, så skal man være mere snedig.

```
\usepackage{calc}% auto udregn
\usepackage{graphicx}
----- slut preamble -----
\begin{minipage}[t]{0.45\linewidth}
tekst tekst tekst tekst tekst tekst
\end{minipage}
\hfill
\begin{minipage}[t]{0.45\linewidth}
\raisebox%
{\heightof{h}-\totalheight}%
{\includegraphics[width=3cm]{fig.3}}
\end{minipage}
```



Eksempel
6.23



Forklaring: [t]-et betyder at første basislinie i de to kasser skal stå i samme højde på siden. Men billedet står jo på den første basislinie i den højre kasse og rager en hel del længere op. Så man løfter først kassen op så overkanten passe med højden af et »h« og sænker derfra figuren i hele dens højde.

6.4 Konfiguration af billedteksten (`\caption`)

Mantra. Vi minder lige om: *En `\label` skal **altid** komme efter `\caption`, ellers vil referencen blive forkert.* Nu har vi sagt det og det bliver *ikke* gentaget.

`caption` Visse dokumentklasser (e.g. *memoir* eller KOMA-systemet) har deres egne konfigurationsmuligheder vedrørende captions, se de respektive brugermanualer. Vi vil i stedet kigge lidt på pakken `caption` af Axel Sommerfeldt. Pakken er kompatibel med alle standardklasserne samt klasserne nævnt ovenfor og den tilbyder endnu flere konfigurationsmuligheder.

Sidebemærkning 6.17. Man skal dog huske at anvender man `caption`-pakken kan man ikke længere anvende de indbyggede konfigurationsmuligheder i de omtalte klasser, deres `\caption`-implementation bliver overskrevet.

Pakken er veldokumenteret og ganske velskrevet – eksempel baseret, se [Sommerfeldt \(2005\)](#).

6.4.1 Brug af pakken `caption`

Pakken loades som normalt og kan tage forskellige options. Disse options kan også angives via

```
\captionsetup[<float type>]{<options>}
```

som desuden kan anvendes lokalt indeni eksempelvis *figure*-environmentet, før `\caption` til at lave en lokal ændring. *<float type>* argumentet anvendes til at lave globale ændringer som kun gælder for f.eks. `figure`, se afsnittet [Mellemrum før og efter billedtekst](#) på side 152.

Da brugermanualen til `caption` er så godt skrevet vil vi bare løse nogle småproblemer og i stedet henvise til manualen for yderligere oplysninger. Makroen `\dftxt` indeholder bare noget tekst, så sparer vi en del plads.

Ændring af billedtekstens font og fontstørrelse

Eksempel 6.24



```
\usepackage{caption}
\captionsetup{font=small,labelfont=bf}
----- slut preamble -----
\begin{figure}[htbp]
\centering
\fbbox{Figur af tekst}
\caption{\dftxt}
\end{figure}
```

Figur af tekst

Figur 1: Dette er en lang og latterlig tekst, så kan man se at der sker noget.

Man kan også lave en hængende indentering samt styre fonten af label tekst og `caption` tekst for sig.

Ændring af separatortegn

```
\usepackage{caption}
\captionsetup{font=small,
labelfont={it,bf},textfont=sf,
format=hang}
```

— slut preamble —

```
\begin{figure}[htbp]
\centering
\fbbox{Figur af tekst}
\caption{\dftxt}
\end{figure}
```

Figur af tekst

Figur 1: Dette er en lang og latterlig tekst, så kan man se at der sker noget.

Eksempel

6.25



Ændring af separatortegn

Antag man har en figur som skal nummereres men der er ikke nogen captiontekst. Her vil man normalt få »Figur 35:«, hvilket ikke ser pænt ud. Vi vil gerne have noget andet i stedet.

\caption er nu med caption-pakken blevet så snedig at den tester om der er en caption tekst og hvis der ikke er nogen skriver den ikke nogen separator.

```
\usepackage{caption}
```

— slut preamble —

```
\begin{figure}[htbp]
\centering
\fbbox{Figur af tekst}
\caption{}
\end{figure}
```

Figur af tekst

Figur 1

Eksempel

6.26



Bredden af caption

Hvis billedteksten er lang, vil den fylde hele den nuværende liniebredde, hvilket kan se underligt ud hvis figuren er meget smal. Man kan så i stedet, lokalt, specificere bredden af denne caption. Har desuden mulighed for at angive en margin, hvor billedteksten så får en indrykning i begge sider svarende til den angivne størrelse.

```
\usepackage{caption}
```

— slut preamble —

```
\begin{figure}[htbp]
\centering
\fbbox{Smal}
\captionsetup{width=4cm}
\caption{\dftxt}
\end{figure}
```

Smal

Figur 1: Dette er en lang og latterlig tekst, så kan man se at der sker noget.

Eksempel

6.27



6. Om grafik, flydende objekter og billedtekst

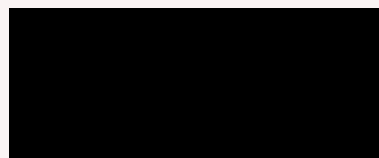
threparttable
measuredfigure

Eksempel 6.28



Tip 6.18. Ved visse figurer er det rart at caption holder bredden af figuren, oftest er dette pænt med figurer (af en vis størrelse) placeret ved siden af hinanden. Her har pakken threparttable (som ellers er omtalt på side 188) et trick i ærmet. Anvend *measuredfigure* omkring figuren og `\caption`:

```
\usepackage{threparttable}
----- slut preamble -----
\begin{figure}[htbp]
  \centering
  \begin{measuredfigure}
    \rule{5cm}{2cm}
    \caption{Bla bla bla bla bla
             bla bla bla bla bla}
  \end{measuredfigure}
\end{figure}
```



Figur 1: Bla bla bla bla bla bla
bla bla bla bla bla

Mellemrum før og efter billedtekst

Der er to længder som styrer spacingen over og under captions: `\abovecaptionskip` og `\belowcaptionskip`. Den første er normalt 10 pt, mens den anden er 0 pt. Dette giver et problem når man placerer captionen over materialet, se [Eksempel 6.29](#).

Eksempel 6.29

```
\begin{figure}[htbp]
  \centering
  \caption{Tekst over}
  \fbox{Figur tekst}
  \caption{Tekst under}
\end{figure}
```

Figur 1: Tekst over

Figur tekst

Figur 2: Tekst under

Nu er det ikke almindeligt at man skriver captionen over figurer, men det ses ofte ved tabeller. Igen hedder løsningen pakken `caption`, enten via

```
\captionsetup [table]{position=top}
```

hvorved værdierne for de to beskrevne længder bytter plads ved alle tabel-captions. Linien skal placeres i preamble. Faktisk er dette en så almindelig ting at man også kan specificere det som et argument til selve pakken.

```
\usepackage[... ,tableposition=top]{caption}
```

Forskellen kan nu ses via [Eksempel 6.30](#).

Eksempel 6.30



```
\usepackage[tableposition=top]{caption}
----- slut preamble -----
\begin{figure}[htbp]\centering\fbox{Figur tekst}\caption{Tekst under.}
\end{figure}
\noindent\hrule
\begin{table}[htbp]\centering\caption{Tekst over.}\fbox{Tabel tekst}\end{table}
```

Displaydelen af eksempel 6.30 findes på næste side

En figur og en tabel i samme float

<div style="border: 1px solid black; display: inline-block; padding: 2px 10px;">Figur tekst</div> <p>Figur 1: Tekst under.</p>
<hr/> <p>Tabel 1: Tekst over.</p> <div style="border: 1px solid black; display: inline-block; padding: 2px 10px;">Tabel tekst</div>

**Eksempel
6.30
fortsat**

En figur og en tabel i samme float

Vi har tidligere set hvordan man kan sætte to figurer ved siden af hinanden i samme float, men hvad med en figur og en tabel? Der er to problemer i dette (i) for det første kan man *ikke* placere en float indeni en anden float (ii) for det andet kan man heller ikke placere en float inden i en *minipage* (eller lignende). Vi må altså gøre noget andet – snyde! Vi ønsker at anvende *figure* og vil gerne have en figur samt en tabel som skal flyde sammen. Anvender følgende makro til at emulere en tabelcaption:

```
\captionof{<float type>}{<tekst>}
```

\captionof

Et eksempel kan ses herunder:

```
\usepackage{caption}
----- slut preamble -----
\begin{figure}[htbp]\centering
\begin{minipage}[b]{0.45\textwidth}
\centering\fbbox{Figur tekst}\caption{Caption til figur.}
\end{minipage}
\hfill
\begin{minipage}[b]{0.45\textwidth}
\centering\fbbox{Tabel tekst}\captionof{table}{Caption til tabel.}
\end{minipage}
\end{figure}
```

**Eksempel
6.31**


<div style="border: 1px solid black; display: inline-block; padding: 2px 10px;">Figur tekst</div> <p>Figur 1: Caption til figur.</p>	<div style="border: 1px solid black; display: inline-block; padding: 2px 10px;">Tabel tekst</div> <p>Tabel 1: Caption til tabel.</p>
--	--

\captionof kan anvendes overalt til at lave captions med, og er altså ikke forbeholdt floats. Men den har altså stadigvæk problemerne med at andre captions (af samme type) kan flyde forbi.

Flere floats med samme nummer

Hvis man har mange små figurer som hører sammen, men de samlet set ikke kan være på en side, kan man splitte dem op i to. [Eksempel 6.34 på side 155](#) viser hvordan man sørger for at give hver lille figur et nummer og hvordan man kan bibeholde figurnummeret ved begge floats. Man kan så evt. kombinere det med \afterpage og \FloatBarrier så de to float sider kommer på to efterfølgende sider.

6. Om grafik, flydende objekter og billedtekst

6.4.2 Undernummerering af figurer – subfig

subfig Pakken *subfig* er afløseren til pakken *subfigure* af samme forfatter. Den er fuldt kompatibel med *caption* (anvender den endda internt). Pakken er igen meget veldokumenteret, specielt kan jeg godt lide at den viser effekten af de forskellige konfigurationsmuligheder via en serie af figur-eksempler (ca. 80–90 forskellige), så vi vil bare vise et par eksempler. Læseren må så selv konsultere manualen, [Cochran \(2004\)](#).¹⁰

De vigtigste kommandoer er:

```
\subfloat      \subfloat[⟨subcaption tekst incl. label⟩]{⟨figur eller tabel⟩}
\subref       \subref{⟨key⟩}
\subref*      \subref*{⟨key⟩}
```

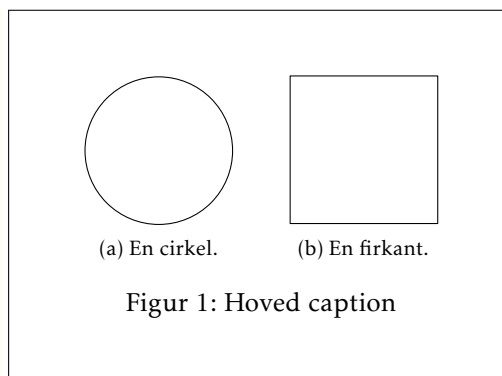
Desuden er der nu en *⟨float type⟩* benævnt ved *subfigure* som man kan konfigurere via *\captionsetup*.

Eksempel

6.32



```
\usepackage{subfig}
\usepackage{graphicx}
\newcommand\inclfig[1]{%
  \includegraphics[width=20mm]{#1}}
----- slut preamble -----
\begin{figure}[htbp]
  \centering
  \subfloat[En cirkel.]{\inclfig{fig.2}}
  \quad
  \subfloat[En firkant.]{\inclfig{fig.3}}
  \caption{Hoved caption}
\end{figure}
```



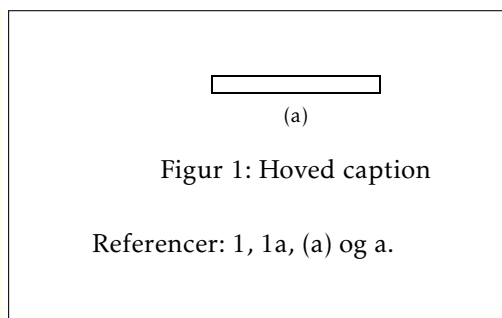
Bredden af subcaptionen justerer sig efter bredden af figuren. Bemærk at man her ikke har brug for *minipage*. *\subref* anvendes i forbindelse med referencer således at man kan få adgang til undernummereringen. Se [Eksempel 6.33](#).

Eksempel

6.33



```
\usepackage{subfig}
\newcommand\dby{\fbox{\hspace{2cm}}}
----- slut preamble -----
\begin{figure}[htbp]
  \centering
  \subfloat[\label{sub}]{\dby}
  \caption{Hoved caption}\label{main}
\end{figure}
Referencer: \ref{main}, \ref{sub},
\subref{sub} og \subref*{sub}.
```



Makroerne *\subref* og *\subref** kan altså få fat på subnummeret på figuren. Bemærk placeringen af *\label{sub}*, den *skal* angives på denne måde ellers fungerer det ikke.

Flere floats under samme nummer

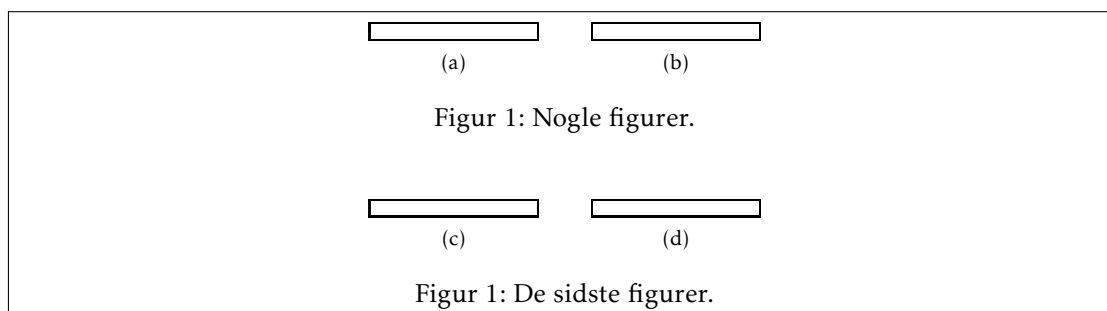
Har man mange figurer som skal samles under et, men fylder mere end en side, kan man dele dem ud over to floats:

¹⁰ Igen har *memoir*-klassen en lignende funktionalitet indbygget, men *subfig*-pakken er langt overlegen i dens konfigurationsmuligheder.

Andre grafikrelaterede ting

```
\usepackage{subfig}
----- slut preamble -----
\begin{figure}[htbp]
  \centering
  \subfloat[{}]{\dby}\qqquad\subfloat[{}]{\dby} \caption{Nogle figurer.}
\end{figure}
\begin{figure}[htbp]
  \ContinuedFloat
  \centering
  \subfloat[{}]{\dby}\qqquad\subfloat[{}]{\dby} \caption{De sidste figurer.}
\end{figure}
```

Eksempel
6.34



Se selv [Cochran \(2004\)](#) for flere eksempler på hvad man kan.

6.5 Andre grafikrelaterede ting

Dette afsnit vil ikke indeholde så mange detaljer, da de fleste ting ligger uden for rammerne af denne bog, men vi ville bare lige nævne dem her.

At skrive tekst rundt om figurer

Har man nogle afsnit med tekst (rettere, uden fremhævet matematik og lister) så kan man godt få \LaTeX til at wrappe rundt om en figur. Tag et kig på pakkerne `wrapfig` og `floatflt`, læs deres manualer på CTAN. Det skal bemærkes at konstruktionerne er en smule sårbare og det kan være at man skal bruge et par bizzarre tricks for at få konstruktionerne til at fungere, så anvend dette varsomt.

`wrapfig`
`floatflt`

Rette på labels i EPS figurer

Det er ofte et problem når man laver grafer i et eksternt program at de labels som kommer på tegningen ikke kommer i samme font som den man anvender i sit dokument.

Dette kan fikses (hvis man anvender `latex` og `dvips`) med pakken `psfrag`. Ideen er at man i stedet for at anvende labels i sin graf derimod skriver nøgleord, f.eks. `p1`, `l2` etc. Man får så bagefter \LaTeX til at erstatte disse med en anden tekst. Manualen [Grant et al. \(1998\)](#) indeholder forskellige eksempler samt noget om begrænsninger og problemer ved metoden.

`psfrag`

Bemærk at det er ikke muligt at anvende `psfrag` på enhver EPS-fil.

6. Om grafik, flydende objekter og billedtekst

Sidebemærkning 6.19 (PStricks i PDF).

Man kan ikke lave noget tilsvarende i PDF.

Lad os lige antage at man har nogle EPS-filer, som der skal laves psfrag på og man bagefter ønsker at anvende resultatet under pdfL^AT_EX.

Nemt nok. Lav et nyt dokument, inkluder dine EPS-figurer, lav dine psfrag-ting, sørg for et billede per side og intet sidehoved eller -fod. Oversæt til PostScript og herefter til PDF. Herefter beskæres hver side via

```
pdfcrop fil.pdf
```

Som genererer den nye fil `fil-crop.pdf`, hvor hver side er et separat beskåret billede. Disse inkluderes så i det originale dokument. Bemærk at i PDF-mode understøtter `\includegraphics` endnu en option: `page=<num>`, således at man kan få fat i en specifik side i en PDF-fil.

Baggrundsbilleder

eso-pic

Man kan lægge figurer og andet ind i baggrunden af siden via konstruktioner stillet til rådighed af pakken `eso-pic`. Læseren bedes venligst læse [Niepraschk \(2002\)](#) for information om brugen af pakken. Det er også en *meget* god ide at kigge på eksemplerne som følger med pakken, se CTAN.

Men i korte træk vil den kunne placere en figur eller vilkårlig tekst et hvilket som helst sted enten på alle sider eller kun på den nuværende. Et eksempel kunne være

Eksempel 6.35



```
\usepackage{eso-pic}
\usepackage{graphicx}

----- slut preamble -----

\AddToShipoutPicture*{%
  \put(0,0){%
    \includegraphics[width=\paperwidth,height=\paperheight]{figur}
  }
}
```

som indsætter en figur som fylder hele siden på netop denne side. Hvis der ikke skal stå mere end denne figur på denne side er det en god ide at huske at indsætte en `»~«` (tilde) eller en `»\quad«` på siden, ellers står der slet ikke noget (i T_EX-forstand) og L^AT_EX vil ikke skrive siden.

Syntaksen som anvendes er den samme som ved *picture*, dvs.

```
\put(<x,y>){<kode>}
```

hvor `(<x,y>)` er i enheder af `\unitlength`.¹¹ Desuden har man i dette tilfælde at `(0,0)` er nederste venstre hjørne af papiret.

Pakken `pdfpages` (beskrevet i afsnit 6.1.3 på side 141) anvender internt `eso-pic`.

Figurer i landscape

Se afsnit 7.8.2 på side 191 for information om hvad man kan gøre hvis man ønsker en figur eller tabel i landscape.

¹¹ Med den rette pakke må man gerne skrive med almindelige enheder i disse koordinater. Se iøvrigt *memoir*-manualen for en nærmere forklaring på *picture*-environmentet. Det er ikke så bekvemt at tegne i (TikZ er bedre), men kan være nyttig at kende til meget specialiserede ting. Vi anvender det f.eks. i et brevpaapirdesign, for at placere kolofonen. I *memoir*-klassen er trimmarks tegnet med dette environment.

6.6 Tips og tricks

6.6.1 At skrive ovenpå inkluderede billeder

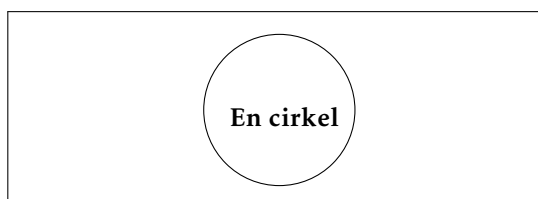
I mange tilfælde kan det være en ide at skrive ovenpå et inkluderet billede, hvad enten det er nogle copyright informationer eller hvad det nu er.

En standard pakke til dette er pakken `overpic`. Den bruger en lidt speciel syntaks: `overpic`

```
\begin{overpic}[\langle options \rangle]{\langle billede \rangle}
\put(\langle koordinater \rangle){\langle tekst \rangle}
...
\end{overpic}
```

Hvor `\langle options \rangle` er options til den indbyggede `\includegraphics` og hvor `\langle koordinater \rangle` er på formen »35,75«, altså to koordinater uden enhed. Koordinatsystemet starter i nederste venstre hjørne af billedet.¹²

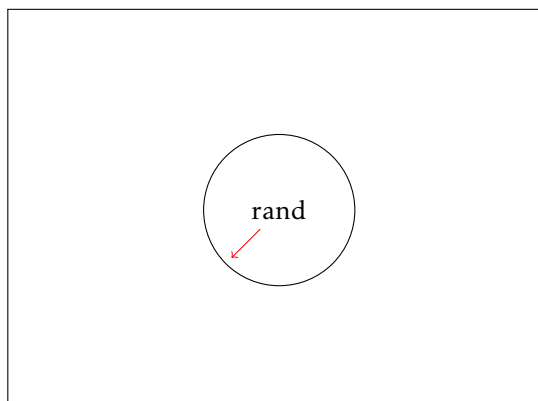
```
% overpic loader automatisk graphicx
\usepackage{overpic}
----- slut preamble -----
\begin{overpic}{fig.2}
\put(18,40){\textbf{En cirkel}}
\end{overpic}
```



Eksempel 6.36


Det er også nærliggende at forsøge sig med TikZ (se afsnit 6.6.6 på side 161) til at skrive ovenpå et billede. Følgende eksempel skyldes Martin Heller.

```
\usepackage{tikz,graphicx}
----- slut preamble -----
\begin{tikzpicture}
% indsæt billedet i en node
% nederste venstre i (0,0)
\node[inner sep=0pt,anchor=south west] (img)
at (0,0) {\includegraphics{fig.2}};
\node (rand) at (img.center) {rand};
% tegn efter absolutte koordinater
\draw[->,red] (rand) -- (0.4,0.4);
\end{tikzpicture}
```



Eksempel 6.37


6.6.2 Genstridige EPS-filer

Det er ikke alle programmer som er lige gode til at lave EPS-filer. Så nogle gange bliver man nødt til at hjælpe lidt til bagefter.

En ting man kan prøve er at køre filen gennem programmet `eps2eps` som har følgende syntaks `eps2eps`

```
eps2eps indfil udfil
```

Programmet forsøger at rydde op i den problematiske EPS-fil. Det fungerer fint, især hvis man har problemer med at få printet en fil. Dog har det den ulempe at programmet

¹² Pakken `picture` udvider `\put` syntaksen, således at koordinaterne kan angives med enhed.

6. Om grafik, flydende objekter og billedtekst

bevarer ikke fonte, tegnene bliver til bitmaps, i de fleste tilfælde kan man ikke se forskel, men alligevel. . .

Jeg er fornyligt blevet bekendt med et lille værktøj som jeg godt ville gøre lidt reklame for: `epstool`. Det findes både til Windows og Linux, og har hjemmeside her: <http://pages.cs.wisc.edu/~ghost/gsview/epstool.htm>, dog kan det anbefales at hente det fra CTAN: <http://www.ctan.org/tex-archive/support/ghostscript/ghostgum/>, da den er nemmere at komme i kontakt med. Programmet `epstool` kan anvendes til at rydde op i EPS-filer, f.eks. fikse den såkaldte *BoundingBox*, det kan fjerne *previews* fra EPS-filer lavet i Adobe Illustrator samt andre sjove ting. De to nævnte ting er ret centrale for os.

Sidebemærkning 6.20. *BoundingBox*'en er en tekststreng indeni EPS-filen som beskriver størrelsen af figuren. Faktisk er dette det eneste \TeX læser fra en EPS-fil når den skal give plads til den. Linien kan se ud som følger:

```
%%BoundingBox: 285 405 323 520
```

hvilket skal fortolkes som (x, y) koordinaterne til hhv. nederste venstre og øverste højre hjørne af boksen. *BoundingBox*'en bør som udgangspunkt være det mindste rektangel som omkranser indholdet af billedet (evt. med en lille smule luft omkring).

Den fejl man oftest ser ved EPS-filer er at *BoundingBox*'en er forkeret. Det kan testes ved at sætte en `\fbox{...}` (i.e. en boks) omkring billedet. Den mest almindelige fejl er at der er alt for meget luft omkring billedet. *BoundingBox*'en bør lige nøjagtigt omkranse selve indholdet af billedet. Mange kendte programmer til Windows som selv siger at de kan gemme som EPS, laver ofte en forkeret *BoundingBox*. Et andet eksempel er programpakken R, den er notorisk kendt for at lave *BoundingBox*'e som er alt for store.¹³

Nuvel, dette kan `epstool` fikse via

```
epstool --copy --bbox indfil.eps udfil.eps
```

Et andet problem man nogle gange ser er EPS-filer med et såkaldt binært preview.¹⁴ Det er en binær fil/billede som ligger indlejret indeni selve EPS-filen. Dette kan få `epstopdf` til at nægte at oversætte filen til PDF. Igen `epstool` to the rescue:

```
epstool -p indfil.eps udfil.eps
```

Desværre kan de to ikke kombineres. Se manualen til `epstool` for yderligere features.

6.6.3 Genstridige PDF-billeder

En ting som vi til tider ser, er fontproblemer ved pdf-billeder genereret for forskellige programmer. Som regel skyledes disse problemer at de fonte som er anvendt i billedet ikke er blevet indlejret i selve PDF-filen.

Generelt bør man *altid* indlejre¹⁵ *alle* anvendte fonte. Så bliver dokumentet mere portabelt og vil virke på de platforme som ikke lige har de anvendte fonte installeret.

¹³ Ved R ser det ud som om at *BoundingBox*'en har en minimumsstørrelse som er ligeglad hvor lille selve billedet faktisk er.

¹⁴ Ofte set når man har lavet tegninger i Adobe Illustrator

¹⁵ Også kaldet *embedde*.

Generering af EPS-filer eller croppede PDF-filer via L^AT_EX

Problemet kan til tider ses hvis fonten helt mangler i PDF-fremviseren eller vises anderledes end man forventede.

Tip 6.21. I de fleste deciderede PDF-fremvisere (dvs. f.eks. *ikke* fremviseren i TeXWorks), kan man se informationer vedr. de fonte som er anvendt i PDF-filen. I Adobe Reader findes de under Filer og så under Egenskaber. Vær især opmærksom på kommentaren »Faktisk skrifttype«, den fortæller at AR anvender en anden font den den som der faktisk står i PDF-filen.

Tip 6.22 (Excel).

Når man i nyere Excel versioner gemmer i PDF-format ser det ikke ud til at gen indlejrer *Times New Roman*. Her kan man evt. komme uden om problemet ved at vælge at billedet skal gemmes i PDF/A format.

6.6.4 Generering af EPS-filer eller croppede PDF-filer via L^AT_EX

Har man lavet noget i L^AT_EX som man gerne vil have lavet om til en EPS-fil som så kan inkluderes i et andet dokument så skal man bare forberede en side hvor der kun står det man ønsker på sin EPS-fil (husk at fjerne sidetallet, f.eks. via `\thispagestyle{empty}`). Her efter skal man anvende et par ekstra options til dvips:

```
dvips -E -o fil.eps fil.dvi
```

Det er ikke altid BoundingBox'en bliver korrekt med dvips, når man skal generere EPS-filer, specielt hvis man anvender rotationseffekter eller andre specielle PostScript ting. Men så har vi jo epstool.

Skulle man på den anden side arbejde rent i PDF, så kan man gøre det samme som ovenfor og oversætte dokumentet med pdf_latex og så croppe PDF-filen via pdfcrop. *pdfcrop*

```
pdfcrop fil.pdf
```

Dette giver en version af `fil.pdf` (`fil-crop.pdf`), hvor de overskydende marginer er fjernet (dette er mere eller mindre hvad »at *croppes*« betyder).

6.6.5 Billed/figur-konvertering

EPS til PDF

Anvend epstopdf – den følger med L^AT_EX distributionen. Visse versioner af programmet kræver at man har GHOSTSCRIPT installeret. Syntaksen er *epstopdf*

```
epstopdf filnavnind.eps
```

Man bør *ikke* anvende ps2pdf til konvertering af EPS-filer, idet dette program normalt vil lave outputfilen om til f.eks. en hel A4-side, dvs. den ødelægger BoundingBox'en, epstopdf bevarer i stedet denne.

6. Om grafik, flydende objekter og billedtekst

epstopdf
ps2pdf

Sidebemærkning 6.23. epstopdf og ps2pdf anvender begge programmet ghostscript internt. Det skal nævnes at dette program kan finde på automatisk at rotere visse sider. Da epstopdf ikke kan tage nogen argumenter skal man på Linux udføre (hvis man anvender bash)

```
export GS_OPTIONS=' -dAutoRotatePages=/None '
```

eller under tcsh (IMF)

```
setenv GS_OPTIONS -dAutoRotatePages=/None
```

før man kører epstopdf. Ved ps2pdf kan »-dAutoRotatePages=/None« bare specificeres som argument.

Under Windows er alt naturligvis anderledes. Ved MiKTeX skulle syntaksen blive

```
epstopdf --gsopt=-dAutoRotatePages=/None fil.eps
```

(dette er dog utestet)¹⁶ **Fixme Note: skal testes**

JPG til EPS

jpeg2ps

Man kan anvende forskellige programmer som kan konvertere figuren til en EPS-fil, men det anbefalede program er jpeg2ps. Programmet kan hentes via <http://www.pdflib.com/download/free-software/jpeg2ps/> og findes både til Windows og Linux. Syntaks:

```
jpeg2ps filnavn.jpg > filnavn.eps
```

som så skaber filnavn.eps. Fordelen ved dette program er at den resulterende EPS-fil kun bliver en lille smule større end originalen. I modsætning til f.eks. konvertering via programmet convert (fra ImageMagick), som faktisk dekomprimerer billedfilen, hvilket ofte forøger filstørrelsen med en faktor 100.

PNG til EPS

Her er det nemmeste nok først at anvende et hvilket som helst billedbehandlingsprogram til at konvertere filen til JPG og derefter anvende jpeg2ps.

MPS til EPS

Outputtet fra METAPOST (som ofte bare er på formen $\langle filnavn \rangle . \langle n \rangle$, $n \in \mathbb{N}$) er en delmængde af EPS...

PDF til EPS

Dette er en smule tricky.¹⁷ Det bedste resultat jeg fik var ved brug af pdftops som følger med xpdf (som desværre kun findes til Linux). Dog giver pdf2ps (del af GHOSTSCRIPT) også brugbare resultater. Syntaksen med pdftops er

```
pdftops -eps fil.pdf
```

¹⁷ I hvert fald ud fra den ikke videnskabelige test jeg afprøvede.

SVG til EPS

Grafik formatet `svg` (Scalable Vector Graphics) kommer i fremtiden til at spille en større og større rolle. Pt. kan $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ikke selv håndtere `svg`, men der findes konverterings værktøjer som gør det muligt at konvertere til `eps`. Tak til Mikkel Kamstrup Erlandsen for at have gjort opmærksom på dette. Programmet Inkscape (<http://www.inkscape.org>) kan f.eks. læse `svg` og gemme det som `eps`.

Sidebemærkning 6.24 (sam2p).

Et interessant, men lidet kendt program er `sam2p`. Hvilket er et program som nærmest kan konvertere et vilkårligt billedformat til PostScript eller PDF. Desuden skulle det være i stand til at levere meget mindre filer end f.eks. `Imagemagick` (`convert`).

Programmet kan downloades via <http://www.inf.bme.hu/~pts/sam2p/>, hvor man både finder en Unix/Linux version samt binære til Windows. I Ubuntu skal man som sædvanlig bare bede om den.

Jeg har ikke personligt den store erfaring med programmet, men de som har testet det har været meget begejstrede.

Sidebemærkning 6.25 (pdfcrop).

Jeg vil også lige selv nævne programmet `pdfcrop` som følger med $\text{T}_{\text{E}}\text{X}$ Live. Det kan anvendes til at beskære eller *croppede* PDF-filer, til PDF ækvivalenten af en `eps`-fil.

Jeg anvender det i [Madsen \(2008b\)](#) sammen med `pdftops`, da det i forhold til `dvips` giver bedre croppede figurer, især hvis der er specielle rotationseffekter.

Sidebemærkning 6.26 (Stregtegninger).

En enkelt formaning: Anvend *ikke* `JPG` eller `PNG` som originalformat til *stregtegninger*, her bør vektorgrafisk `eps` (eller `PDF` anvendes). Dette sikrer at figuren kan skaleres frit og uden opløsningsproblemer.

Pas også på med skalering af `JPG` eller `PNG`, thi idet disse er bitmatformater (bygget op af små firkanter) er den eneste måde disse kan gøres større på, at gøre firkanterne større. Resultatet bliver så at figuren kommer til at se grynet ud.

Nedskalering er der dog ikke noget problem med.

6.6.6 Generering af grafik til brug i $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

Det er tidligere blevet nævnt at dette er ikke noget vi kommer så meget ind på. Vi vil dog give en introduktion til `TikZ/PGF`, da dette kan være interessant for en del brugere.

Vi vil starte dette afsnit med kort at præsentere nogle værktøjer som enten anvendes inden i $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -selv, eller som gør brug af $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ internt (således at ens figurer kan få labels i samme font som man anvender i sit dokument).

METAPOST

Helt tilbage i starten af bogen nævnte vi, at Donald Knuth, ud over $\text{T}_{\text{E}}\text{X}$, også skabte et fontformat kaldet `METAFONT`, outputtet fra denne var bitmaps. Inspireret af dette blev `METAPOST` skabt af John D. Hobby. `METAPOST` genbruger en del ideer og syntaks fra `METAFONT`, men giver vektorgrafik som resultat.

6. Om grafik, flydende objekter og billedtekst

Der er her tale om at man helt eller delvist *koder* sit billede. Koden oversættes så med programmet `mpost` til en simplificeret form for PostScript (faktisk en simpel form for EPS billeder).

Når METAPOST bliver bedt om at skrive en label et sted, så anvender den \TeX (eller \LaTeX) til at formatere det med.

Udviklingen af METAPOST lagde stille i mange år, men er nu genoplevet, så der kommer flere nye features til. Se [Hobby et al. \(2011\)](#) for mere information. Blandt de mere sjove anvendelse af METAPOST, så har en gruppe polske \TeX og font hackere skabet et system til at designe fonte ud fra METAPOST definitioner.

METAPOST er specielt god til tegninger i planen.



Jeg har selv anvendt METAPOST i mange år, især den meget analytiske tilgang til opbygningen af en tegning finder jeg meget appellerende.

Desværre er METAPOST eksempler ikke noget som det flyder så meget med på nettet, men har man spørgsmål vedr. METAPOST, er man velkomment til at sende mig en mail.

Du har allerede METAPOST, den er en del af både Mik \TeX og \TeX Live.

Asymptote

Dette minder lidt om METAPOST, bare med en mere C++-agtig syntaks. En af dets forcer skulle være 3D support. Se <http://asymptote.sourceforge.net/> for mere information.

Gnuplot

Gnuplot er måske et lidt nørdet program til at lave grafik med (det findes til alle platforme og er gratis). Det er især god til at lave grafer over eksperimentelle data.

Se <http://www.gnuplot.info> for mere information. Gnuplot kan eksportere til mange forskellige formater bl.a. \LaTeX .

PSTRICKS

Selv om PostScript måske ikke anvendes af flertallet mere, så skal man ikke undervurdere dette format. PostScript er faktisk er temmeligt kraftigt programmeringssprog og PSTRICKS er en måde hvorpå man fra et \LaTeX -dokument kan gøre brug af disse egenskaber til at skabe fantasifulde billeder. Dette sker så gennem en hel del pakker, de fleste skrevet af Herbert Voß.

Da der er tale om PostScript så *skal* man gennem `dvips` for at tingene bliver som de skal. Er man `pdf \LaTeX` -bruger kan man derfor ikke anvende PSTRICKS direkte. Der findes en pakke (`pdftricks`) som under `pdf \LaTeX` vil lave billedet om til en ekstern-fil og så oversætte med de rette programmer således at man ender op med et stykke eksternt PDF-grafik.

Mere information kan ses på <http://tug.org/PSTricks/main.cgi/>.

Løber man ind i problemer, så læg et spørgsmål på nyhedsgruppen `comp.text.tex` (husk at nævne PSTRICKS i emnet), så kan man være næsten sikker på at få et svar fra Herbert Voß, som nok er den som ved mest om PSTRICKS.

TikZ/PGF

PGF/TikZ

Syntaksen minder en del om syntaksen anvendt ved METAPOST. Kan køres

TikZ/PGF

Da PGF/tikz kan anvendes *både* med latex (skal gennem dvips) og med pdf-latex så er dette system meget anbefalelsesværdigt. Se [Tantau \(2007\)](#) for yderligere information. Via dette link <http://www.fauskes.net/pgftikzexamples/all/> kan man finde en masse TikZ/PGF eksempler.¹⁸ Andre eksempler kan findes via <http://www.texample.net/tikz/>.

Faktisk findes der redskaber til at konvertere en given EPS-fil til PGF-formatet, se <http://sourceforge.net/projects/eps2pgf/>.

Programmet gnuplot skulle også kunne eksportere til PGF.

gnuplot

Manualen til TikZ/PGF er meget omfattende så der er ikke plads til at beskrive systemet her, men vi vil give et lille eksempel på hvordan man kan tegne en form for vægтет graf, ofte anvendt i operations-analyse.

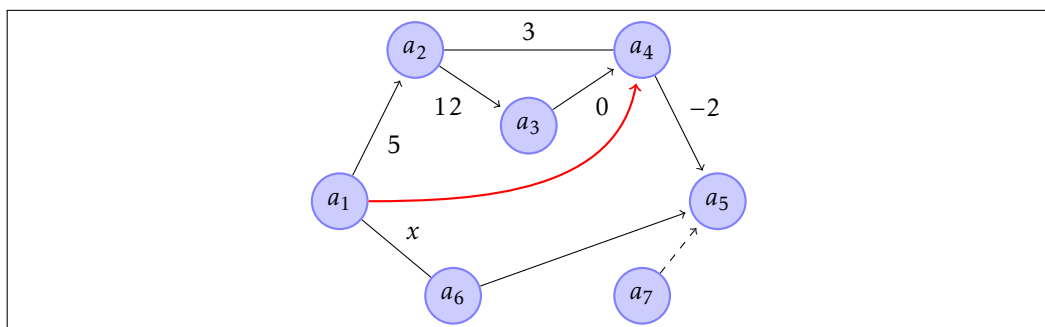
Hvis nogen skulle have lyst til at skrive en lidt længere TikZ intro til denne bog, så de meget gerne kontakte undertegende.

```
\usepackage{tikz}
----- slut preamble -----
\tikzstyle{knude}=[circle,draw=blue!50,fill=blue!20,thick]
\begin{tikzpicture}[shorten >=2pt]
  \node (a_1) at (0,0) [knude] {$a_1$};
  \node (a_2) at (1,2) [knude] {$a_2$};
  \node (a_3) at (2.5,1) [knude] {$a_3$};
  \node (a_4) at (4,2) [knude] {$a_4$};
  \node (a_5) at (5,0) [knude] {$a_5$};
  \node (a_6) at (1.5,-1.25) [knude] {$a_6$};
  \node (a_7) at (4,-1.25) [knude] {$a_7$};

  \draw[->] (a_1) -- node [auto,swap] {5} (a_2);
  \draw[->] (a_2) -- node [auto] {3} (a_4) -- node [auto] {$-2$} (a_5);
  \draw[->] (a_2) -- node [auto,swap] {12} (a_3);
  \draw[->] (a_3) -- node [auto,swap] {0} (a_4);
  \draw[->] (a_1) -- node [auto] {$x$} (a_6) --
  \draw[->,dashed] (a_6) -- (a_7);
  \draw[->,red,thick] (a_1) to [out=0,in=-100] (a_4);
\end{tikzpicture}
```

Eksempel

6.38



METAPOST

Er et grafisk programmeringssprog. Man kan sige at det minder lidt om L^AT_EX, man får bare simpel PostScript ud i stedet. Jeg laver generelt alle mine tegninger i METAPOST. Eneste ulempe ved METAPOST er at fejlmeddelelserne fra programmet, kan være meget svære at debugge.

¹⁸ Tak til Martin Heller for dette link.

6. Om grafik, flydende objekter og billedtekst

PSTRICKS

En samling af pakker til L^AT_EX som gør det muligt, via PostScript, at kunne tegne mange forskellige typer figurer og grafer. Da man skal over PostScript (dvs. dvips) for at det virker, vil det meste naturligvis ikke fungere hvis man oversætter med pdf_latex. Der er dog visse ting man kan gøre for at kunne anvende sin PSTRICKS under PDF, se <http://tug.org/PSTricks/main.cgi/> for mere information.

Der findes en JAVA frontend som kan anvendes til at lave tegninger med PSTRICKS, se <http://latexdraw.sourceforge.net/>.

gnuplot

Et skummelt meget avanceret tegne/plotte system. Kan outputte i mange sjove formater.

diverse

Der findes diverse andre programmer man kan anvende til at lave tegninger med. Eneste krav er at den kan eksportere/gemme i EPS eller PDF format.

Navne til listen modtages gerne.

Sidebemærkning 6.27. I 2007 udkom den udgave af *The L^AT_EX Graphics Companion* udkommet, se [Goossens et al. \(2007\)](#). Den beskriver grundlæggende brugen af grafik i L^AT_EX, samt forklarer ret udførligt om f.eks. METAPOST og PSTRICKS.

Bogen ser dog helt bort fra TiKz/PGF, primært fordi TiKz/PGF er først er kommet til i den sidste del af 0-erne.

Hvis vi skulle anbefale noget man skulle tage et nærmere kig på, så bliver det TiKz/PGF, det ser ud til at udvikle sig til noget meget meget kraftfuldt. Nærmere behandling af TiKz/PGF ligger dog uden for rammerne af denne bog. Se i stedet [Tantau \(2007\)](#) samt <http://www.texample.net/tikz/examples/>.

Mange Linux og MAC baserede programmer har desuden muligheden for at eksportere/printe figurer/tegninger til EPS eller PDF format. Denne funktionalitet mangler ofte på Windows. Visse programmer siger de kan eksportere/printe til EPS, men det viser sig ofte at være en sandhed med modifikationer.

På Windows kan man i stedet installere en EPS printerdriver. Dvs. det er en virtuel printer hvor man i stedet for at sende noget til en fysiskprinter, så får man i stedet en EPS-fil ud. Med en sådan driver burde der ikke være de store problemer med den genererede EPS-fil. Via <http://www.adobe.com/support/downloads/product.jsp?product=44&platform=Windows> (eller googl efter »*postscript printer driver adobe*«) skulle man kunne downloade en PostScript printerdriver til Windows. Denne installerer man, og så angiver man i engenskaber at man ønsker en eps-fil (eller noget i den retning).

Ved hjælp af en kvalitets EPS-driver i Windows kan man sagtens lave figurer i f.eks. Excel og så inkludere dem i sit L^AT_EX dokument.

Om tabeller

Indenfor mange områder er (specielt data-) tabeller vigtige dele af det at skrive et værk, men desværre ser man ofte at forfattere bare smider dette hæmningsløst sammen uden at tænke på hvordan data præsenteres bedst overfor læseren

»Det er godt nok for mig«

»Øh, nej, det er det bestemt ikke...!«

Typografi-eksperten Robert Bringhurst skriver i [Bringhurst \(2002\)](#) side 70 at:

»Tables are notoriously time-consuming to typeset, but the problems posed are often editorial as much as typographic. If the table is not planned in a readable form to begin with, the typographer can render it readable only by rewriting or redesigning it from scratch.«

Konklusionen er altså at man som forfatter bør bruge lidt mere tid på at sætte sine tabeller op på en pæn og læselig måde, således at de er nemmere at forstå eller fortolke for læseren. Den gode nyhed er så at det faktisk ikke er så svært endda, man skal bare kende de rigtige redskaber.



Dette kapitel handler – åbenlyst – om tabeller i L^AT_EX. Vi vil starte med en introduktion til hvordan man laver tabeller så alle har styr på syntaksen. Herefter vil vi se på forskellige redskaber som enten kan noget smart til specielle situationer eller som generelt kan hjælpe med at lave flotte tabeller. Vi starter dog med nogle forslag til hvordan man bør opbygge datatabeller.

I kapitlet vil vi koncentrere os om formateringen af datatabeller og lignende. Der er naturligvis også andre typer af tabeller, men da de ikke er lige så almindelige for vores publikum har vi valgt at behandlingen af disse ligger uden for rammerne af bogen.

7.1 Guidelines vedr. opbygning af en (data)tabel

Det følgende er en serie gode råd vedr. opbygningen af en god og læsbar (data)tabel, inspiration er hentet fra [Ritter \(2002\)](#), [Chicago \(2003\)](#) og [Bringhurst \(2002\)](#).

– **Vær sikker på at tabellen er relevant at inkludere her**

Hvis tabellen er af en mere supplerende natur, ville det sikkert være bedre at placere den i et appendix.

– **Datatabeller skal bruges til at sammenligne data så overvej hvordan dette gøres bedst**

Det er ikke altid bedst bare at klaske data sammen i en stor pærevælling. Overvej hvad det er du gerne vil sige med denne tabel. Ville læseren få mere ud af data hvis man i stedet lavede en form for graf?

– **Nummerér alle tabeller**

Dette er både for egen skyld, men også for læseren som nemmere kan henviser til en tabel vedkommende er i tvivl om.

7. Om tabeller

– *Placér caption (og dermed nummerering) over tabellen*

Dette er der to årsager til: (1) Har man noter efter tabellen, så er disse fint separeret fra captionen, samt (2) ved at placere caption før tabellen gør det ikke noget at tabellen fylder flere sider.

– *Hold captionteksten kort og i generelle termer*

Caption bruges til at identificere tabellen, se den som en form for *overskrift*. Den skal derfor ikke indeholde nogen fortolkning af data, dette hører i stedet hjemme i teksten. Hvis der er brug for en speciel forklaring til noget af data, så læg det i en tabelnote. Hvis alle data i tabellen bruger samme enhed kan dette nævnes som en del af caption (ofte i en parentes).

– *Hold søjleoverskrifterne så korte som mulige*

Anvend evt. en forkortelse og forklar eller uddyb denne i en tabelnote. Hvis data i søjlen har samme enhed nævnes denne i søjleoverskriften, evt. på en helt ny linie (gør det i så fald for alle datasøjlerne).

– *Rækkeoverskrifterne bør holdes korte og præcise*

Hvis der er behov for det kan man skrive dem over flere linier, eller give en længere forklaring i en tabelnote. Hvis rækken har en fælles enhed nævnes den her (i parentes). Sørg også for at de enkelte overskrifter harmonerer med hinanden, dvs. undlad at være meget specifik et sted og mere generel et andet sted i samme tabel.

– *Hold brug af streger på et minimum*

Lodrette og dobbelte linier anvendes ikke mere. En tabel som er ordentligt balanceret med luft/afstand er meget nemmere at overskue. Især lodrette linier forstyrrer mere end de hjælper.

Hold også de vandrette streger på et niveau hvor man kun har de streger som er relevante for forståelsen af data. Ofte vil en linie før og efter tabellen samt en linie under søjleoverskrifterne være nok.

Underopdeling af tabellen kan foretages med luft eller linier som ikke dækker over første søjle (den med rækkeoverskrifterne).

– *Brug tabelnoter til at give ekstra informationer vedr. tabellen*

Dette være sig (i) angivelse af kilde, hvis man ikke selv har lavet data, (ii) generelle noter til tabellen samt (iii) specifikke noter til enkelte dele af data, bemærk at disse noter skal nummereres på en måde så de ikke forveksles med fodnoter.

Brug gerne disse noter til at give en nærmere forklaring til noget af data hvis dette er specielt relevant.

– *Juster data i cellerne således at det er nemme at sammenligne*

Dette vil for det meste betyde, at indeholder tallene i en søjle et decimalkomma, da justeres søjlen så disse kommer over hinanden. Alternativt kan data højrejusteres hvis data er meget blandet. (Venstrejustering ses sjældent.)



Vedr. dataformatering er der flere ting som vi vil forvente af en formateringsløsning:

- (i) Indgangene skal være i matematik-mode, da der nemt kan forekomme negative tal, eller matematiske konstruktioner (så som $\times 10^6$)
- (ii) Vi skal kunne stille tallene op således at f.eks. decimaltegn kommer under hinanden.
- (iii) Afrunding samt omformatering input (f.eks. $3.4e-7$ til 3.4×10^{-7})

En læserundersøgelse

- (iv) Det ville være rart hvis man nemt kan lave overskrifter (som jo *ikke* skal sættes i matematik-mode).

I den forbindelse har vi yderligere et par ekstra anbefalinger

- (a) Hvis tallene er sammenlignelige med decimaler, så juster data således at decimalerne står over hinanden.
- (b) Hvis data ikke indeholder decimaler, eller hvis de slet ikke er sammenlignelige anbefales det at man højrejusterer data. Alternativt kan data centreres, men man bør i så fald lave en sammenligning først.
- (c) Hvis data i søjlen har decimaler og er næsten-sammenlignelig, dvs. det er kun nogle få celler i søjlen som ikke kan sammenlignes med resten, så kan man justere på decimalen, og så f.eks. centrere de få celler som ligger udenfor.

7.2 En læserundersøgelse

Vi lægger ud med en lille opinionsmåling. I tabel 7.1 og tabel 7.2 finder man to tabeller som præsenterer samme datasæt. Den ene tabel er lavet som de fleste nybegyndere ville lave tabellen, mens den anden tabel er (forsøgt) sat op således at layouttet fremstår mere professionelt.

Spørgsmålet er nu hvilken en af de to tabeller er mest behageligt at læse?

Tabel 7.1: Almindelig datatabel som de fleste begyndere ville starte med at skrive den.

		Nedre 95%	Øvre 95%	
$\mu_i - \mu_j$	$\bar{x}_i. - \bar{x}_j.$	konfidensgrænse	konfidensgrænse	Signifikant
$\mu_3 - \mu_4$	35.23	11.63	58.83	Ja
$\mu_3 - \mu_1$	58.77	35.17	82.37	Ja
$\mu_3 - \mu_2$	91.15	67.56	114.75	Ja
$\mu_4 - \mu_1$	23.54	-0.06	47.14	Nej

Tabel 7.2: Forsøg på at lave en professionel udgave af Tabel 7.1.

$\mu_i - \mu_j$	$\bar{x}_i. - \bar{x}_j.$	Nedre 95%*	Øvre 95%*	Signifikant
$\mu_3 - \mu_4$	35.23	11.63	58.83	Ja
$\mu_3 - \mu_1$	58.77	35.17	82.37	Ja
$\mu_3 - \mu_2$	91.15	67.56	114.75	Ja
$\mu_4 - \mu_1$	23.54	-0.06	47.14	Nej

* Procentsatsen henviser til konfidensgrænsen.

De anvendte data er venligst udlånt af Jørgen Granfeldt, opsætningen er dog ændret en smule. Der er flere problemer med at præsentere data som i Tabel 7.1:

- (1) Lodrette streger hører slet ikke hjemme i tabeller! Det har vist sig at de faktisk nedsætter læsehastigheden (thi øjnene gerne vil følge strengen nedad og derfor skal vi koncentrere os mere for at forstå data). Desuden får linierne data til at ligne et Excel-ark, hvilket det jo ikke er, det er en visuelt fremlæggelse af data, en datatabel, dette er noget helt andet.

7. Om tabeller

- (2) De vandrette linierne kommer alt for tæt på data i de enkelte celler, og der er for mange af dem.
- (3) Data vil i dette tilfælde være mere sammenlignelig hvis decimaltegnet stod over hinanden i søjlerne.

På grund af kommentarerne (1) og (2) vil vi slet *ikke* forklare hvordan man laver hhv. lodrette linier samt L^AT_EXs almindelige vandrette linier i tabeller. Vi vil i stedet koncentrere os om at forklare hvordan man laver konstruktioner som vi ser i Tabel 7.2 på foregående side.

7.3 Tabelsyntaks

I L^AT_EX opskriver vi tabellens data rækkevis, cellerne adskilles med tegnet »&« og rækker afsluttes med »\\« (som her skal tolkes »start en ny række«). Alt dette skrives indeni et specielt indrettet environment. Nogle vil sikkert genkende syntaksen, faktisk er de fleste fler-linie matematikenvironments bygget omkring matematiske tabelkonstruktioner. Her er et hurtigt eksempel, hvor vi har en tabel med tre søjler, med hver sin justering:

Eksempel 7.1

```
\begin{tabular}{rcl}
aaa & bbb & ccc\\
a   & b   & c   \\
\end{tabular}
```

aaa	bbb	ccc
a	b	c

Det mest almindelige tabel-environment er *tabular*. Nogle ville måske have gættet på *table*, men dette environment anvendes i stedet til at pakke omkring tabellen for at lade den flyde omkring, så samme måde som vi tidligere har beskrevet environmentet *figure* for figurer.

Environmentet *tabular* tager et obligatorisk argument som anvendes til at specificere hvor mange søjler den skal forvente (samt evt. information om hvordan de skal formateres). Det tager også et positionsargument, som vi dog ikke skal kommentere her, det er kun relevant hvis man f.eks. har to tabeller på samme linie. Den almindelige syntaks for en tabel bliver altså:¹

```
tabular \begin{tabular}[\langle pos \rangle]{\langle tabel preamble \rangle}
        \langle evt. en linie \rangle
        \langle celle 1 \rangle & \langle celle 2 \rangle & \dots & \langle celle n \rangle
        \\
        \langle evt. en linie \rangle
        \dots
        \end{tabular}
```

I Eksempel 7.1 så vi de tre mest fundamentale søjle typer.

Tip 7.1. Hvis der er plads til det, er det en god ide skrive »&«-erne over hinanden i kildekoden. Det giver et godt overblik. Nogle editorer kan hjælpe med dette. En anden fix ting, er at man i modsætning til i matematik, *gerne* må have blanke linier mellem rækkerne i tabeller. Dette kan igen hjælpe med overblikket.

¹ Som vi tidligere har foreslået vedr. matematik, så er det en fordel at skrive »\\« på linier for sig selv.

7.3.1 Flydende tabel

For at få en almindelig tabel (vi skal senere se nogle ikke helt almindelige konstruktioner) til at flyde, skal vi som ved figurer pakke et environment uden om. Syntaksen er nøjagtigt den samme som ved figurer.

```
\begin{table}[<loc>]
\centering
\caption{<tekst>}
<selve tabellen>
\end{table}
```

Hvor vi naturligvis har placeret caption over tabellen som anbefalet i afsnit 7.1. Man bedes bemærke at luften under \caption nok bliver for lille, når man placerer \caption over tabellen. Dette fikser man nemt via caption-pakken, se afsnittet *Mellemrum før og efter billedtekst* på side 152.

7.3.2 Angivelse af søjlerne

For at L^AT_EX kan trykke tabellen pænt, skal den vide hvor mange søjler der maksimalt »kan« forekomme.² Opskriver man flere søjler i en række end tabellen er konfigureret til får man fejlen:

Extra alignment tab has been changed to \cr

Angivelsen af søjlerne skriver man i det obligatoriske argument til det omkringliggende environment.³ Lad os kalde dette for tabellens »preamble«.

Tabel-preamblen er centreret omkring specifikationen af de enkelte søjler, dette kommer vi til om lidt. Vi kan desuden anvende preamblen til at lægge ekstra luft mellem søjlerne (hvilket tit er en god ide) eller lave om på konfigurationen af søjlen (f.eks. ved at gøre en hel søjle kursiv i ét hug i stedet for at gøre det cellevist).

Her følger de mest almindelige konfigurationer man kan anvende i <søjle preamble> argumentet til **tabular**. Vi starter med dem som altid er tilstede, og lister bagefter de ekstra muligheder vi får med array pakken (eller via *memoir*-klassen). For god ordens skyld skal vi nævne at lodrette streger også normalt specificeres i preamblen, men af pædagogiske årsager vil vi slet ikke forklare dette.⁴ Vi skal senere komme lidt ind på andre pakker som giver ekstra features til tabel-preamblen.

Først starter vi med de specifikationer som *ikke* kræver ekstra pakker.

- l Specificerer én venstrejusteret søjle, uden automatisk orddeling. Søjleens bredde justeres automtisk til bredden af bredeste celle.
- c Specificerer én centreret søjle, uden automatisk orddeling. Samme fortolkning af bredden.
- r Specificerer én højrejusteret søjle, uden automatisk orddeling. Samme fortolkning af bredden.
- p{<bredde>}
 Specificerer en venstrejusteret søjle med den angivne bredde og automatisk orddeling. Man kan ændre justeringen indeni søjlen med en konstruktion som nævnes

² Man behøver ikke udfylde alle søjler i slutningen af en række.

³ Nogle tabel-environments kan have flere obligatoriske argumenter, så her skal man være specielt opmærksom på syntaksen.

⁴ Når vi alligevel anbefaler at man ikke anvender lodrette streger, så hvorfor forklare hvordan man gør.

7. Om tabeller

senere på listen. Et eventuelt tvunget lineskift *skal* ske med »\newline«. Den lodrette placering justeres efter første linie i cellen.

@{<noget>}

Indsætter <noget> i stedet for mellemrummet mellem to søjler. F.eks. kan @{} fjerne mellemrummet før en søjle og @{\quad} kan forøge afstanden.

*{<antal>}{<kode>}

Gentager <kode> <antal> gange. F.eks. er *{3}{lr} lig med »lr|lr|lr«. Så hvis man har mange søjler efter hinanden med samme specifikation, så kan dette lette notationen. Er man rigtig smart kan man definere en genvej som f.eks. angiver hele preamble. Se [Eksempel 7.6 på side 173](#).



Følgende specifikationer kræver array-pakken (eller *memoir*-klassen).

m{<bredde>}

Specificerer én venstrejusteret søjle med automatisk orddeling. I stedet for at justere efter den første linie, centrerer indholdet nu vertikalt i stedet.

b{<bredde>}

Specificerer én venstrejusteret søjle med automatisk orddeling. Svarer til »p{<bredde>}«, her justeres dog efter sidste linie.

>{<kode>}

Anvendes *før* f.eks. l, c, r, p, m, b. Den indsætter så <kode> i *starten* af hver celle i søjlen. Man kan tilsidesætte indsættelsen af <kode> i en specifik celle via \multicolumn, se afsnit [7.3.3 på side 173](#). F.eks. vil »>{\itshape} l« lave en venstrejusteret søjle, hvor alle cellerne sættes med kursiv tekst.

<{<kode>}

Anvendes *efter* l, c, r, p, m, b. Indsætter så <kode> i *slutningen* af hver celle i denne søjle.

!{<noget>}

Indsætter <noget> mellem søjlerne. Svarer til @{<noget>}, men her tilføjer man til bredden mellem søjlerne i stedet for at starte helt fra nul.

Sidebemærkning 7.2. Et par bemærkninger:

- (a) Husk at »<code></code>« og <code></code>« udføres *hver for sig*. En almindelig fejl er at forsøge med <code></code>, altså forsøge at anvende <code></code> omkring hver indgang, hvilket fejler fordi de krøllede parenteser ikke er velbalancerede. I dette tilfælde ville <code></code> været nok.
- (b) Cellerne som er angivet med bredde, f.eks. med <code></code>, er venstrejusterede som standard. Hvis man ønsker at centrere data, kan man anvende <code></code>, til højrejustering anvendes <code></code>.

Her skal man dog passe en smule på! <code></code> har den lille feature at den piller ved kommandoen <code></code>. I tabeller betyder denne <code></code>, men efter <code></code> betyder den nu noget helt andet. Det anbefales derfor at man beskytter <code></code> via <code></code>, se [Eksempel 7.5 på næste side](#), derved kommer <code></code> tilbage til sin tabelbetydning.

Ud over <code></code> skal man tage tilsvarende forholdsregler ved brugen af <code></code>, <code></code>, <code></code>, <code></code> og <code></code>.

Teknisk set er det kun nødvendigt at man tilføjer <code></code> i *sidste* søjle, og kun hvis denne søjle anvender en af de problematiske konstruktioner.

- (c) Makroerne <code></code> og specielt <code></code> (fra pakken <code></code>) er en stor fordel når man har søjler med masser af tekst og hvor teksten automatisk deles. De nævnte to makroer fjerner helt eller delvist orddelingen, den sidste er især smart thi den sørger for at der kun kommer orddeling hvis vi har brug for det, og ellers lader teksten være u-orddelt hvilket ofte er behageligere at læse.

- (d) Der er en lille hage ved anvendelsen af <code></code> og <code></code>, f.eks. i

<code></code>

vil <code></code> blive placeret yderst til højre i den første søjle. Dette kan i visse situationer få 'interessante' konsekvenser. Senere vil vi se lidt nærmere på tricks til at få mere luft mellem søjlerne.

Sidebemærkning 7.3. Bare lige for ordens skyld skal vi lige minde om at man altså ikke kan skrive enhver konstruktion i en tabelcelle. Har man noget som automatisk skal skifte linie, f.eks. en liste, så bør man anvende en p-søjle, eller en *varwidth* konstruktion, se afsnit [B.5 på side 371](#).

Lad os komme med nogle eksempler, husk [Eksempel 7.1 på side 168](#). Først illustrerer vi hvordan bredden af cellerne ofte svarer til bredeste indhold, samt hvordan man kan få automatisk linieombrydning med <code></code>, bemærk desuden hvordan vi kan fremtvinge et lineskift.

```
\begin{tabular}{c @{\quad} p{2cm}}
  lang tekst & en meget lang eksempel\line
  A & B \newline C
\end{tabular}
```

lang tekst	en meget lang eksempel\line
A	B
	C

Eksempel 7.2

I [Eksempel 7.2](#) kan vi se at teksten i højre søjle kan blive grim, da søjlen forsøger at holde lige venstre- og højrekant. Dette fikser vi via <code></code> og <code></code>, bemærk her brugen af <code></code>.

7. Om tabeller

Eksempel

7.3



```
\usepackage{ragged2e,array}
----- slut preamble -----
\begin{tabular}%
{ >{\RaggedRight\arraybackslash}p{2cm} }
en meget lang eksempellinie\
den næste linie fylder endnu mere
\end{tabular}
```

en meget
lang eksem-
pellinie
den næste
linie fylder
endnu mere

Dette med at kunne indsætte noget kode i slutningen af en celle, kan jo f.eks. anvendes til at sætte en hel søjle i matematik-mode, så er man fri for at gøre det pr. håndkraft i hver eneste celle. Se desuden afsnittet om `\multicolumn` (afsnit 7.3.3 side 173). Bemærk hvordan vi har fåt lagt en \$ ind i starten og slutningen af hver celle i første søjle.

Eksempel

7.4



```
\usepackage{amssymb,array}
----- slut preamble -----
\begin{tabular}{>{$} l <{$} p{5.5cm}}
\mathbb{R} & Reelle tal\
\mathbb{C} & Komplekse tal\
\mathrm{SU}(n) & Gruppe af specielle
unitære $n\times n$-matricer\
\dots & \dots
\end{tabular}
```

\mathbb{R}	Reelle tal
\mathbb{C}	Komplekse tal
$\mathrm{SU}(n)$	Gruppe af specielle unitære $n \times n$ -matricer
...	...

Tip 7.4. Skriver man sig en symbolliste som ovenfor, er det nok en ide at kombinere det med *longtable* (beskrevet i afsnit 7.5.2 på side 183), så det kan deles over flere sider.

Symbolister og ordforklaringer kan også laves på andre måder, se afsnit 9.13 på side 258.

Her er et eksempel som viser hvordan man kan få indholdet i en p-søjle centreret automatisk. Bemærk igen brugen af `\arraybackslash`.

Eksempel

7.5



```
\usepackage{array,booktabs}
----- slut preamble -----
\begin{tabular}%
{ >{\centering\arraybackslash}p{3.5cm} }
noget mega langt
som ikke kan være i et 3.5\,cm felt \
\midrule
endnu en linie
\end{tabular}
```

noget mega langt som
ikke kan være i et
3.5 cm felt

endnu en linie

Sidebemærkning. Herfra antages brugen af `array`-pakken (eller *memoir*-klassen).



De specielle meningsfyldte tegn vi anvender til at angive *tabelpreamblen* med, kan man også selv lave. Dette kan f.eks. anvendes til at lave en genvej til en preamble som man tit anvender. Syntaks:

```
\newcolumnstype      \newcolumnstype{\langle tegn \rangle}[\langle antal argumenter \rangle]{\langle kode \rangle}
```

Overskriftsceller

Man kunne så lave tabel-preamblen fra [Eksempel 7.4](#) på forrige side med

```
\newcolumnntype{B}[1]{>{$}l<{$} p{#1}}
\begin{tabular}{B{5.5cm}}
...
```

Eksempel
7.6

Har man mange tabeller med samme ‘preamble’ så kan man jo bare lave f.eks.

```
\newcolumnntype{A}{ l c c c c c c c r}
...
\begin{tabular}{A}
...
\begin{tabular}{A}
...
```

Eksempel
7.7

Vi skal senere se mere komplekse tabelpreambler.

7.3.3 Overskriftsceller

Før eller siden får man desuden brug at lave celler som spænder over flere søjler, f.eks. til generelle overskrifter. Dette opnås via

```
\multicolumn{⟨antal søjler⟩}{⟨søjle konfiguration⟩}{⟨tekst⟩}
```

`\multicolumn`

Hvor *⟨søjle konfiguration⟩* svarer til det man skriver i tabel-preamblen bare gældende for denne ene celle, oftest c – husk at dette nu er én meget bred celle, så der skal kun konfigureres en enkelt søjle. Selve `\multicolumn` placeres i den første celle den skal erstatte. I det næste eksempel har vi en bred celle som spænder over tre søjler. Bemærk hvordan det efterfølgende `&` markerer overgangen til søjle 4.

```
\begin{tabular}{cccc}
\multicolumn{3}{c}{12345678} & B \\
A & B & C & D \\
\end{tabular}
```

12345678	B		
A	B	C	D

Eksempel
7.8

Sidebemærkning 7.5. Man kan sagtens anvende `\multicolumn` til at pille ved en celle som kun spænder én søjle. I så fald vil den angivne *⟨søjle konfiguration⟩*-specifikation for netop denne celle tilsidesætte specifikationen for hele søjlen. Denne feature anvendes ofte til lokalt at ændre formateringen af en celle. F.eks. hvis søjlen var sat i matematik-mode, og man gerne ville lave en overskrift, så anvendes f.eks. `\multicolumn{1}{c}{...}` til at lave titelcellerne.

Sidebemærkning 7.6. Er man gået i mod vores anbefaling og alligevel anvender lodrette streger i en tabel, så skal man lige være opmærksom på at den lodrette streg tilhører cellen til venstre for den (cellen yderst til venstre kan have to). Så anvender man `\multicolumn` i en søjle med en lodret linie, så forsvinder denne i cellen med `\multicolumn`, med mindre man sørger for at huske at tage linien med i *⟨søjle konfiguration⟩*

7. Om tabeller



Når man nu har rodet med `\multicolumn` kunne man jo også få brug for en `\multirow`, hvilket man får via standardpakken `multirow`,⁵ se [Leichter og van Oostrum \(2004\)](#).

`\multirow` `\multirow{\langle antal rækker \rangle}{\langle bredde \rangle}[\langle vmove \rangle]{\langle tekst \rangle}`

`\multirowsetup`

Teksten vil altid blive skrevet `\raggedright` (i.e. venstrejusteret ingen orddeling), men kan ændres ved at omdefinere `\multirowsetup`. Boksen hvori teksten befinder sig vil blive vertikalt centret inden for det område `\multirow` råder over. `[\langle vmove \rangle]` kan anvendes til at justere på dette, `\langle vmove \rangle` skal være en positiv eller negativ længde.

Eksempel 7.9



```
\usepackage{multirow,booktabs}
----- slut preamble -----
\begin{tabular}{l c} \toprule
\multirow{2}{1cm}{0000} & 1 \\
& 2 \\
& 2 \\ \midrule
aa & 3 \\ \bottomrule
\end{tabular}
```

0000	1
	2
aa	3

Eksempel med første søjle centreret (bemærk hvordan første celle under linien er venstrejusteret, mens teksten i `multirow`-cellen er centreret):

Eksempel 7.10



```
\usepackage{multirow,booktabs}
\renewcommand\multirowsetup{\centering}
----- slut preamble -----
\begin{tabular}{l c} \toprule
\multirow{3}{1cm}{noget txt} & 1 \\
& 2 \\
& 3 \\ \midrule
aa & 4 \\ \bottomrule
\end{tabular}
```

noget	1
txt	2
	3
aa	4

7.3.4 Linier i tabeller

`booktabs`

Tilbage i kommentarerne til Tabel 7.1 på side 167 nævnte vi at L^AT_EXs standard tabellinier ikke er ret gode, da de kommer for tæt på teksten, så dem hopper vi let og elegant over. Vi vil i stedet anbefale brugen af linierne fra `booktabs`-pakken (de er direkte inkluderet i `memoir`-klassen).

Pakken `booktabs` er skrevet for at kunne lave den slags tabeller som man anvender i high-end publikationer og bøger, se eventuelt [Chicago \(2003\)](#), [Ritter \(2002\)](#) eller [Bringham \(2002\)](#). Selve `booktabs` stiller forskellige vandrette linier til rådighed, det specielle ved disse er at der er en anden luft over og under disse streger (engelsk: `rules`).⁶ Da `booktabs` kun har med streger at gøre, kan pakken anvendes sammen med stort set alle tabelkonstruktioner, samt desuden i visse matematik sammenhænge. Syntaksen for makroerne følger herunder hvorefter vi ser på deres anvendelse.

⁵ Da celler som spænder over flere rækker ikke er lige så almindelige som celler som spænder over flere søjler, er `\multirow` ikke en del af L^AT_EX-formatet.

⁶ En spændende effekt af denne ekstra luft, er at de nu slet ikke kan anvendes sammen med L^AT_EXs almindelige konstruktion til lodrette linier, så har man endnu en grund til at lade være. Se evt. [Eksempel 7.14 på side 176](#).

Booktabs makrosyntaks

```
\toprule[⟨tykkelse⟩]
\bottomrule[⟨tykkelse⟩]
\midrule[⟨tykkelse⟩]
\cmidrule[⟨tykkelse⟩](⟨trim⟩){⟨S1-S2⟩}
\specialrule{⟨tykkelse⟩}{⟨luft over⟩}{⟨luft under⟩}
```

Syntaksen med (⟨trim⟩) kan udelades. ⟨luft over/under⟩ skal være en længde. Indsæt ekstra luft før eller efter en række:

```
\addlinespace[⟨længde⟩]
```

Standardtykkelsen af linierne kan justeres globalt ved at pille ved længderne:

```
\heavyrulewidth
\lightrulewidth
\cmidrulewidth
```

Brug af fuldbredde-linier

Den mest almindelige måde at anvende linierne er at man starter tabellens data med en `\toprule` og afslutter den tilsvarende med `\bottomrule`. I selve tabellen kan vi opdele via meningsfyldte `\midrule`'s, måske krydret med `\cmidrule` (som laver linier som ikke spænder over hele tabellens bredde).

Forskellen mellem `\toprule` og `\bottomrule` er luften over og under. Top- og bundstregerne er normalt tykkere (`\heavyrulewidth`) end `\midrule` (`\lightrulewidth`).

Et simpelt eksempel kan ses i [Eksempel 7.11](#). Man observerer at linierne står som det aller første i første celle af den række hvor linien skal stå *over*. En linie er en speciel række, så man *skal* huske at afslutte rækken *før* linien med »\«. Dette er specielt vigtigt i forbindelse med bundlinien.

```
\usepackage{array,booktabs}
----- slut preamble -----
\begin{tabular}{l !{\quad} r}
\toprule
Post      & Udgift (kr) \\ \midrule
Rådne æg  & 35.53      \\
And       & 250.00     \\
Gnu       & 11.50      \\ \bottomrule
\end{tabular}
```

Post	Udgift (kr)
Rådne æg	35.53
And	250.00
Gnu	11.50

Eksempel 7.11


Sidebemærkning 7.7. Bemærk hvordan vi jævnfør anbefalingerne i starten, angiver enheden i søljoerskriften.

`\addlinespace` kan anvendes alle steder til at give lidt mere afstand mellem to rækker, før eller efter en linie etc. `\addlinespace`

7. Om tabeller

Eksempel 7.12



```
\usepackage{array,booktabs}
----- slut preamble -----
\begin{tabular}{l !{\quad} r}
\toprule
Post & Udgift (kr) \\midrule
Frimærker & 235.53\\
Konvolutter & 153.25\\
Kopiering & 423.50\\midrule
\addlinespace
Total & 830.28\\bottomrule
\end{tabular}
```

Post	Udgift (kr)
Frimærker	235.53
Konvolutter	153.25
Kopiering	423.50
Total	830.28

Afbrudte linier

`\cmidrule` Makroen `\cmidrule` tillader os at lave linier som ikke spænder hele bredden af tabellen. Dette anvendes oftest til at give et bedre overblik vedr. overskrifter i tabellen.

Dog har `\cmidrule` en lidt sjov syntaks, i det obligatoriske argument skal man med » $\langle tal1 \rangle$ - $\langle tal2 \rangle$ « angive at strengen skal starte fra *venstre* kant af søjle $\langle tal1 \rangle$ og slutte ved *højre* kant af søjle $\langle tal2 \rangle$.⁷ Dvs. 1-2 strækker sig over de første to søjler, 2-4 fra søjle 2 til og med søjle 4 og 6-6 dækker kun søjle 6. Det underlige ($\langle trim \rangle$)-argument, hvor $\langle trim \rangle$ består af mulighederne *r*, *l* eller *rl* som angiver at man vil trimme denne streg til højre, venstre eller i begge sider, sammenlign de to tabeller i [Eksempel 7.13](#).

Eksempel 7.13



```
\usepackage{array,booktabs}
----- slut preamble -----
\begin{tabular}{l r r}
\toprule
\multicolumn{2}{c}{Overskrift} & Pris \\
\cmidrule{1-2}\cmidrule{3-3}
Punkt & Antal & kr \\
\midrule
\addlinespace[10mm]
\toprule
\multicolumn{2}{c}{Overskrift} & Pris \\
\cmidrule{1-2}\cmidrule(l){3-3}
Punkt & Antal & kr \\
\midrule
\end{tabular}
```

Overskrift		Pris
Punkt	Antal	kr
Overskrift		Pris
Punkt	Antal	kr

Som det ses af [Eksempel 7.13](#) er $\langle trim \rangle$ delen meget anvendelig når man har to `\cmidrule`'s som støder op mod hinanden, hvis vi ikke havde tilføjet (*l*) så ville man ikke kunne se at der var tale om to linier. Man kan selv styre hvor meget »gab« der skal være via $r\langle længde \rangle$ eller $l\langle længde \rangle$. En anden mulighed for at lægge luft mellem to af disse korte linier er direkte at indsætte en ekstra søjle. Dette ser vi på i et senere afsnit.



Her til slut må vi hellere overbevise læseren om at man ikke skal anvende lodrette linier sammen med `booktabs`-linierne.

Eksempel 7.14

Overskrift	Tekst
noget	andet

⁷ Søjlerne tælles fra 1 (bare lige af hensyn til datalogerne, der jo som bekendt tæller fra 0).

Lodrette linier i forbindelse med booktabs-linierne kan dog manuelt simuleres i de få tilfælde hvor det er berettiget, men det kommer vi ikke nærmere ind på her.

7.4 Dataformatering

Langt hovedparten af alle tabeller vil være oversigter over tal i forskellige afskygninger. Det ville det være rart at kunne få L^AT_EX til at formatere disse tal på passende vis for os.

Dette er emnet i dette afsnit. Vi skal se nærmere på to pakker `dcolumn` og `siunitx`. Den første er af ældre dato, og funktionaliteten er indbygget i *memoir*-klassen. Den anden pakke er faktisk til formatering af tal og enheder, men giver også en langt mere avanceret dataformatering end `dcolumn`.



7.4.1 Automatisk decimaljustering med `dcolumn`-pakken

Pakken `dcolumn` har den fordel at den er en standarddel af L^AT_EX-installationerne, og er desuden en del af *memoir*-klassen. Men den er ikke videre fleksibel, så til søjler som er bare en smule mere avancerede anbefaler vi pakken `siunitx`, se afsnit 7.4.3 på side 179.

Pakken `dcolumn` løser et par af de ting vi gerne ville kunne konfigurere. Pakken `dcolumn` giver os en ny søjletype bemævnt `D`, som tager tre argumenter:

$$D\{\langle alignmenttegn \rangle\}\{\langle output \rangle\}\{\langle decimaler \rangle\}$$

Argument forklaring:

$\langle alignmenttegn \rangle$ er her det tegn (og der må kun anvendes ét tegn) som L^AT_EX skal finde i hver celle i denne søjle. Dette vil ofte være `»`.`«` (punktum) eller `»,``«` (komma).

$\langle output \rangle$ er det L^AT_EX skal indsætte i outputtet i stedet for det fundne $\langle alignment\ tegn \rangle$, ofte er det bare det samme som $\langle alignment\ tegn \rangle$ 'et, men kan være hvad som helst, se senere eksempel.

$\langle decimaler \rangle$ anvendes til at angive hvordan det maksimale antal decimaler forholder sig. Man kan gøre flere ting, men de flere vil anvende syntaksen `» $\langle antal\ tegn\ før\ decimal \rangle$. $\langle antal\ tegn\ efter \rangle$ «`. I denne konfiguration vil det formaterede materiale så danne en speciel søjle som så centreret under overskriften. I søjlen vil det som står før $\langle alignmenttegn \rangle$ blive højrejusteret, og det som står efter bliver venstrejusteret. Hvis der ikke bliver fundet noget $\langle alignmenttegn \rangle$ i cellen, formateres data som var det den del som står før decimalen.

Sidebemærkning 7.8. D-søjlen sætter alle cellerne i søjlen i matematik-mode. Så overskrifter skal man selv sørge for at beskytte.⁸

Løsning: Indkapsling af hver D-søjleoverskrift i en `\multicolumn{1}{c}{...}`-konstruktion. Evt. anvendt med følgende genvej:

```
\newcommand\mc[1]{\multicolumn{1}{c}{#1}}
```

Således at overskriften skrives `\mc{overskrift}`.

Kodetip
7.1

⁸ `siunitx` har en smartere metode.

7. Om tabeller

Tip 7.9. Man kommer sikkert ofte til at anvende `D{.}{.}{x.y}` hvilket er træls at skrive hele tiden, så lav en ny column specifier.

Eksempel 7.15

```
\newcolumntype{d}[1]{D{.}{.}{#1}}
```

og anvend `d{3.4}` i stedet.

Vi vil nu lave et eksempel, som viser en del af de ting man kan gøre med `dcolumn`, blandt andet ændre alignmenttegnet.

Husk at »-« (minus) eventuelt også skal tælle når man tæller antallet af tegn før og efter decimalen. Her kommer et eksempel hvor vi også leger lidt med *output*.

Eksempel 7.16



```
\usepackage{amsmath,dcolumn,booktabs}
\newcommand\mc[1]{\multicolumn{1}{c}{#1}} % sparer plads
----- slut preamble -----
\begin{tabular}{ D{-}\text{--}{2.2} D{.}{.}{3.2} D{x}{\times}{6.4} D{x}{4.0}
\toprule
\mc{Periode} & \mc{Tal} & \mc{Noget i MeV} & \mc{Klassisk snyd}\
\midrule
9-10 & 234.43 & 2.33x10^3 & 12 \\
10-11 & -34.4 & -4.00x10^3 & 124 \\
13-17 & .43 & 0.33x10^3 & 3452 \\
18-02 & 999 & \mc{---} & 0 \\
\bottomrule
\end{tabular}
```

Periode	Tal	Noget i MeV	Klassisk snyd
9–10	234.43	2.33×10^3	12
10–11	-34.4	-4.00×10^3	124
13–17	.43	0.33×10^3	3452
18–02	999	—	0

Det man skal lægge mærke til i [Eksempel 7.16](#) er

- hvordan alignmenttet forløber i søjle to, især når der ikke er noget på venstre hhv. højre side af decimalen.
- første søjle hvor man kan se hvordan man kunne lave et skema, bemærk brugen af `\text{--}` for at kunne lave den rigtige intervalstreg.
- søjle tre viser hvordan man kan erstatte alignment tegnet i kildekoden (x) med et andet outputtegn. Bemærk brugen af `{\times}`, de to `{}`-par sørger for at spacingen omkring `\times` er helt rigtig.
- brugen af `\multicolumn{1}{c}{---}`⁹ for at kunne markere en tom indgang. Dette er normalt måden man gør dette på, om man så vil anvende `--` eller `---` er en smagssag.
- at man i søjle tre anvender ikke helt åbenlyse værdier for antallet af tegn på hver sin side af decimalen. Dette trick kan man anvende når man gerne vil have noget mere luft ind i søjlen, thi de almindelige `!\dots` eller `@{\dots}` fungerer ikke sammen med en `D`-søjle.

⁹ Her gennem vores hjemmelavede forkortelse `\mc{---}`.

7. Om tabeller

så vi kigger lidt på nogle eksempler, og vil i stedet henvide til [Wright \(2010\)](#) samt afsnit [10.2.1 på side 263](#).

Vi vil se på afrunding, påfyldning af nuller, håndtering af overskrifter¹¹, generel justering, håndtering af afvigelser, inkonsistente data.

Vi starter simpelt. Som standard vil `siunitx` justere på decimalen, hvor vi er ligeglade om decimalen er komma eller punktum, begge erstattes med punktum, hvilket dog kan justeres til f.eks. komma, hvilket vi gør i anden søjle (dette er egentlig korrekt dansk typografi, selvom de fleste efterhånden anvender punktum). Tredje søjle viser hvordan man kan justere datasøjlen anderledes under overskriften (bemærk rækkefølgen af de to options, den første sætter nemlig den sidste til en anden værdi end den vi gerne ville have).

Eksempel 7.19



```
\usepackage{siunitx,booktabs}
----- slut preamble -----
\begin{tabular}{S % uden config
S[output-decimal-marker={,}]
S[table-format=3.3,
table-number-alignment=right] }
\toprule
{Noget tekst} & {Noget tekst} & {Noget tekst} \\
\midrule
123.3 & 123.3 & 123.3 \\
45,67 & 45,67 & 45,67 \\
-8.901 & - 8.901 & -8.901 \\
\bottomrule
\end{tabular}
```

Noget tekst	Noget tekst	Noget tekst
123.3	123,3	123.3
45.67	45,67	45.67
-8.901	-8,901	-8.901

Bemærk hvordan indholdet sættes i matematik-mode så minuset kommer ud rigtigt, og hvordan man bare anvender et sæt `{}` til at beskytte overskriften.¹²

Det næste vi ser på er afrunding og påfyldning af nuller.

Eksempel 7.20



```
\usepackage{siunitx,booktabs}
----- slut preamble -----
\begin{tabular}{S
S[table-auto-round,table-format=3.3]
S[table-auto-round,table-format=3.1] }
\toprule
{Noget tekst} & {Noget tekst} & {Noget
tekst} \\
\midrule
123.3 & 123.3 & 123.3 \\
45.67 & 45.67 & 45.67 \\
-8.901 & -8.901 & -8.901 \\
\bottomrule
\end{tabular}
```

Noget tekst	Noget tekst	Noget tekst
123.3	123.300	123.3
45.67	45.670	45.7
-8.901	-8.901	-8.9

Første søjle er som før, nummer to viser hvordan vi automatisk kan fylde nuller på hvis vi har lyst. Man bliver nødt til at fortælle søjlen hvor mange decimaler den skal fylde nuller ud til (via `table-format=3.3`, lidt a la det vi så med `dcolumn`). Bemærk at man kan reservere plads til fortegn via f.eks. `table-format=-3.3`, skulle man have brug for det.

I det næste eksempel vil vi vise hvordan man kan fjerne justeringen på decimalen, samt vise hvordan man kan håndtere irregulære værdier.

¹¹ For det er meget nemmere end med `\mc` som vi så tidligere.

¹² Meget nemmere end med `\mc...`

Dataformatering via pakken siunitx

```

\usepackage{siunitx,booktabs}
----- slut preamble -----
\begin{tabular}{
l S S[table-parse-only,
table-number-alignment=right] }
\toprule
& {Noget tekst} & {Noget tekst}\
\midrule
Tekst & 123,3 & 123,3 \
Tekst & 45.67 & 1200 \
Tekst &
{\numrange[range-phrase={ til }]{10}{25}}
& 23.34567 \
Tekst & -8.901 & 1 \
\bottomrule
\end{tabular}

```

	Noget tekst	Noget tekst
Tekst	123.3	123.3
Tekst	45.67	1200
Tekst	10 til 25	23.34567
Tekst	-8.901	1

Eksempel
7.21

Bemærk hvordan vi anvender `{...}` omkring intervallet for at få den centreret under overskriften. Husk selv matematik-mode hvis den irregulære værdi påkræver det. Observer også hvordan tallet med de mange decimaler skrives, der indsættes en smule luft for at gøre tallet nemmere at læse. Dette kan konfigureres, se [Wright \(2010\)](#). Pt. kan `siunitx` ikke højjustere og centrere denne blok under overskriften så som `dcolumn` kan, men dette vil blive tilføjet til en senere version.

Pakken understøtter naturligvis også ‘videnskabelig notation’:

```

\usepackage{siunitx,booktabs}
----- slut preamble -----
\begin{tabular}{
l S[table-format=-2.2e2] }
\toprule
& {Energi (\si{\mega\electronvolt}) } \
\midrule
Tekst & 1.34e6 & \
Tekst & 12.1e-3 & \
Tekst & 6.1e34 & \
Tekst & +-13.23e3 & \
\bottomrule
\end{tabular}

```

	Energi (MeV)
Tekst	1.34×10^6
Tekst	12.1×10^{-3}
Tekst	6.1×10^{34}
Tekst	$\pm 13.23 \times 10^3$

Eksempel
7.22

Bemærk tricket med `»+«`, dette virker dog ikke til at angive afvigelser. Her skal man i stedet anvende syntaksen `1.23(4)` for at mene 1.23 ± 0.04 . Fremvisningen af denne syntaks kan man så styre på forskellig vis.

```

\usepackage{siunitx,booktabs}
----- slut preamble -----
\begin{tabular}{
S[table-format=2.2(2)]
S[separate-uncertainty, table-format=2.2(2),
table-figures-uncertainty=1] }
\toprule
{Noget tekst} & {Noget tekst}\
\midrule
12.3(5) & 12.3(5) & \
5 & 5 & \
9.87(65) & 9.87(65) \
\bottomrule
\end{tabular}

```

Noget tekst	Noget tekst
12.3(5)	12.3 ± 0.5
5	5
9.87(65)	9.87 ± 0.65

Eksempel
7.23

Her skal man observere forskellen mellem `»(5)«` og `»(65)«` og hvordan alignment vælges.

7. Om tabeller

Man kan desuden lave mange andre interessante talformateringer via `siunitx`, f.eks. farve alle negative værdier i en søjle røde.

Sidebemærkning 7.11. Man kan lave generelle konfigurationer via kommandoen `\sisetup`, f.eks. angive at alle søjlerne skal (med mindre andet er angivet) anvende et bestemt antal decimaler, se [Wright \(2010\)](#) for mere info.

7.5 Andre tabel-environments

Vi nævnte tidligere at *tabular* kun var et blandt andre environments hvori man skriver sit tabelindhold. Vi vil her se på nogle af disse environments.

7.5.1 Tabularx – tabel med fast bredde

`tabularx` Pakken `tabularx` (direkte inkluderet i *memoir*) giver os environmentet *tabularx* samt et nyt tegn til at specificere en søjle »X«. Ideen med environmentet er at man har en tabel, som tildeles en specifik bredde, tabellen behandler først søjlerne som ikke er angivet som X, og herefter fordeles den resterende bredde ligeligt blandt X-søjlerne. Bag X står et `p{speciel bredde}`.

Bredden af tabellen angives som det første af to obligatoriske argumenter til *tabularx*. Den generelle syntaks er derfor:

```
\begin{tabularx}{\tabelbredde}{\tabel preamble}  
<indhold som normalt>  
\end{tabularx}
```

Tip 7.12. Det er en fordel at angive bredden af tabellen i termer af `\linewidth` eller `\textwidth`, så rejstures tingene fornuftigt hvis man senere vælger at ændre marginerne.

Konstruktionen anvendes ofte når man har p-søjler hvor man ikke selv gider specificere en passende bredde. Her følger et simpelt eksempel. Observer at søjle 1 og 3 får den bredde de har brug for og resten er ligeligt fordelt mellem søjle 2 og 4.

Eksempel 7.24



```
\usepackage{tabularx,booktabs}  
----- slut preamble -----  
\begin{tabularx}{0.9\textwidth}{cXc>{\raggedright\arraybackslash}X}  
  \toprule  
  Søjle 1 & Søjle 2 & Søjle 3      & Søjle 4 \\ \midrule  
  noget  & \txt      & noget andet & \txt      \\ \bottomrule  
\end{tabularx}
```

Søjle 1	Søjle 2	Søjle 3	Søjle 4
noget	Masser af tekst som bare er fyld og ikke skal bruges til noget.	noget andet	Masser af tekst som bare er fyld og ikke skal bruges til noget.

Longtable – tabel over flere sider

Konstruktionen er f.eks. yderst anvendelig når man skal stille et regnskab op, der har man ofte to søjler, hvor beløbssøjlen har en fast bredde og hvor man så bare sætter bredden af første søjle til at autojustere via X. Det kunne se sådan ud

```
\begin{tabularx}{\linewidth}{X<{\enspace\dotfill} !{\quad} r}
\toprule
Udgift \let\dotfill\relax & Beløb/kr \\
\midrule
Spisning & 16000\\
Transport m.m. (8 pers) & 2500\\
Bestikkelse & 1000\\
\midrule
\multicolumn{1}{X}{Ialt} & 19500\\
\bottomrule
\end{tabularx}
```

Eksempel
7.25

Udgift	Beløb/kr
Spisning	16000
Transport m.m. (8 pers)	2500
Bestikkelse	1000
Ialt	19500

Bemærk hvordan vi anvender `\multicolumn` til at fjerne prikkerne i den sidste række.

Vi skal i et senere afsnit se hvordan man kan anvende X til at give luft mellem søjlerne i en tabel, se afsnit 7.6 på side 186.

7.5.2 Longtable – tabel over flere sider

Til tider har man så meget data at man har behov for at kunne skrive en eller flere tabeller som fylder mere end én side. Hvad gør man så? Der er flere løsninger, vi skal her se på `longtable`.

`longtable`

En ting man lige skal huske er at floats, så som **table**, kun kan være max én side, dvs. vi kan ikke kombinere en flersidet tabel med en float. Dette er normalt ikke noget problem, man skal dog lige huske at man kan godt komme ud for at en endnu ikke placeret flydende tabel kan flyde forbi vores fler-sidede tabel, dermed få en forkert nummerering. Dette er dog forholdsvist sjældent.

Pakken `longtable` opfylder de følgende to krav til en tabelkonstruktion som fylder mere end én side:

- (1) Det skal være således at ønsker man det, skal systemet automatisk kunne gøre opmærksom på at tabellen fortsætter på næste side.
- (2) Man skal have metoder til automatisk at kunne gentage tabeloverskrifter på de efterfølgende sider.

Environmentet **longtable** tager de samme argumenter som den almindelige **tabular**, men de første rækker i tabellen kan anvendes til noget ret specielt. `Longtable` giver mulighed for at brugeren specificerer:

`longtable`

- (a) Hvad der skal stå aller først i tabellen, her placerer man normalt den almindelige caption, samt søjleoverskrifterne.
- (b) Hvad der skal når de næste sider af tabellen starter.

7. Om tabeller

- (c) Hvad der skal stå i bunden af tabellen før et sideskift, her kan man f.eks. skrive at tabellen fortsætter.
- (d) Hvad der skal stå i bunden af tabellen til aller sidst.

Hvis en af konstruktionerne som normalt kun anvendes på én side, ikke angives, anvendes den generelle i stedet.

Dette lyder ret kompliceret, men det er det i praksis ikke. [Eksempel 7.26](#) er nok for de fleste.

Eksempel 7.26

```
\begin{longtable}{\tablepreamble}}

% starten af tabellen, anvendes paa foerste side
\caption{tekst}
\label{tab:27}
\\ % caption er en speciel raeke, som skal afsluttes
\toprule
<Overskrifterne til søjlerne>
\\
\midrule
\endfirsthead % markerer slutningen af foerste header

% de resterende headers, bemaerk [] ved \caption
\caption[ ]{tekst (fortsat)}
\\ % caption er en speciel raeke, som skal afsluttes
\toprule
<Overskrifterne til søjlerne>
\\
\midrule
\endhead % slut

% den gennereelle bund foer et sideskift
\bottomrule
\multicolumn{\langle antal søjler \rangle}{r}{\textit{tabellen fortsættes}}
\endfoot % slut

% sidste footer
\bottomrule
\endlastfoot % slut

%
% Konfigurationen slutter her

% resten af raeckerne indsættes her:

\end{longtable}
```

Der er følgende bemærkninger til [Eksempel 7.26](#):

- (1) Der kan kun forekomme et sideskift *mellem* rækker, dette gælder også selvom der kan være meget tekst i en af cellerne i rækken.
- (2) **longtable** kan ikke flyde så lad være med at sætte en `\begin{table}... \end{table}` omkring den.
- (3) Selve **longtable**-tabellen er automatisk centreret så det er heller ikke nødvendigt at centrere den. Ønsker man ikke at den skal ventreres, så tag et kig i [Carlisle \(2004\)](#).
- (4) Hvis tabellen skal have en caption skal man gøre følgende

Ændre ved luft i tabeller

- (i) Selve `\caption{<tekst>}` skal placeres enten indenfor `\endfirsthead` eller `\endlastfoot`. En eventuel label placeres *sammen* med den primære caption og som sædvanligt *efter* `\caption`. I `longtable` er `\caption` en meget speciel række (den skrives jo inden i selve tabellen), så vi skal huske at afslutte denne række med `»\«`.
 - (ii) Har man f.eks. placeret den primære caption indenfor `\endfirsthead` kan man placere en sekundær så som `\caption[]{Tekst ... (fortsat)}`¹³ indenfor `\endhead`. Men det er vigtigt at man husker `[]`-erne i denne forbindelse.
- (5) På grund af den måde `longtable` er opbygget vil man ofte blive nødt til at oversætte dokumentet nogle gange før overskrifterne og bundteksten falder til ro. Dette gælder især hvis man har anvendt `\multicolumn` i en af overskrifterne. Et tegn på at man mangler at kompilere nogle gange er hvis bredderne i overskriften ikke matcher med søjlebredderne.¹⁴
 - (6) Hvis tabellen kan være på én side vil overskriften være den som er givet via `\endfirsthead` og bundmaterialet vil være det som er givet via `\endlastfoot`.
 - (7) Hvis makroerne `\endfirsthead` eller `\endlastfoot` ikke er anvendt anvendes materialet indfanget af hhv. `\endhead` og `\endfoot`.
 - (8) På grund af (1) er det en god ide at inkludere linier (rules) i de gemte 'headere' og 'footere'.
 - (9) Man kan fremtvinge et sideskift i en *longtable* via den almindelige `\newpage`, som naturligtvis skal placere mellem ro rækker.
 - (10) Man kan anvende almindelige fodnoter i *longtable*. Det kan dog se bedre ud hvis alle noterne til tabellen er samlet efter tabellen, se afsnit 7.7.1 på side 188.
 - (11) Anvender man vandrette linier til at opdele tabellen på en fornuftig måde, kan man risikere at der forekommer et sideskift omkring denne linie, hvilket ikke altid ser pænt ud. Lige pt. har vi ikke nogen fornuftig løsning på denne 'feature'.¹⁵

7.6 Ændre ved luft i tabeller

En af metoderne til at gøre en tabel mere læsbar, er at anvende meget mere luft i sin tabel, f.eks. til at adskille søjler. Dette mellemrum erstatter behovet for at adskille med lodrette linier.

Vi starter med at kigge på vertikal luft da det er det hurtigste.

Vertikal luft

Det mest almindelige problem folk kommer med vedr. manglende vertikal luft, er at linierne i tabellen kommer for tæt på indholdet. Dette har vi jo allerede løst ved at anbefale brugen af linierne fra `booktabs`-pakken i stedet (eller at man anvender *memoir*). Denne giver også mulighed for manuelt at indsætte noget luft via `\addlinespace`, se [Eksempel 7.12 på side 176](#).

Anvender man `array`-pakken har den længden `\extrarowheight` som (lokalt) kan anvendes til at lave mere luft i *toppen* af cellerne. Pas på med at ændre denne globalt,

¹³ Som kommer ud som: Tabel X: Tekst ... (fortsat).

¹⁴ Bredden af søjlerne er jo bestemt via den bredeste celle i hver søjle, så da tabellen kan deles over sider, bliver man nødt til at gemme denne data via `.aux`-filen og anvende dette ved næste overesættelse til at justere bredden på overskrifterne.

¹⁵ Fixme Dødelige: tjek lige dette

7. Om tabeller

da det også ændrer alle *array*-baserede konstruktioner, f.eks. matricer. Men indenfor f.eks. et *table*-environment før et *tabular*-environment er det helt ok.

cellspace En anden løsning er pakken *cellspace*. Man skal stort set ikke selv gøre noget, man skal bare i stedet for eksempelvis at angive »l c p{5cm}« nu desuden sætte et *S foran*, altså »Sl Sc S{p{5cm}}«. Pakken *cellspace* vil så sørge for at der er minimum $\backslash\cellspacetoplimit$ fri luft over teksten i cellen og $\backslash\cellspacebottomlimit$ fri luft under teksten i cellen. Man bør naturligvis angive *S* på alle søjlerne for at opnå det bedste resultat. De to 'grænser' er bare almindelige længder som man så kan justere på i f.eks. sin preamble.

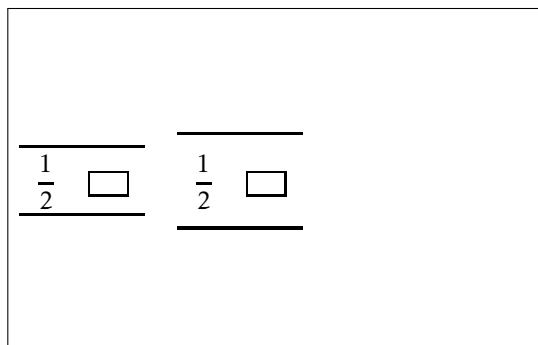
Eksempel 7.27



```
\usepackage{cellspace,amsmath,booktabs}
\addtolength\cellspacetoplimit{4pt}
\addtolength\cellspacebottomlimit{4pt}

----- slut preamble -----

\begin{tabular}{c c }
\toprule $\dfrac{1}{2}$ & $\dbx \ \ \ \backslash\bottomrule
\end{tabular}
\quad
\begin{tabular}{Sc Sc }
\toprule $\dfrac{1}{2}$ & $\dbx \ \ \ \backslash\bottomrule
\end{tabular}
```



Ønsker man at anvende den sammen med *tabularx*-konstruktionen (se senere i dette kapitel) skal man lige tage et kig i [Noirel \(2006\)](#).

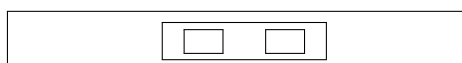
Sidebemærkning 7.13. Anvender man desuden pakken *siunitx* (hvilket anbefales), så skifter *cellspace* 'S'-et navn til 'C', for *siunitx* anvender jo selv 'S'.

Horisontal luft

Der er flere metoder til at ændre ved luften mellem søjlerne i en tabel. Først skal man lige vide at alle cellerne får en form for padding til venstre og til højre. Dette kan nemt illustreres via:

Eksempel 7.28

```
\setlength\fbboxsep{0pt}
\fbbox{\begin{tabular}{c c}\dbx & \dbx\end{tabular}}
```

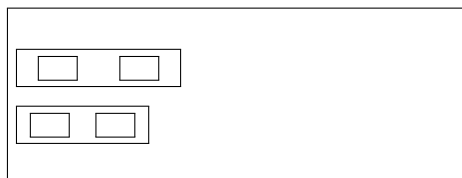


Bemærk hvordan luften mellem de to søjler er dobbelt så stor som luften ved venstre og højre kant. Størrelsen af denne luft styres af $\backslash\text{tabcolsep}$. Har man en tabel hvor man gerne lige vil trække tingene en smule mere sammen, så kan man forsøge sig med følgende (indenfor *table*)

Eksempel 7.29

```
\setlength\fbboxsep{0pt}%
\fbbox{\begin{tabular}{c c}\dbx & \dbx\end{tabular}}

\medskip\noindent%
\addtolength\text{tabcolsep}{-3pt}%
\fbbox{\begin{tabular}{c c}\dbx & \dbx\end{tabular}}
```



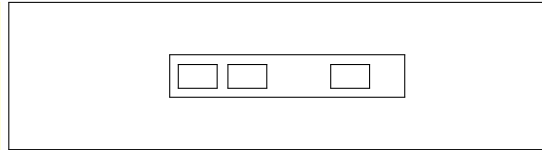
Normalt har $\backslash\text{tabcolsep}$ længden 6 pt.

Tilbage i afsnit 7.3.2 på side 169 nævnte vi at man kan lægge ting ind mellem søjlerne via $\@{\dots}$ eller $!{\dots}$ konstruktioner. Forskellen illustreres nok bedst med et eksempel¹⁶

¹⁶ Vi sætter $\backslash\text{tabcolsep}$ op for nemmere at kunne se forskellen.

Horizontal luft

```
\setlength\tabcolsep{10pt}
\setlength\fbxsep{1pt}
\fbx{\begin{tabular}{@{} l @{} l !{} l}
  \dbx & \dbx & \dbx
\end{tabular}}
```

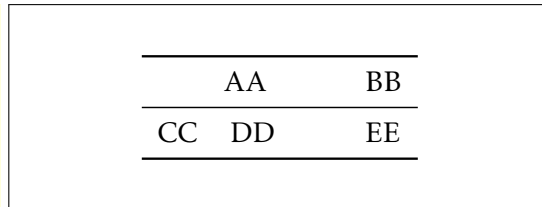


**Eksempel
7.30**

Bemærk hvordan @{} fjerner al padding til venstre i den første søjle samt hvordan @{} mellem to søjler fjerner al padding.¹⁷ Omvendt gør !{} ikke noget. Dette betyder blandt andet at hvis man gerne vil have 1 em ekstra plads mellem to søjler kan man forsøge med »!\quad«.

Omvendt er dette *ikke* den hellige gral vedr. luft mellem søjler. Ser man bort fra venstre kant af første søjler, vil @{...} og !{...} nemlig placere deres indhold i slutningen af den foregående søjle. Dette bliver et problem når vi bruger brede søjleoverskrifter:

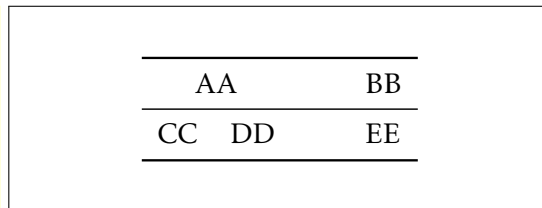
```
\begin{tabular}{l l !{\quad} l}
  \toprule
  \multicolumn{2}{c}{AA} & BB\\
  \midrule
  CC & DD & EE \\
  \bottomrule
\end{tabular}
```



**Eksempel
7.31**

Bemærk hvordan overskriften »AA« ikke længere er centreret. Det er jo fordi \multicolumn overrider søjlekonfigurationen. En løsning kunne være

```
\begin{tabular}{l l !{\quad} l}
  \toprule
  \multicolumn{2}{c!{\quad}}{AA} & BB\\
  \midrule
  CC & DD & EE \\
  \bottomrule
\end{tabular}
```



**Eksempel
7.32**

Men det er jo lidt noget bøvl at skulle gå rundt og huske på.

En noget mere fleksibel løsning får man ved direkte at anvende *luft-søjler*, altså søjler hvis eneste opgave er at styre luften mellem søjler, uden at forstyrre bredde overskrifter. Ulempen er naturligvis at man skal holde styr på flere &-tegn og specielt skal holde tungen lige i munden vedr. de brede overskrifter.

Ideen er især god sammen med *tabularx*, se følgende eksempel.

```
\begin{tabularx}{0.8\linewidth}{l !{\quad} r X r X r X r}
  \toprule
  & \multicolumn{3}{c}{Gruppe 1} & & \multicolumn{3}{c}{Gruppe 2} \\
  \cmidrule{2-4} \cmidrule{6-8}
  & Mænd & & Kvinder & & Mænd & & Kvinder \\
  \midrule
  Test 1 & 12 & & 32 & & 3 & & 5 \\
  Test 2 & 1234 & & 234 & & 334 & & --- \\
  \bottomrule
\end{tabularx}
```

**Eksempel
7.33**

Displaydelen af eksempel 7.33 findes på næste side

¹⁷ Den luft som er tilbage skyldes \dbx.

7. Om tabeller

Eksempel 7.33 fortsat

	Gruppe 1		Gruppe 2	
	Mænd	Kvinder	Mænd	Kvinder
Test 1	12	32	3	5
Test 2	1234	234	334	—

Man bliver nødt til at prøve sig lidt frem med om der skal indsættes nogle »@{ }« rundt omkring. De kan være nyttige omkring »X«, for så kan luft søjlens bredde blive mindre end $2 \times$ padding, hvilket til tider kan være nyttigt. Om man anvender @{ . . . } eller !{ . . . } til at indsætte fastbredde luft, er lidt en smagssag, det kommer an på man man ønsker at have paddingen med eller ej.

Skulle man ønske at ændre [Eksempel 7.33 på forrige side](#) således at der er fast bredde mellem søjlerne i hver af de to grupper, så kunne man f.eks. erstatte de to relevante »X«-er med »!\quad«.

Dette ender op med at give et meget mere luftigt tabeldesign som er nemmere for læseren at overskue.

7.7 Noter i tabeller

Der er mange som har det med at give forklaringer til figurer eller tabeller i captionteksten. Dette er ikke velanset, caption skal ses som en overskrift eller en form for billedtekst til en tabel eller figur. Længere forklaringer til figurer eller tabeller hører til i selve teksten. Kortere relevante noter bør dog skrives sammen med selve tabellen. I forbindelse med `longtable` (omtalt på side på side 183) kan man anvende almindelige fodnoter (via `\fodnote`), men dette kan man ikke generelt, det giver ofte heller ikke mening at anvende almindelige fodnoter da man ofte vil henvise til samme note i flere celler. Desuden er det pænere at noterne til tabellen kommer direkte efter tabellen, de skal ikke stå nederst på siden. Tabelnoter er ofte vigtigere end almindelige fodnoter (som tit bare kan ignoreres af læseren).

7.7.1 *Threeparttable(x)*

`threeparttable` Den nemmeste metode til at skrive noter til en tabel er at anvende pakken `threeparttable` som giver et environment af samme navn. Dette environment skal så pakkes omkring det tabelenvironment man ønsker at lave noter til. Syntaks:

```
\begin{threeparttable}[\langle options \rangle]
\langle almindelig tabel incl. \begin/end{tabular} etc. \rangle
\tnote
\langle Note markeringer indsættes via \tnote{\langle nummer \rangle}, se Eksempel 7.34 på modstående side \rangle
\begin{tablenotes}
\item[\langle nummer \rangle]
...
\end{tablenotes}
\end{threeparttable}
```

Makroen `\tnote` anvendes så i det tabulære materiale til at markere at der er en note til dette. `\langle nummer \rangle` er her bare det tegn eller hvad man nu ønsker at markere med, dermed er det også nemt at henvise til den samme note flere gange. Se [Eksempel 7.34 på næste side](#).

Threeparttable(x)

```
\usepackage{threeparttable}
\usepackage[tableposition=top]{caption}
----- slut preamble -----
\begin{table}
\centering
\caption{en caption}
\begin{threeparttable}
\begin{tabular}{l l}
\hline
123 & De tre første tal\tnote{a}\
321 & En anden
rækkefølge\tnote{\textdagger} \
\hline
\end{tabular}
\end{threeparttable}
\footnotesize
\begin{tablenotes}
\item[a] en note
\item[\textdagger] en
anden note
\end{tablenotes}
\end{threeparttable}
\end{table}
```

123	De tre første tal ^a
321	En anden rækkefølge [†]
^a	en note
[†]	en anden note

Eksempel
7.34



Her i bogen er *threeparttable* f.eks. anvendt ved tabel 1.1 på side 9. Man skal bemærke følgende features ved *threeparttable*:¹⁸

- Bredden af *tablenotes* (samt `\caption`, hvis den er placeret indenfor *threeparttable*) bliver den samme som selve tabellen. Ved smalle tabeller kan man omdefinere minimumsbredden af *tablenotes* (og `\caption`) via

```
\renewcommand\TPTminimum{10em}
```

Det kan være en fordel at placere `\caption` udenfor *threeparttable*, for så kan `\caption` blive bredere end tabelnoterne. Dette har dog den ulempe at `\tnote` ikke kan anvendes i `\caption`.

- Man skal selv sørge for at skrive »nummeret« til `\tnote` og til `\item` i tabelnotelisten.

Man kan ikke anvende environmentet *threeparttable* sammen med *longtable*, da dette ødelægger sideombrydningen i *longtable*. Dette kan man fikse med pakken *threeparttablex* som man kan hente på CTAN, se Madsen (2009). Denne pakke gør det også muligt at man kan referere til tabelnoter, samt endda give os et alternativ til `\tnote` så man kan markere sin note i teksten med en label. Eksempel 7.36 på næste side viser hvordan man laver tabelnoter sammen med *longtable* samt anvender muligheden for at markere via labels, derudover kan man finde eksempler i Madsen (2009).

threeparttablex

Eksempel
7.35

Tip 7.14. Bemærk hvordan vi i Eksempel 7.36 på den følgende side sætter standardfontstørrelsen for tabelnoterne globalt. Samt desuden de to ekstra makroer `\note` og `\source` som *threeparttablex* stiller til rådighed.

¹⁸ FixMe Dødelige: skal skrives om via *threeparttablex*

7. Om tabeller

Eksempel

7.36



```
\usepackage[referable]{threeparttablex}
\usepackage{longtable,booktabs}
\setTableNoteFont{\footnotesize}
\renewcommand\TPTLsourcename{Kilde}

----- slut preamble -----

\begin{ThreePartTable}
  \begin{TableNotes}
    \source daleif
    \note dette er en note
    \item[a] \label{tn:a} Note a
    \item[b] \label{tn:b} Note b
  \end{TableNotes}
  \begin{longtable}{l l l}
    \caption{A long table}\
    \toprule
    Søjle 1 & & Søjle 2 \
    \midrule
    \endhead
    \cmidrule{3-3}
    \multicolumn{3}{r}{\textit{fortsættes}}
    \endfoot
    \bottomrule
    \insertTableNotes\
    \endlastfoot
    AAAA\tnotex{tn:a} & XXXXXXXX & BBBB \
% \newpage
    CC & & DD\tnotex{tn:b} \
  \end{longtable}
\end{ThreePartTable}
```

Tabel 1: A long table

Søjle 1		Søjle 2
AAAA ^a	XXXXXXXX	BBBB
CC		DD ^b

Kilde: daleif
Note: dette er en note
^a Note a
^b Note b

Sidebemærkning 7.15 (Mine tabelnoter).

Tabelnoterne i denne bog er *ikke* skrevet med de metoder som er beskrevet her. Jeg arbejder på at lave en mere konfigurerbar måde at angive tabelnoter på. Løsningen bliver nok baseret på lister gennem enumitem-pakken. Arbejdet er under udvikling og derfor endnu ikke frigivet.

Her er en liste over features jeg gerne ville have

- Det skal være listen som laver nummereringen, altså ingen `\item[a]`.
- Henvisningen til specifikke noter skal i tabellen ske gennem specielle referencer for at mindske muligheden for indtastningsfejl.
- Klar adskillelse mellem i kildekoden mellem kilde noter, generelle noter samt specifikke noter, således at formateringen af denne data kan ske hver gruppe for sig (hvis man har lyst).
- Hvis muligt kunne det være interessant at lave en feature således at korte noter kunne stå *efter* hinanden i stedet for over.

7.8 Ekstra bemærkninger

7.8.1 Lade tabellen rage lidt ud i marginen

Skulle tabellen være lidt¹⁹ for bred til at kunne være indenfor tekstbredden kan man lade den rage lidt udenfor. Skulle man anvende *memoir*-klassen så har den en rar løsning til dette er

```
\begin{adjustwidth}{\langlejustering til venstre margin\rangle}{\langlejustering til højre margin\rangle}
\langletekst/kode\rangle
\end{adjustwidth}
```

(denne fås også via pakken *chngpage*) eller

```
\begin{adjustwidth*}{\langlejustering ved indre margin\rangle}{\langlejustering ved ydre margin\rangle}
\langletekst/kode\rangle
\end{adjustwidth*}
```

Forskellen er at *adjustwidth** sørger for at skifte justeringen alt efter om det er en højre eller en venstreside (i et to-sidet setup). Dvs. med

```
\begin{adjustwidth*}{0pt}{2em}
tabel/figur
\end{adjustwidth*}
```

Eksempel
7.37

vil tabellen stikke 2 em ud i den ydre margin. Versionen uden stjerne, har ikke dette tjek for højre- og venstresider. De to environments kan anvendes overalt ikke kun til figurer og tabeller.

7.8.2 Tabeller eller figurer i landskabsorientering (i.e. lagt ned)

Man kan sagten komme til at presse for megen information ind i en tabel²⁰ så man får brug for at kunne lægge figuren ned. Der er forskellige metoder til dette – her ser vi kort på pakken *rotating* som bl.a. giver tre interessante environments

rotating

sidewaystable

anvendes i stedet for det omkransende `\begin{table}... \end{table}` environment. Den laver en float som fylder hele siden og er smart nok til at sørge for at i *twoside*-opsætninger vil tabellen altid orienteres således at opad på tabellen vender ind mod ryggen af bogen. Bemærk at `\caption`-en vender på samme måde. Pakken har desuden options så den ikke vender tabellen på denne måde.

sidewaysfigure

som for *sidewaystable* bare for figurer.

sideways

kan anvendes til bare at vende tabellen. Anvendes den udenfor noget som flyder bør den f.eks. omkranses med *center*.

¹⁹ Hvis den er meget for bred skal man enten skrive den om, eller vælge at lægge tabellen ned, se afsnit 7.8.2.

²⁰ F.eks. alt for brede overskrifter.

7. Om tabeller

Eksempel 7.38



```
\usepackage{rotating}
|----- slut preamble -----|
\begin{sidewaystable}
  \centering
  \begin{tabular}{|l|}
    \hline
    en meget simpel tabel\\
    \hline
  \end{tabular}
  \caption{En caption.}
\end{sidewaystable}
```

en meget simpel tabel
Tabel 1: En caption.

Samt et eksempel hvor vi ikke vender caption:

Eksempel 7.39



```
\usepackage{rotating}
|----- slut preamble -----|
\begin{table}
  \centering
  \begin{sideways}
    \begin{tabular}{|l|}
      \hline
      en meget simpel tabel\\
      \hline
    \end{tabular}
  \end{sideways}
  \caption{En caption.}
\end{table}
```

en meget simpel tabel
Tabel 1: En caption.

Den opmærksomme bruger har sikkert opdaget et mindre problem her.

*Hvad gør man hvis man har en **longtable** som fylder flere sider, men er for bred til den almindelige tekstbredde?*

Problemet er at løsningerne beskrevet i [Eksempel 7.38](#) og [7.39](#) jo er floats og disse kan ikke være mere end én side.

lscap
landscape

En mulig løsning på problemet er at anvende pakken `lscap` og så pakke materialet man ønsker skal være i *landscape* ind i environmentet **landscape**. Dvs. man får en syntaks som ligner

Eksempel 7.40



```
\usepackage{lscap}
|----- slut preamble -----|
\begin{landscape}
  Noget tekst \\ den anden vej
\end{landscape}
```

Noget tekst
den anden vej

forløbet vil så være således, at når L^AT_EX når til *landscape*-environmentet, så afsluttes den nuværende side med den tekst der kommer før *landscape* (inklusive floats), hvorefter der på næste side startes en ny side hvor teksten skrives i *landscape*. Dette fortsætter indtil der ikke er mere tekst, hvorefter siden afsluttes og den resterende tekst fortsætter på siden herefter.

Bemærk at det er kun teksten indenfor tekstblokken som vendes, sidehoved og -fod ændres ikke.

Tip 7.16. Da siden afsluttes før *landscape* begynder, kan det være en god ide at få fyldt hele den nuværende side ud, inden landskabstabellen kommer. Dette gør man simplest på følgende vis:

- (a) Anvend `\usepackage{afterpage}` i preamble.
- (b) Placér koden til tabellen i en ekstern fil, f.eks. `tabel.tex`.
- (c) I teksten anvendes så

```
\afterpage{\input{table.tex}}
```

Så sikrer man sig at siden fyldes helt op før tabellen kommer.

`\afterpage` kan på denne måde anvende som en form for ‘semi-float’.

**Eksempel
7.41**

7.8.3 Brug af farver i tabeller

Generelt anbefales det *ikke* at man anvender farvede celler i tabeller, da det normalt ikke hjælper videre på forståelsen af indholdet. Dog kan det være nyttigt i forbindelse med præsentationer eller andre meget specielle applikationer. Men generelt vil vi ikke anbefale det til brug i specialer og lignende.

Vi vil henvise til pakkerne `colortbl` samt farvepakken `xcolor` som ud over at give adgang til farver, desuden også udvider funktionaliteten fra `colortbl`.

colortbl
xcolor

Sidebemærkning 7.17. Linierne vi har valgt at anbefale (linierne fra `booktabs`-pakken), fungerer slet ikke sammen med farvede celler, farven når ikke op til linierne, lidt på samme måde som med de lodrette linier i [Eksempel 7.14](#) på side 176.

Man kan dog via et trick godt farve hele baggrunden for en tabel som er i fuld bredde. I følgende eksempel anvender vi en speciel version af *shaded*-environmentet fra `framed` pakken (eller *memoir*-klassen).

```
\usepackage{xcolor,booktabs,framed,tabularx}
% version af shaded uden padding omkring
\newenvironment{shadedyellow}{%
  \def\FrameCommand{\fboxsep=0pt\colorbox{yellow}}%
  \MakeFramed {\FrameRestore}}%
  {\endMakeFramed}

----- slut preamble -----

\begin{shadedyellow}
\noindent
\begin{tabularx}{\textwidth}{X<{\enspace\dotfill} !{\quad} r}
\toprule
Udgift \let\dotfill\relax & Beløb/kr \\
\midrule
Spisning & 16000\\
Transport m.m. (8 pers) & 2500\\
Bestikkelse & 1000\\
\midrule
\multicolumn{1}{X}{Ialt} & 19500\\
\bottomrule
\end{tabularx}
\end{shadedyellow}
```

**Eksempel
7.42**



Displaydelen af eksempel 7.42 findes på næste side

7. Om tabeller

Eksempel
7.42
fortsat

Udgift	Beløb/kr
Spisning	16000
Transport m.m. (8 pers)	2500
Bestikkelse	1000
Ialt	19500

Med den rette konfiguration kan man også anvende `mdframed`-pakken til at lave baggrunden med.

Hvis tabellen *ikke* er i fuld bredde så må man ty til et lidt mere beskidt trick

Eksempel
7.43

```
\begingroup
\setlength\fbboxsep{0pt}
\colorbox{yellow}{%
Tabel kode%
}
\endgroup
```

7.8.4 Data fra eksterne kilder

Excel til L^AT_EX

Der findes en speciel makro man kan installere i Excel, således at Excel kan konvertere Excel data til L^AT_EX-kode. Excel makroen kan findes på CTAN via <http://www.ctan.org/tex-archive/support/excel2latex/> hvor man også finder en README med info om installation samt brug.

excel2latex

Gnumeric

En anden metode er at anvende programmet Gnumeric, som stort set er en Excel klon, primært til Linux, men findes også i en variant til Windows. Gnumeric har den specielle feature at den kan læse Excel-filer og kan gemme/konvertere disse til L^AT_EX, hvorfra man så kan copy'n'paste sine tabeldata ind i sit L^AT_EX-dokument.

Data fra CSV

datatool

Her kommer man ikke udenom pakken `datatool`. Den kan utroligt mange spændende ting, f.eks. lave grafer (via TiKz) ud fra data, behandle data og så bruge L^AT_EX til at formatere det med. Læs nærmere i [Talbot \(2009a\)](#).

Vi vil nøjes med at lave et eksempel hvor vi ønsker at formatere noget data fra en CSV-fil (vi snyder lidt og lader L^AT_EX selv lave CSV-filen), se [Eksempel 7.44 på modstående side](#). Det man skal bemærke i eksemplet er hvordan man hente data ind og formatere dem uden så mange dikkedarer, og hvordan man kan arbejde mere med data og dermed selv f.eks. styre hvordan søjlerne skal håndteres. Faktisk kunne man i sidste del af eksemplet sagtens ignorere søjler man ikke havde brug for.



```

\begin{filecontents*}{data.csv}
Forkortelse,Navn,Hjemmeside
DK-TUG,"Dansk TeX-brugergruppe",http://www.tug.dk
TUG,"The TeX Users Group",http://www.tug.org
DANTE,"Deutschsprachige Anwendervereinigung TeX e.V",http://www.dante.de
\end{filecontents*}
\usepackage{datatool,url,booktabs}

----- slut preamble -----

\DTLloaddb{data}{data.csv}
\renewcommand{\dtldisplaystarttab}{\toprule}
\renewcommand{\dtldisplayafterhead}{\midrule}
\renewcommand{\dtldisplayendtab}{\bottomrule}
\DTLdisplaydb{data}

Mere kontrol over formateringen:

\begin{tabular}{lll}
\toprule
\textbf{Forkortelse} & & 
\textbf{Navn} & & 
\textbf{Hjemmeside} \\
\DTLforeach{data}{%
\forkortelse=Forkortelse,\navn=Navn,\hjemmeside=Hjemmeside}{%
\DTLiffirstrow{\midrule}{\}%
\forkortelse & \navn & \expandafter\url\expandafter{\hjemmeside}
}
\bottomrule
\end{tabular}

```

Forkortelse	Navn	Hjemmeside
DK-TUG	Dansk TeX-brugergruppe	http://www.tug.dk
TUG	The TeX Users Group	http://www.tug.org
DANTE	Deutschsprachige Anwendervereinigung TeX e.V	http://www.dante.de

Mere kontrol over formateringen:

Forkortelse	Navn	Hjemmeside
DK-TUG	Dansk TeX-brugergruppe	http://www.tug.dk
TUG	The TeX Users Group	http://www.tug.org
DANTE	Deutschsprachige Anwendervereinigung TeX e.V	http://www.dante.de

Nu er de to eksempler i [Eksempel 7.44](#) nok ikke så relevante i virkeligheden. Det aller smarteste er at kombinere pakkerne `datatool` og `siunitx`, så lad os lave et eksempel med dem begge, for man skal lige være en smule vågen. Følgende bygger på omtalen af `datatool` i et eksempel i [Wright \(2010\)](#).

Vi skal ind og bruge `\DTLforeach`, men denne kan ikke bruges i en af `siunitx`'s S-søjler. Derfor skal vi bruge et trick, da alt hvad `\DTLforeach` laver er globalt²¹ vil vi i stedet lave en søjle som ikke bruges til andet en at lave for-løkken (fabrikerede data).

[Eksempel 7.45 på den følgende side](#) er en smule teknisk, så lad os forklare et par ting:

- (a) Vi loader csv-filen med nogle ekstra options, `noheader` for at fortælle at den ikke

²¹ Ellers kan det ikke bruges i en tabel.

7. Om tabeller

Eksempel 7.45



```
\begin{filecontents*}{data2.csv}
Dag 1,1.23444,23.344e5
Dag 2,10.23555,4.567e7
\end{filecontents*}
\usepackage{datatool,booktabs}
\usepackage[load-configurations=abbreviations]{siunitx}

----- slut preamble -----

\DTLloaddb[noheader,keys={h,a,b}]{data2}{data2.csv}

\begin{tabular}{l}
S[table-format=2.3,table-auto-round]
S[table-format=2.2e1,table-auto-round]
@{} l}
\toprule
& { Forbrug (\si{kWh}) } & { Energi (\si{MeV}) } &
\DTLforeach{data2}{%
\Ih=h,\Ia=a,\Ib=b}{%

\DTLiffirstrow{ \ \ \midrule }{ \ \ }%

\Ih & \Ia & \Ib &
}
\ \
\bottomrule
\end{tabular}
```

	Forbrug (kWh)	Energi (MeV)
Dag 1	1.234	23.34×10^5
Dag 2	10.236	4.57×10^7

skal lede efter den, og en key-liste som er interne navne (headere om man vil) som vi så kan henvise til senere.

- Vi starter og formaterer en tabel som vanligt, der er 3 søjler i outputtet men bemærk »@{} l« til slut, det er dette som er tricket, den tomme søjle. Vi smider @{} foran for at den ikke skal optage nogen plads.
- Så laves overskrifterne, der er tre en tom og to tekster.
- I den femte overskrift er det så vi udnytter den tomme søjle.
 - Her looper vi over rækkerne i csv-filen, vi tildeler henvisningsmakroer via de headernavne vi valgte tidligere.
 - Indeni selve loopet starter vi med at markere at vi vil begynde en ny række.
 - Herefter skriver vi selve indholdet med de makroer vi definerede i deklara-tionen af loopet.

I en senere version af bogen vil vi inkludere et eksempel hvor vi kobler to databaser af samme størrelse, f.eks. så man kan tage søjle 1 og 3 fra database A og søjle 1, 2 og 5 fra database B.²²

²² Fixme Note: husk at få dette med

Konstruktion af litteraturliste

Til tider også kendt som *bibliografi*, hvilket er noget *alle* får brug for på et eller andet tidspunkt. Kapitlet her fortæller om hvordan man citerer i L^AT_EX (meget lig det at lave en krydsreference) og hvordan man sætter citeringsmaterialet op. Vi vil både se på hvordan man skriver litteraturlisten pr. håndkraft samt hvordan man automatiserer processen via det eksterne program B_IB_TE_X.

8.1 Citeringskonventioner

Indenfor naturvidenskab (hovedmålgruppen for denne bog) anvendes normalt to forskellige citeringskonventioner.

- *talsystem*, dvs. når man citerer en artikel kommer der f.eks. til at stå [3], eller [3,5–7]
- *forfatter-dato*, dette kunne være »Mittelbach og Goossens (2004)«, altså et (eller flere) navn(e) samt et årstal.

Til tider anvendes endda en kombination af begge systemer.

Inden for jura og humaniora anvendes nogle lidt mere avancerede systemer. Her vil det være en ide at tage et kig på jurabib-pakken. Mittelbach og Goossens (2004) har et længere afsnit vedrørende dette emne. Vi vil i afsnittet »*Eksempler fra jurabib pakken*« på side 214 vise nogle eksempler fra jurabib, venligst stillet til rådighed af Mogens Larsen. Jurabib kan f.eks. håndtere at skrive citeringen som fodnoter, hvor teksten fra litteraturlisten så gentages i fodnoten. Denne type er ikke så meget brugt indenfor naturvidenskab, men kan f.eks. være interessant indenfor Matematik-Økonomi.

Nogen bruger også at skrive citeringerne som fodnoter, igen er det ikke videre normalt indenfor matematik, men kunne måske ses indenfor Matematik-Økonomi eller f.eks. videnskabshistorie. Dette kan enten jurabib eller pakken opcit (se Garcia (2006)) klare.

Vi vil kigge på de to systemer som normalt anvendes inden for naturvidenskab. Først ser vi på hvordan man kan lave bibliografien pr. håndkraft, hvilket kan anvendes til en kort artikel eller et bachelorprojekt, man bliver dog hurtigt træt af det. Herefter vil vi automatisere processen via B_IB_TE_X og den mangfoldighed af muligheder dette program bringer med sig.

8.2 Bibliografienvironmentet

Environmentet *thebibliography* anvendes helt generelt til at lave *alle* bibliografierne med. Anvendes B_IB_TE_X så vil outputtet fra dette program være et udfyldt *thebibliography*-environment som så automatisk indlæses af L^AT_EX.

8.2.1 Citerings syntaks

Syntaksen for *thebibliography*-environmentet er:

thebibliography

8. Konstruktion af litteraturliste

```
\begin{thebibliography}{\langle bredeste 'labeltekst' \rangle}
\bibitem[\langle txt1 \rangle]{\langle nøgle1 \rangle} \langle bibliografisk tekst \rangle
\bibitem[\langle txt2 \rangle]{\langle nøgle2 \rangle} \langle bibliografisk tekst \rangle
...
\end{thebibliography}
```

Man citerer meget nemt en artikel via

```
\cite[\langle tekst efter citering \rangle]{\langle nøgle \rangle}
```

ligesom ved krydsreferencer og man skal igen huske at kompilere nogle gange før citeringerne er helt opdateret. Det valgfri argument demonstreres i følgende eksempel.

Eksempel 8.1



```
\cite{t1c2}, \cite[kapitel~8]{memman}
```

```
\begin{thebibliography}{2}
\bibitem{t1c2}
Frank Mittelbach og Michel Goossens.
\emph{The \LaTeX{} Companion}.
Addison-Wesley, 2. udgave, 2004.
```

```
\bibitem{memman}
Peter Wilson.
\emph{The Memoir Class
for Configurable Typesetting}.
The Herries Press,
6. udgave, 2004.
\end{thebibliography}
```

[1], [2, kapitel 8]

Litteratur

[1] Frank Mittelbach og Michel Goossens. *The L^AT_EX Companion*. Addison-Wesley, 2. udgave, 2004.

[2] Peter Wilson. *The Memoir Class for Configurable Typesetting*. The Herries Press, 6. udgave, 2004.

Som det ses af eksemplet er standardopsætningen at man får nummersystemet. Man kan, i stedet for nummeret, skrive en brugerdefineret tekst. Her kommer det valgfri argument til `\bibitem` ind, se [Eksempel 8.2](#). Bemærk desuden argumentet til *thebibliography*-environmentet.

Eksempel 8.2

```
\cite{t1c2}, \cite[kapitel~8]{memman}
```

```
\begin{thebibliography}{MG04}
\bibitem[MG04]{t1c2}
Frank Mittelbach og Michel Goossens.
\emph{The \LaTeX{} Companion}.
```

```
\bibitem[PW04]{memman}
Peter Wilson.
\emph{The Memoir Class
for Configurable Typesetting}.
\end{thebibliography}
```

[MG04], [PW04, kapitel 8]

Litteratur

[MG04] Frank Mittelbach og Michel Goossens. *The L^AT_EX Companion*.

[PW04] Peter Wilson. *The Memoir Class for Configurable Typesetting*.

bredeste 'labeltekst'

[Eksempel 8.2](#) viser også hvad `\langle bredeste 'labeltekst' \rangle`-argumentet til *thebibliography* skulle bruges til. Den sørger for at venstre margin af den bibliografiske tekst passer. Så man skal her enten vælge den bredeste tekststreng eller det bredeste tal som er anvendt yderst i listen, sammenlign med [Eksempel 8.1](#).

Sidebemærkning 8.1. Der er følgende bemærkninger:

- (a) Når man laver bibliografien på denne måde er det op til en selv at sørge for at formatere tingene på den måde som man nu en gang ønsker / er blevet påkrævet. F.eks. ser man ofte at titler står med kursiv. Journal bind (nummer) skrives ofte i fed.
- (b) En fejl jeg ofte retter i preprints og lignende er når forfatteren har skrevet bibliografien pr. håndkraft og *glemmer* at intervaller *skal*¹ laves via -- og ikke via -, husk derfor at tjekke især sidetals angivelser.
- (c) Sammentrækninger i stil med [4,6–9,11] kan laves via pakken cite, se [Arseneau \(2003a\)](#) for yderligere information om brugen af denne pakke.

cite

<pre> \usepackage{cite} ----- slut preamble ----- \cite{a,b,c,e} \cite{c,a,b,d} \cite{a,c,d,e} \begin{thebibliography}{9} \bibitem{a} A \bibitem{b} B \bibitem{c} C \bibitem{d} D \bibitem{e} E \end{thebibliography} </pre>	<pre> [1–3, 5] [1–4] [1, 3–5] Litteratur [1] A [2] B [3] C [4] D [5] E </pre>
--	--

Eksempel 8.3


Pakken natbib som vi skal kigge på senere, har også mulighed for automatisk at kunne lave denne sammentrækning, så der behøver man ikke pakken cite.

8.2.2 Forfatter-år citeringer lavet uden BibTeX

Det er ikke videre fleksibelt at lave forfatter-år citeringer direkte via *thebibliography* (uden at skrive alt for meget i hånden) men det kan faktisk godt lade sig gøre, selv om man anbefales kraftigt at anvende BibTeX i stedet hvilket vi skal se senere. Den interesserede læser kan tage et kig på harvard-pakken og dens tilhørende udvidede syntaks.

harvard

8.3 BibTeX

8.3.1 Hvad er BibTeX?

BibTeX er et eksternt program som ud fra data angivet i LaTeX-dokumentet (gennem .aux-filerne), en eller flere bibliografiske databasefiler samt en BibTeX-konfigurationsfil kan lave en bibliografi formateret ud fra forskellige regler (specificeret af konfigurationsfilen), f.eks. fremsat af en journal. Et BibTeX forløb kan se nogenlunde sådan ud:

Trin 1 Forfatteren citerer en bog som endnu ikke tidligere har været citeret i dette dokument. Denne information tilføjes ved oversættelsen af dokumentet til .aux-filen.

¹ Ud fra et typografisk synspunkt.

8. Konstruktion af litteraturliste

Trin 2 Man kører B_IT_EX på hoveddokumentets .aux-fil (eller bare på filnavnet (uden endelse) på masterfilen), f.eks.:

```
bibtex master
```

B_IT_EX laver et specielt formateret *thebibliography*-environment i en ekstern fil (master.bbl) som kommandoen `\bibliography` (se nedenfor) så sørger for at inkludere i dokumentet.

Trin 3 Man oversætter igen hoveddokumentet, den nyligt genererede .bbl-fil inkluderes automatisk og *thebibliography* sørger for at skrive de rette citeringsinformationer (det er de informationer som `\cite` skal bruge for at skrive citeringsteksten) til .aux-filen, her i blandt informationerne til det nye værk vi lige har citeret.

Trin 4 Man oversætter igen. Nu er citeringsoplysningerne, som skal indsættes i teksten af `\cite` (og venner), tilgængelige og bliver indsat i teksten.

Man skal altså udføre (i rækkefølge):

```
latex, bibtex, latex og latex
```

hver gang man har citeret noget som *ikke* har været citeret før. Har materialet tidligere være citeret, findes oplysningerne allerede (i .aux-filerne) og man har ikke brug for at køre B_IT_EX igen.

For at kunne anvende B_IT_EX skal man placere følgende to makroer i sit dokument:

```
\bibliographystyle{\bibliographystyle{\langle bibstyle \rangle}
\bibliography{\bibliography{\langle liste af bib-filnavne \rangle}}
```

`\langle bibstyle \rangle` er navnet på konfigureringsfilen B_IT_EX skal anvende (angivet uden filendelse) og denne afgør hvordan bibliografien skal konfigureres, ting forkortes osv. Se afsnit 8.5 på side 207 for nogle eksempler.

`\langle liste af bib-filnavne \rangle` er en kommasepareret liste af navne (uden endelse) på B_IT_EX databasefiler, som alle *skal* have endelsen » .bib«. Vi skal senere se hvordan man laver en database fil, afsnit 8.4 på side 204.

`\bibliographystyle` kan sagtens placeres i preambelen, mens `\bibliography` placeres der hvor man gerne vil have ens litteraturliste placeret. Her er et simpelt eksempel (skrevet med en mindre font for at spare lidt plads).

Eksempel 8.4



```
\cite{texbytopic},
\cite[kapitel-5]{bringhurst}
\bibliographystyle{plain}
\bibliography{minbib}
```

[2], [1, kapitel 5]

Litteratur

- [1] Robert Bringhurst. *The Elements of Typographic Style*. Hartley & Marks, 2. edition, 2002. Version 2.5.
- [2] Victor Eijkhout. *T_EX by Topic—A T_EXnician's Reference*. Addison-Wesley, 2001.

Alt efter stilen vil B_IT_EX selv sørge for at skrive bibliografien i alfabetisk rækkefølge (efter første forfatter) eller forkorte navnene osv. Man kan også nemt få B_IT_EX til at indsætte punkterne i bibliografien i den rækkefølge de er citeret, hvilket de f.eks. *ikke* er i [Eksempel 8.4](#).

Hvad er BibTeX?

Sidebemærkning 8.2. I modsætning til direkte brug af *thebibliography*-environmentet vil man ved brug af BibTeX *kun* få de bøger med i litteraturlisten, som man faktisk citerer. Skulle man ønske at inkludere værker (i litteraturlisten) som ikke citeres direkte kan dette gøres via

```
\nocite{⟨(kommasepareret) liste af nøgleord⟩}
```

\nocite

eller

```
\nocite{*}
```

som inkluderer *alle* indgange i alle de .bib-filer man har inkluderet, med i litteraturlisten.

Modsat af hvad man skulle tro, er det ikke nogen ulempe, at man normalt kun får de bøger med man faktisk citerer. Denne funktionalitet betyder at man kan genbruge ens .bib-filer i mange forskellige projekter, hvilket ikke er så nemt med den håndformaterede litteraturliste. Man kan så at sige sagtens have *alle* de artikler og bøger man har læst, liggende i sin(e) BibTeX database(r) og så gøre brug af dem igen og igen. Da BibTeX databaserne bare er informationer, er det eneste som skal til for at skifte fra layoutet for »Journal A« til layoutet for »Journal B«, at man skifter bibliografistilen (konfigurationsfilen).

Tip 8.3. Skulle en journal kræve at man inkluderer et *thebibliography*-environment (det kan jo være de ikke selv kan finde ud af at anvende BibTeX), skal man bare erstatte `\bibliography{...}` med indholdet af den genererede .bbl-fil (f.eks. master.bbl).

En anden fordel er naturligvis at man nemt kan skifte litteraturlistens udseende – man skifter jo bare bibliografi stilen og oversætter det hele forfra.²

```
\cite{bib1}
\nocite{bib2}
\bibliographystyle{alpha}
\bibliography{minbib}
```

[Eft05]

Litteratur

[Eft99] Fornavn2 Efternavn2. Titel2. *Journal2*, 35:233–245, 1999.

[Eft05] Fornavn1 Efternavn1. Titel1. *Journal1*, 2005.

Eksempel 8.5



Eksempel 8.5 viser en stil hvor forfatterefternavn(e) samt året giver citeringsteksten. Det følgende eksempel viser en citerings sorteret version.

```
\cite{texbytopic}
\cite{bringhurst}
\bibliographystyle{unsrt}
\bibliography{minbib}
```

[1] [2]

Litteratur

[1] Victor Eijkhout. *TeX by Topic—A TeXnician's Reference*. Addison-Wesley, 2001.

[2] Robert Bringhurst. *The Elements of Typographic Style*. Hartley & Marks, 2. edition, 2002. Version 2.5.

Eksempel 8.6



I de eksempler vi har set indtil nu har vi hele tiden anvendt talsystemet som citerings-system, lad os nu i stedet se på forfatter-år systemet.

² latex, bibtex, latex og latex.

8. Konstruktion af litteraturliste

8.3.2 Forfatter-år citeringer med pakken natbib

Forfatter-år citeringer kommer normalt i to varianter »Forfatternavn (år)« samt »(forfatter, år)« og ofte har det mening (alt efter hvordan sætningen er formuleret) at kunne anvende begge versioner. Til dette anvender vi oftest natbib-pakken. Her har man nu ikke kun `\cite` men anvender i stedet forskellige andre varianter:

```
\citet      \citet[⟨tekst før citering⟩][⟨tekst efter citering⟩]{⟨nøgle liste⟩}
\citep      \citep[⟨tekst før citering⟩][⟨tekst efter citering⟩]{⟨nøgle liste⟩}
```

`\cite` vil nu (ofte) være et alias for `\citet`. Hvis man kun ønsker en *⟨tekst efter citering⟩* (som ved `\cite`) anvender man bare kun et enkelt valgfrit argument, men ønsker man kun en *⟨tekst før citering⟩* skal man anvende begge valgfri argumenter og lade det højre være blankt, se [Eksempel 8.7](#).

Eksempel 8.7



```
\usepackage{natbib}
\bibliographystyle{chicago}
----- slut preamble -----
\citet{perl}           \\\
\citet[kapitel-8]{perl} \\\
\citet[se][kapitel-8]{perl}\\\
\citet[se][{}]{perl}   \\\[7pt]
\citep{perl}           \\\
\citep[kapitel-8]{perl} \\\
\citep[se][kapitel-8]{perl}\\\
\citep[se][{}]{perl}
```

```
Wall et al. (2000)
Wall et al. (2000, kapitel 8)
Wall et al. (se 2000, kapitel 8)
Wall et al. (se 2000)

(Wall et al., 2000)
(Wall et al., 2000, kapitel 8)
(se Wall et al., 2000, kapitel 8)
(se Wall et al., 2000)
```

»Et al.«-delen betyder at der her er en bog med mere end to forfattere (kommer an på bibliografistilen). Anvend `\citet*` og `\citep*` for at få de fulde navne.

`\citet*`
`\citep*`

Eksempel 8.8



```
\usepackage{natbib}
\bibliographystyle{chicago}
----- slut preamble -----
\citet*{perl}          \\\
\citep*[se][kapitel-8]{perl}\\\
```

```
Wall, Christiansen, and Orwant (2000)
(se Wall, Christiansen, and Orwant, 2000, kapitel 8)
```

Sidebemærkning 8.4. Som man kan se ovenfor vil man alt efter BibTeX-konfigurationen få »et. al«-linier, hvis forfatter antallet kommer over en vis grænse (bestemt af stilen). Der er desværre et potentielt problem med dette. Når BibTeX laver de data som f.eks. bliver til »Wall et. al« som ovenfor, så kender den ikke til nogen af de andre indgange i litteraturlisten. Tager vi stilen chicago som eksempel, så giver den »et. al« ved tre forfattere og op. Betragt så følgende to publikationer, udgivet samme år med forfatterlisten F_1, F_2, F_3 og F_1, F_2', F_3' , da vil begge give » F_1 et. al (2007)« selv om det er to forskellige forfatterkombinationer!

Eneste måde at undgå dette er ved først at tjekke ens bibliografi for at se om der er potentielle problemer og så sørge for *altid* at citere disse via fulde navne, i.e. `\cite*{⟨key⟩}`.

Man har desuden følgende varianter:

```
\citealt      \citealt{⟨nøgle⟩} som \citet men uden parenteser (plus *-version)
\citealp      \citealp{⟨nøgle⟩} som \citep men uden parenteser (plus *-version)
\citeauthor   \citeauthor{⟨nøgle⟩} giver forfatterdelen, evt. forkortet
\citeauthor*  \citeauthor*{⟨nøgle⟩} giver forfatterdelen i fuld version
```

Pakkeoptions til natbib

`\citeyear{⟨nøgle⟩}` året uden parenteser og
`\citeyearpar{⟨nøgle⟩}` med parenteser

`\citeyear`
`\citeyearpar`

Via de beskrevne makroer kan man så lave sine egne specielle citerings udtryk. De anvendte parenteser (som er styret af natbib og eventuelt bibliografistilen) kan specificeres via

`\citetext{⟨tekst⟩}`

`\citetext`

Eksempel:

```
\usepackage[square]{natbib}
% nu med en anden type parenteser
\bibliographystyle{chicago}
----- slut preamble -----
\citetext{se \citealt{texbytopic} og
om perl se \citealp*{perl}}
```

[se Eijkhout 2001 og om perl se Wall, Christiansen, and Orwant, 2000]

Eksempel 8.9



Man kan sagtens komme ud for at lave en citering i begyndelsen af en sætning – men her kan der så være et problem hvis forfatterens efternavn faktisk starter med små bogstaver – så som *van* eller *von* etc. Men det kan natbib også klare, se [Daly \(2003b\)](#) afsnit 4.4.

En bog kan også være mere kendt under en specifik betegnelse, f.eks. var den gamle udgave af »*LaTeX Companion*« kendt som »*Hundebogen*«, da den havde en hund på forsiden. Dette kan man lave via

```
\defcitealias{⟨nøgle⟩}{⟨anden tekst⟩}
\citetalias{⟨nøgle⟩}
\citepalias{⟨nøgle⟩}
```

`\defcitealias`
`\citetalias`
`\citepalias`

Eksempelvis

```
\usepackage{natbib}
\bibliographystyle{chicago}
----- slut preamble -----
\defcitealias{tlc2}{TLC2}
Vi kan nu citere \citet{tlc2} eller som
\citetalias{tlc2}
```

Vi kan nu citere Mittelbach and Goossens (2004) eller som TLC2

Eksempel 8.10



Sidebemærkning 8.5. Det skal lige bemærkes at har man under forfatter-år konfigurationen en `\cite` med to citeringer af samme forfatter, så vil denne blive forenklet, se f.eks. [Daly \(2003b\)](#), side 7, nederst på siden. Det kunne se ud som i [Eksempel 8.11](#).

```
\cite{SASdisplay,MemoirChapStyles}
```

Madsen [2006b,a]

Eksempel 8.11

Pakkeoptions til natbib

En større oversigt kan ses i [Daly \(2003b\)](#), men her er et par stykker:

- (i) `round`, `square` styrer hvilke typer parenteser der skal anvendes.
- (ii) `colon` (standard, navnet er misvisende, der anvendes semikolon) eller `comma` styrer separationen mellem citeringer når man citerer flere på en gang. Natbib version 8 (fra 2007), har ændret dette, således at `semicolon` og `colon` nu betyder det samme.

8. Konstruktion af litteraturliste

- (iii) `authoryear` (default), `numbers` eller `super` ændrer måden hvorpå man citerer (kan også styres fra stilen). `numbers` giver numre i litteraturlisten, men stadigvæk forfatter informationer i citeringer i teksten, mens `super` skriver citeringen som et løftet tal (som i journalen *Nature*).

Det skal bemærkes at man ved `numbers` får ændret fortolkning af `\cite`, den bliver nemlig nu lig `\citep`.

Eksempel 8.12



<pre>\usepackage[numbers]{natbib} \bibliographystyle{chicago} ----- slut preamble ----- \cite{perl} \ \citet{perl}\ \citep{perl}</pre>	<pre>[1] Wall et al. [1] [1]</pre>
--	------------------------------------

Det vil sige at `\citet` faktisk her giver os både forfatternavnene samt en numerisk reference. Det er der nogen som kan lide.

Skulle man have fundet ovenstående interessant nok til at man hellere vil anvende `\citet` i stedet for `\cite`, så kan dette nemt fikses: Placer følgende efter man har aktiveret `natbib`.

Eksempel 8.13

```
\let\cite\citet
```

- (iv) `sort` ændrer udseendet ved en citering af flere værker på en gang, `sort` vil sortere dem så de kommer i samme rækkefølge som i litteraturlisten.

Anvendes de numeriske citeringer vil `sort&compress` betyde at man får sammentrykket citeringen, i.e. [3–9,12], svarer altså til virkningen af pakken `cite`.

- (v) `longnamesfirst` betyder at første gang et værk er citeret vil den lange udgave af forfatternavnene blive anvendt mens de efterfølgende citeringer bliver i en eventuelt forkortet »et al.«-version.

8.4 Generering af BIB_TE_X-database

Der findes forskellige værktøjer til at håndtere genereringen af bibliografidatabaserne, se [Mittelbach og Goossens \(2004\)](#) afsnit 13.4, men for de flestes vedkommende kan man sagtens lave en sådan via en almindelig teksteditor.

Beskrivelse af format

En BIB_TE_X databasefil er bare en ganske simpel tekstfil, hvor hver indgang er formateret på en speciel måde således `bibtex` kan forstå det.

I selve formatet består hver værk-indgang af konstruktioner af følgende type:

```
@<type>{<nøgle>,
  felt_navn_1 = {<tekst 1>},
  felt_navn_2 = {<tekst 2>},
  felt_navn_3 = {<tekst 3>},
  ...
  felt_navn_n = {<tekst n>}
}
```

I stedet for at omkrænse `<tekst>` med `{ }`-er man kan også anvende et `" "`-par. Der gælder visse regler man skal overholde når man skriver database indgangene

Beskrivelse af format

- man *skal* afslutte alle felter med komma (»,«) (behøver dog ikke efter det sidste punkt).
- hvert fysisk punkt i filen *skal* have en entydig *<nøgle>* hvilket er den tekst man citerer efter med citeringsmakroerne (bemærk placeringen af nøglen, lige efter den åbne {). Nøglen bør *ikke* indeholde mellemrum, danske tegn eller komma.
- man må *ikke* anvende non-ASCII tegn i bibliografien, dvs. dem skal man i stedet skrive med L^AT_EX koder, æ=\ae, ø=\o og å=\aa samt diverse andre accent tegn og disse *skal* alle omkranses via { }-er, i.e. J{\o}rgen, det samme gælder f.eks. \ 'e.³
Som jeg ser det er det dog kun af hensyn til sorteringen at man skal tænke på dette. I »note« feltet (se nedenfor) kan man sagtens bruge æøå, det plejer ikke være noget problem.
Man bør også placere { }-omkring andre L^AT_EX makroer, f.eks. \TeX, thi nogle bibliografistile har det med at ændre visse store bogstaver til små.
- Al luft omkring lighedstegnet (=) ignoreres af B_IB_TE_X.

Alt efter den type indgangen har, vil visse felter være obligatoriske,⁴ nogle vil være valgfri (dvs. kommer med hvis de er tilstede), mens andre helt ignoreres. Hvilke som fortolkes og hvilke som ikke gør, kommer helt an på bibliografistilen. Nogle stile understøtter f.eks. et issn-felt, andre måske et url-felt.

Her er nogle af de mest almindelige typer og deres forhold til feltnavne (fra [Mittelbach og Goossens \(2004\)](#) side 763 hvor man også kan finde flere⁵).

article *artikel i journal eller tidsskrift*

Obligatorisk: author, title, journal og year.

Valgfri: volume, number, pages, month og note.

book *bog med udgiver*

Obligatorisk: author eller editor⁶ title, publisher og year.

Valgfri: volume eller number, series, address, edition, month og note.

booklet *trykt værk som mangler udgiver*

Obligatorisk: title

Valgfri: author, howpublished, address, year og note.

inbook *del af en bog*

Obligatorisk: author eller editor, title, chapter og/eller pages, publisher og year.

Valgfri: volume eller number, series, type, address, edition, month og note.

phdthesis *ph.d.-afhandling*

Obligatorisk: author, title, school og year.

Valgfri: type, address, month og note.

proceedings *fra en konference*

Obligatorisk: title og year

Valgfri: editor, volume eller number, series, address, publisher, note, month og organization.

³ Det kan være at bib_TE_X ikke kan finde ud af at sortere ordentligt på dansk, i så fald kan man prøve programmet bib_TE_X8 i stedet. Der er dog det forbehold at bib_TE_X8 forventer at finde tegnene æ, ø eller å, og *ikke* \ae, \o eller \aa som bib_TE_X forventer. Tak til Arne Jørgensen for den afklaring.

⁴ B_IB_TE_X giver en advarsel hvis et påkrævet felt ikke er med.

⁵ Vi mangler her misc, masterthesis, techreport, unpublished og manual.

⁶ Det kan se ud som om man enten skal anvende den ene eller den anden, men ikke begge. Det kan dog sagtens afhænge af stilen.

8. Konstruktion af litteraturliste

inproceedings *en artikel i en konferenceproceeding (bog)*

Obligatorisk: author, title, booktitle og year.

Valgfri: *samme som ved proceedings.*

På den måde kan man altså få:

Eksempel 8.14

```
@Book{texbytopic,  
  author = {Eijkhout, Victor},  
  title = {{{\TeX} by Topic---A {\TeX}nician's Reference},  
  publisher = {Addison-Wesley},  
  year = {2001},  
  note = {Er skrevet i 1991 men nu udsolgt fra forlaget.  
         Forfatteren har senere overtaget alle  
         copyright rettigheder og har stillet bogen  
         frit til download via  
         \url{http://www.eijkhout.net/tbt/}. },  
}
```

Syntaks for author-feltet

Et navn i author feltet indtastes normalt på en af to måder:

```
Fornavn Mellempnavn(e) Efternavn  
Efternavn, Fornavn Mellempnavn(e)
```

I de fleste tilfælde kan BibTeX sagtens finde ud af hvilke dele af navnet der er hvad. Den kan endda identificere »von« navne. Men hvor der kan opstå tvivl om hvordan navnene skal fortolkes (nogen har to ligeværdige efternavne), foretrækkes den sidste formulering.

Sidebemærkning 8.6. Bemærk at der kan være visse grænsetilfælde som kan volde problemer, i dette tilfælde henvises til [Mittelbach og Goossens \(2004\)](#), afsnit 13.2.2.

Sidebemærkning 8.7. Når man skriver forfatternavnene i sin .bib-fil, er det en god ide at være konsekvent i den måde man skriver tingene på, så er man sikker på at BibTeX formaterer det samme forfatternavn på samme måde hver gang.

and Når man har mere end en forfatter adskilles de med ordet »and« (også på dansk, ordet »and« er en del af BibTeX formatet). Her er et eksempel fra [Rose og Moore \(1999\)](#).

Eksempel 8.15

```
author = {Rose, Kristoffer H. and Moore, Ross},
```

Bemærk for øvrigt at en eventuel forfattersortering (af selve litteraturlisten) gælder efter den første forfatter. Se også afsnit [8.6 på side 216](#).

Specielt om titelfeltet

Man skriver naturligvis titlen på værket, som det står på forsiden af værket. Men der kan være specielle sproglige regler, evt. indbygget i en bibliografistil, som gør at titlen kommer til at se anderledes ud. Det man oftest ser, er at ord som i .bib-filen stod med stort nu er skrevet med lille (dog ikke det første ord i titlen). Hvis dette er et problem, eksempelvis hvis der i titlen indgår navnet på en by, skal man tage sine forholdsregler. Man sætter det berørte ord ind i et {}-par:

URL i bibliografier

Brug af `\LaTeX` ved `{Aarhus}` `{Universitet}`

Her er `\LaTeX` også omkranset, idet man kunne risikere at få `\latex` i outputtet, og den er ikke defineret som standard.

Sidebemærkning 8.8. Man anbefales at omkranske hele ordet og ikke bare første bogstav i ordet (Mittelbach og Goossens, 2004, side 768). Grunden kan ses her:

`{V}a` vs. `{Va}` → `Va` vs. `Va`

Fænomenet kaldes *kerning*, eller *knibning* på dansk. Vi kan se at »a«'et kommer (korrekt) tættere på »V«'et når man anvender `{Va}`. Desværre anbefaler mange journalers info til formattere stadigvæk at man anvender »`{V}a`«, men det er altså forkert. Husk i den forbindelse at gøre det samme omkring matematik i titlen.

URL i bibliografier

Det er de færreste bibliografistile som understøtter et `url`-felt. Er man så uheldig at den stil man anvender ikke understøtter et `-`felt, anbefales det at man placerer `URL`-en i `note`-feltet.

Tip 8.9. Når man skal inkludere en `URL` er det fornuftigt at anvende pakken `url` og dens `\url` makro. Denne makro vil nemlig kun dele `URL`-en på de steder hvor det er tilladt at dele en `url`. Det går jo ikke at man orddele ord i en `URL`, i stedet ombrydes ved »:«, ».« og »/«.

`url`

`\url`

8.5 Bibliografistilarter

7

Der findes mange forskellige konfigurationsfiler (bibliographystyles) til `BIBTEX`. Mittelbach og Goossens (2004) har en længere oversigt, vi vil nævne nogle stykker i afsnit 8.5.1. Bibliografistilfiler har altid endelsen `.bst`, skulle man have lyst til at lede efter dem.

8.5.1 Eksempler på `BIBTEX` konfigureringsfiler

I alle eksemplerne vil vi anvende følgende `BIBTEX`-data (gemt som `testbib.bib`):⁸

```
@incollection{Ahr02,
  author = "Ahr, D. and Reinelt, G.",
  title = "New heuristics and lower bounds for the min-max  $k$ -Chinese postman problem",
  editor = "Möhring, R. and Raman, R.",
  series = "Lecture Notes in Computer Science",
  booktitle = "Lecture Notes in Computer Science",
  volume = 2460,
  pages = "64--74",
  publisher = "Springer Verlag",
  address = "Berlin",
  year = 2002,
```

⁷ Fixme Note: det vil være naturligt at få blandet `babelbib` ind i det også

⁸ Data venligst stillet til rådighed af Christian Roed Pedersen.

8. Konstruktion af litteraturliste

```
}
@article{Dror00b,
  author = "Dror, M. and Haouari, M.",
  title = "Generalized {S}teiner problems and other variants",
  journal = "Journal of Combinatorial Optimization",
  year = 2000,
  volume = 4,
  pages = "415--436",
  file = "dro00b.pdf",
  annote = {Denne note har intet budskab, men viser blot dens funktion.},
}
@book{Lawler76,
  author = "Lawler, E.L.",
  title = "Combinatorial Optimization: Networks and Matroids",
  publisher = "Holt, Rinehart and Winston",
  address = "New York",
  year = 1976,
  shorttitle = {Combinatorial Optimization},
}
```

Og den dokumentkildekode vi vil gøre brug af er:⁹

Eksempel 8.16

```
\renewcommand\section[2]{}
\noindent
\cite{Dror00b}, \cite{Ahr02}, \cite{Lawler76}
\bibliography{testbib}
```

Man kan desuden tage et kig på <http://www.cs.stir.ac.uk/~kjt/software/latex/showbst.html> hvor man finder en oversigt over nogle B_WT_EX-stile.

I eksemplet er der tilføjet to ekstra nøgleord: *annote* og *shorttitle*, disse ignoreres af de fleste B_WT_EX stile, men kan anvendes i forbindelse med jurabib, se afsnittet »*Eksempler fra jurabib pakken*« på side 214.

Standardstile

Eksempel 8.17

```
\bibliographystyle{plain}
```

[2], [1], [3]

- [1] D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.
- [2] M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.
- [3] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

⁹ Ved at omdefinere \section slipper vi for overskriften og sparer derfor en del plads.

`\bibliographystyle{alpha}`

Eksempel
8.18

[DH00], [AR02], [Law76]

[AR02] D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

[DH00] M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

[Law76] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.



`\bibliographystyle{unsrt}`

Eksempel
8.19

[1], [2], [3]

[1] M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

[2] D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

[3] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.



`\bibliographystyle{abbrv}`

Eksempel
8.20

[2], [1], [3]

[1] D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

[2] M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

[3] E. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.



Denne stil er ikke så god sammen med vores data idet data allerede er skrevet med forkortede fornavne. Men ideen bag stilen er altså at den automatisk vil sørge for at gøre fornavne til initialer.

Danske oversættelser

Arne Jørgensen har (via Dag Langmyhrs norske oversættelser) oversat ovenstående fem BibTeX-konfigurationer til dansk, således at man får danske ord i stedet for de almindelige engelske. Desuden er der kommet support for et par ekstra BibTeX-felter. For yderligere information se [Jørgensen \(2006\)](#).

`\bibliographystyle{dk-plain}`

Displaydelen af eksempel 8.21 findes på næste side

Eksempel
8.21



8. Konstruktion af litteraturliste

Eksempel 8.21 fortsat

- [2], [1], [3]
- [1] D. Ahr og G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. I R. Möhring og R. Raman, redaktører, *Lecture Notes in Computer Science*, bind 2460 af *Lecture Notes in Computer Science*, side 64-74. Springer Verlag, Berlin, 2002.
- [2] M. Dror og M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415-436, 2000.
- [3] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Eksempel 8.22

```
\bibliographystyle{dk-alpha}
```

- [DH00], [AR02], [Law76]
- [AR02] D. Ahr og G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. I R. Möhring og R. Raman, redaktører, *Lecture Notes in Computer Science*, bind 2460 af *Lecture Notes in Computer Science*, side 64-74. Springer Verlag, Berlin, 2002.
- [DH00] M. Dror og M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415-436, 2000.
- [Law76] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Eksempel 8.23

```
\bibliographystyle{dk-unsrt}
```

- [1], [2], [3]
- [1] M. Dror og M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415-436, 2000.
- [2] D. Ahr og G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. I R. Möhring og R. Raman, redaktører, *Lecture Notes in Computer Science*, bind 2460 af *Lecture Notes in Computer Science*, side 64-74. Springer Verlag, Berlin, 2002.
- [3] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Eksempel 8.24

```
\bibliographystyle{dk-abbrev}
```

- [2], [1], [3]
- [1] D. Ahr og G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. I R. Möhring og R. Raman, redaktører, *Lecture Notes in Computer Science*, bind 2460 af *Lecture Notes in Computer Science*, side 64-74. Springer Verlag, Berlin, 2002.
- [2] M. Dror og M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415-436, 2000.
- [3] E. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Eksempel 8.25

```
\usepackage{apalike}{dk-bib}  
\bibliographystyle{dk-apali}
```

Displaydelen af eksempel 8.25 findes på næste side

BibTeX-stile kompatibel med natbib-pakken

(Dror og Haouari, 2000), (Ahr og Reinelt, 2002), (Lawler, 1976)

Ahr, D. og Reinelt, G. (2002). New heuristics and lower bounds for the min-max k -Chinese postman problem. I Möhring, R. og Raman, R., redaktører, *Lecture Notes in Computer Science*, bind 2460 af *Lecture Notes in Computer Science*, side 64-74. Springer Verlag, Berlin.

Dror, M. og Haouari, M. (2000). Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415-436.

Lawler, E. (1976). *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York.

Eksempel
8.25
fortsat

Grunden til at vi bliver nødt til at indlæse pakken dk-bib er at apalike-stile kræver eksistensen af visse makroer.

BibTeX-stile kompatibel med natbib-pakken

Bemærk igen at der generelt vil blive anvendt engelske ord i de formaterede indgange.

```
\usepackage{natbib}
\bibliographystyle{plainnat}
```

Eksempel
8.26


Dror and Haouari [2000], Ahr and Reinelt [2002], Lawler [1976]

D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

For lige at demonstrere natbib-pakkens stil-options

```
\usepackage[numbers]{natbib}
\bibliographystyle{plainnat}
```

Eksempel
8.27


[2], [1], [3]

[1] D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

[2] M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

[3] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Her viser vi også referencerne i teksten (lidt ændret fra de andre eksempler)

```
\usepackage[super]{natbib}
\bibliographystyle{plainnat}
```

Eksempel
8.28


Displaydelen af eksempel 8.28 findes på næste side

8. Konstruktion af litteraturliste

Eksempel 8.28 fortsat

A^2, B^1, C^3

- [1] D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.
- [2] M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.
- [3] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Eksempel 8.29

```
\usepackage{natbib}
\bibliographystyle{unsrnat}
```

Dror and Haouari [2000], Ahr and Reinelt [2002], Lawler [1976]

M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Eksempel 8.30

```
\usepackage{natbib}
\bibliographystyle{abbrvnat}
```

Dror and Haouari [2000], Ahr and Reinelt [2002], Lawler [1976]

D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

E. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Eksempel 8.31

```
\usepackage{natbib}
\bibliographystyle{chicago}
```

Dror and Haouari (2000), Ahr and Reinelt (2002), Lawler (1976)

Ahr, D. and G. Reinelt (2002). New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman (Eds.), *Lecture Notes in Computer Science*, Volume 2460 of *Lecture Notes in Computer Science*, pp. 64–74. Berlin: Springer Verlag.

Dror, M. and M. Haouari (2000). Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization* 4, 415–436.

Lawler, E. (1976). *Combinatorial Optimization: Networks and Matroids*. New York: Holt, Rinehart and Winston.

BibTeX-stile kompatible med natbib-pakken

Man skal dog bemærke at `chicago` ikke er fundt kompatibel med `natbib` i alle henseende.

```
\usepackage{natbib}
\bibliographystyle{apalike}
```

Eksempel
8.32



Dror and Haouari (2000), Ahr and Reinelt (2002), Lawler (1976)

Ahr, D. and Reinelt, G. (2002). New heuristics and lower bounds for the min-max k -Chinese postman problem. In Möhring, R. and Raman, R., editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin.

Dror, M. and Haouari, M. (2000). Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436.

Lawler, E. (1976). *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York.

Se også `dk-apali`, som desværre (endnu) ikke er kompatibel med `natbib`.

```
\usepackage{natbib}
\bibliographystyle{dfltxbbibtex}
```

Eksempel
8.33



Dror og Haouari (2000), Ahr og Reinelt (2002), Lawler (1976)

D. Ahr og G. Reinelt, *New heuristics and lower bounds for the min-max k -Chinese postman problem*, i R. Möhring og R. Raman, red., *Lecture Notes in Computer Science*, bind 2460 af *Lecture Notes in Computer Science*, pp. 64–74, Springer Verlag, Berlin, 2002.

M. Dror og M. Haouari, *Generalized Steiner problems and other variants*. *Journal of Combinatorial Optimization*, 4, 415–436, 2000.

E.L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.

Dette er en stil som jeg personligt lavede til denne bog via `makebst`. Stilen findes sammen med `dfltxb` bundtet på CTAN, hvor man også finder `makebst` konfigurationsfilen. Ønsker man at lege med konfigurationsfilen, bedes man omdøbe den først.

```
\usepackage{natbib}
\bibliographystyle{kluwer}
```

Eksempel
8.34



Dror and Haouari (2000), Ahr and Reinelt (2002), Lawler (1976)

Ahr, D. and Reinelt, G.: 2002, New heuristics and lower bounds for the min-max k -Chinese postman problem, in R. Möhring and R. Raman (eds), *Lecture Notes in Computer Science*, Vol. 2460 of *Lecture Notes in Computer Science*, Springer Verlag, Berlin, pp. 64–74.

Dror, M. and Haouari, M.: 2000, Generalized Steiner problems and other variants, *Journal of Combinatorial Optimization* 4, 415–436.

Lawler, E.: 1976, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York.

Denne kræver desuden at `incollection` har et udfyldt `booktitle`-felt.

8. Konstruktion af litteraturliste

Eksempler fra jurabib pakken

Vi vil ikke gå i detaljer¹⁰ men vil i stedet henvise til pakkedokumentationen Berger (2004), samt Mittelbach og Goossens (2004) afsnit 12.5.1, side 715–743.

Det meste af den følgende tekst samt alle eksempler er udarbejdet og venligst stillet til rådighed af Mogens Larsen.

Først et eksempel på en almindelig opsætning (anvendt af Mogens til hans speciale i historie). Som man ser kommer citeringen frem i en fodnote på siden hvor der citeres.

Eksempel 8.35



```
\usepackage[%
authorformat=firstnotreversed, % Ved flere forfattere er kun den
% første på formen ``Efternavn, Fornavn''.
titleformat=all, % Næste citation er ``efternavn, titel, side''.
% Titel vil være shorttitle i bibtex fil hvis angivet, ellers alm. titel.
commabeforerest, % Kommaer før sidenumre og andre tilføjelser.
ibidem=strict, % Brug af ibid ved gentagne referencer til samme titel.
% Idem kan også benyttes. Begge kan justeres.
citefull=first, % Fuld citation ved første reference.
oxford, % Oxford stil (humaniora).
super, % Ændrer \cite til \footcite.
pages=format, % Tilføjer ``p.'' og ``pp.''
annote, % Tilføjer indholdet af bibtex feltet ``annote'',
% som kan være en note til titlen.
]{jurabib}

----- slut preamble -----

\A\cite{Dror00b}, B\cite{Lawler76}, C\cite{Ahr02} og D.\cite[430]{Lawler76}
\bibliographystyle{jox} \renewcommand\section[2]{} % fusk for at spare plads
\bibliography{testbib}
```

A¹, B², C³ og D.⁴

Ahr, D. og G. Reinelt, 'New heuristics and lower bounds for the min-max k -Chinese postman problem', i: Möhring, R. og R. Raman, redaktører, *Lecture Notes in Computer Science*, volume 2460 (Berlin: Springer Verlag, 2002), pp. 64–74.

Dror, M. og M. Haouari, 'Generalized Steiner problems and other variants', *Journal of Combinatorial Optimization* 4 (2000), pp. 415–436.

Denne note har intet budskab, men viser blot dens funktion.

Lawler, E.L., *Combinatorial Optimization: Networks and Matroids* (New York: Holt, Rinehart and Winston, 1976).

¹M. Dror og M. Haouari, 'Generalized Steiner problems and other variants', *Journal of Combinatorial Optimization* 4 (2000).

²E.L. Lawler, *Combinatorial Optimization: Networks and Matroids* (New York: Holt, Rinehart and Winston, 1976).

³D. Ahr og G. Reinelt, 'New heuristics and lower bounds for the min-max k -Chinese postman problem', i: R. Möhring og R. Raman, redaktører, *Lecture Notes in Computer Science*, volume 2460 (Berlin: Springer Verlag, 2002).

⁴Lawler, *Combinatorial Optimization*, p. 430.

Det skal lige bemærkes at der kan være en fejl i filen `dajbbib. ldf` som styrer den danske opsætning vedr. `jurabib`. Definitionen af `\editorsname`, bør være

```
\def\editorsname{\unskip, \space{}redakt\o rer}%
```

Her er et modificeret eksempel fra `jurabib`s hjemmeside.

¹⁰ Det er der slet ikke plads til, hvis man skal dække bare lidt af det `jurabib` kan.

Andet

```
\usepackage[%
citefull=first,           % Fuld citation ved første reference.
authorformat={smallcaps,year}, % Forfatternavne i small caps og
                             % tilføjet år ved citation.
titleformat=colonsep,     % Kolon mellem forfatter og titel
                             % ved citation.
titleformat=italic,       % Bogtitler i italic.
commabeforerest,         % Komma før sidenummer .
ibidem=strict,           % Brug af Ibid for gentagne referencer
                             % i sekvens til samme bog/artikel. .
bibformat=ibidemalt,     % Alternativt bibliografi formatide fra tysk jura.
dotafter=bibentry,       % Slutpunktum.
]{jurabib}
```

----- slut preamble -----

```
A\footcite{Dror00b}, B\footcite{Lawler76},
C\footcite{Ahr02} og D.\footcite[430]{Lawler76}
\bibliographystyle{jurabib}\renewcommand\section[2]{} % fusk
\bibliography{testbib}
```

Eksempel

8.36



A¹, B², C³ og D.⁴

Ahr, D./Reinelt, G.:

- New heuristics and lower bounds for the min-max k -Chinese postman problem. I Möhring, R./Raman, R., redaktører: Lecture Notes in Computer Science. Volume 2460, Berlin: Springer Verlag, 2002, 64–74.

Dror, M./Haouari, M.:

- Generalized Steiner problems and other variants. Journal of Combinatorial Optimization, 4 2000, 415–436.

¹DROR, M./HAOUARI, M.: Generalized Steiner problems and other variants. Journal of Combinatorial Optimization, 4 2000.

²LAWLER, E.L.: *Combinatorial Optimization: Networks and Matroids*. New York: Holt, Rinehart and Winston, 1976.

³AHR, D./REINELT, G.: New heuristics and lower bounds for the min-max k -Chinese postman problem. I MÖHRING, R./RAMAN, R., redaktører: Lecture Notes in Computer Science. Volume 2460, Berlin: Springer Verlag, 2002.

⁴LAWLER (1976), 430.

Som det ses kan man konfigurere jurabib nærmest i det uendelige. Man skal naturligvis huske at anvende en BibTeX-stil som understøtter jurabib, men så er man ellers kørende. Faktisk kan man også justere den til at få et udseende som vi tidligere har et med natbib.

8.6 Andet

Sortering af danske navne

Det kan give problemer hvis man har en forfatter som f.eks. hedder »Sørensen«, dette skal jo i BibTeX skrives som »S{\o}rensen«, men så bliver det sorteret som »So« i stedet for som »Sø«. Dette kan faktisk fikses med et lille trick. Tilføj først følgende i starten af .bib-filen

```
@PREAMBLE{{ \providecommand{\noopsort}[1]{} }}
```

Dette tilføjer en kommando til bibliografien som bare æder sit input, men med den kan man forklare BibTeX hvordan man ønsker at sortere et bestemt ord. Det skal så f.eks. anvendes som

Eksempel

8.37



8. Konstruktion af litteraturliste

Eksempel 8.38

```
author = {S{\noopsort{z}{\o}}rensen, Bent},
```

Så vil »Sørensen« blive sorteret som »Szrensen«, hvilket kan afhjælpe en forkert sortering. Et andet eksempel kunne være hvis man ønsker at citere »Lord Kelvin« og hvor »Lord« skal placeres foran efternavnet.

Eksempel 8.39

```
author = {{\noopsort{Kelvin}{Lord Kelvin}} },
```

Sprogproblemer ved brug af B_IB_TE_X

På dansk vil vi gerne afskille forfatternavnene i bibliografien med ordet »og« ikke med ordet »and«. Man *skal* anvende »and« til at adskille forfatternavnene i .bib-filen, det er op til bibliografistilen at bestemme hvilket ord der skal adskille forfatternavnene. Vi har lige set at Arne Jørgensen har sørget for at standardstilene samt apalike er oversat til dansk, men der er desværre ingen af disse som er kompatible med natbib.

Man har nogle muligheder:

- babelbib* (1) Prøv om pakken babelbib er noget for dig. Denne pakke giver, sammen med oversættelser af de mest almindelige stilfiler, en feature hvor specialord i litteraturlisten, kan styres gennem babel eller via brugeren. Se [Harders \(2009\)](#) for mere information. De tilhørende B_IB_TE_X-stilfiler understøtter ud over de mere normale felter desuden, ISSN, ISBN, URL og URLDATE.
- Så vidt vides er babelbib er kompatibel med natbib.
- (2) Ignorer problemet og accepter at der står visse engelske betegnelser i bibliografien. Hvis ens vejleder er emsig (eller man selv er det) kan man også rette bibliografien pr. håndkraft. Gøres ved at man *efter* man sidste gang har kørt B_IB_TE_X, inkluderer indholdet af .bbl-filen ind i ens dokument i stedet for `\bibliography{<style>}` (grunden til at man ikke bør rette det direkte i .bbl-filen er at denne overskrives hvis man ved et uheld kommer til at køre B_IB_TE_X igen).
- makebst* (3) Lav din egen bibliografistil med makebst, se [Mittelbach og Goossens \(2004\)](#) for mere information. Bibliografistilen anvendt i denne bog (daleif.bst) er hjemmelavet via makebst. (I begrænset omfang kan undertegnede godt være behjælpelig med at lave en stil via makebst, især hvis vi kan tage udgangspunkt i stilen anvendt til denne bog (daleif.bst). Ønsker man en kopi af daleif.bst, eller makebst-konfigurationsfilen daleif.dbj, er man velkommen til at sende mig en mail.)

Muligvis kan biblatex også hjælpe med sprogproblemer, se afsnit 8.9 på side 222.

Et andet sprogproblem blev beskrevet i fodnote 3 på side 205, nemlig at B_IB_TE_X i sig selv ikke kan finde ud af at sortere mht. de specielle danske vokaler. En løsning på dette er programmet bibtex8. Bemærk at Arne Jørgensen har i forbindelse med hans dk-bib-pakke også lavet nogle specielle konfigurations filer til bibtex8, så den desuden kan finde ud af dobbelt a.

En helt tredje mulighed er at forsøge sig med jurabib som jo fint har understøttelse for dansk. Se eksemplerne fra afsnittet »[Eksempler fra jurabib pakken](#)«, på side 214.

Online reference databaser

B_IB_TE_X kan ikke selv hente data fra nettet, men der findes faktisk databaser på nettet, hvor man kan hente bibliografiske information i blandt andet i B_IB_TE_X-format. Desværre er adgangen til disse databaser sjældent gratis. Den gode nyhed er dog at de fleste

Emneopdelt litteraturliste

universiteter har fuld adgang til disse databaser og man kan så tilgå informationerne fra universiteternes egne netværk. Har man tilknytning til Aarhus Universitet har man desuden mulighed for at anvende Statsbibliotekets fjernadgang til at tilgå databaserne hjemmefra, se

<http://imf.au.dk/bibliotek/adgang/>

Ellers skal man spørge til mulighederne hos ens lokale institutbibliotek.

Her er en liste over nogle enkelte databaser.

- MathSciNet (<http://ams.mathematik.uni-bielefeld.de/mathscinet>)
Stor database over diverse artikler og bøger indenfor de fleste områder af matematikken. Dog kan matematik-økonomi halte lidt. Man finder den artikel man søger, sætter mærke ved det søgte, går til toppen af siden og finder »Citations (BibTeX)« i dropdown-menuen og trykker »Retrieve Marked«. Så får man en fin BibTeX-indgang for netop denne artikel. Her er et eksempel (hvis man har adgang):

<http://ams.math.uni-bielefeld.de/mathscinet/search/publications.html?fmt=bibtex&b=2219303>.

- Zentralblatt MATH (<http://www.zentralblatt-math.org/zmath/en/>)
Ligner meget MathSciNet og er endnu nemmere at anvende.
- Der er sikkert også databaser indenfor andre discipliner som faktisk tilbyder citeringsdata i BibTeX-format. Hvis de kun tilbyder i andre formater skulle man evt. kunne konvertere denne data via programmet jabref, se afsnit 8.8 på side 221, eller via RefWorks, se afsnit 8.8 på side 221.

8.7 Emneopdelt litteraturliste

Til tider kunne man godt tænke sig at emneopdele litteraturlisten i f.eks. artikler og bøger, eller i primær og sekundær litteratur. Dette kan vi naturligvis også, men det kræver lidt ekstra arbejde.

De pakker som normalt nævnes er: `multibib`, `bibtopic` og `bibunits` (samt `chapter-bib` hvis man ønsker en litteraturliste til hvert kapitel). Fælles for alle tre er at de genererer ekstra »`.aux`«-filer som man *manuelt* skal køre BibTeX på. Dette kan være et problem for de som ikke er vant til at anvende et terminalvindue (Kommando-prompt eller Xterm). Jeg ved ikke om der er videre mange LaTeX-specifikke editorer som giver een mulighed for at køre BibTeX på en anden fil end hoveddokumentet, bare ved tryk på en knap.

Vi vil kigge lidt på de to første pakker. Fordele og ulemper kan ses i Tabel 8.1 på næste side.

Brug af pakken `multibib`

Hele ideen bag `multibib` er at man til hvert emne nu i stedet genererer citeringsmakroer specielt til dette emne. Dette har den fordel at man direkte i kildekoden kan se hvad citeringen henviser til.

multibib

8. Konstruktion af litteraturliste

Tabel 8.1: Fordele og ulemper ved pakkerne *multibib* og *bibtopic*.

	multibib	bibtopic
Fordele	Kræver kun en enkelt ».bib«-fil. Nemt at se i kildekoden hvad der hører til hvor.	Kan nemt tilføjes et eksisterende dokument. Kræver blot at man laver nye ».bib«-filer. Spændende muligheder for at kunne opdele i citerede og ikke citerede værker indenfor hvert emne.
Ulemper	Besværligt at tilføje et eksisterende dokument, da citeringer skal skrives om.	Hvert emne skal i hver sin ».bib«-fil. Hvilket er besværligt hvis man har mange indgange. Har man en artikel af samme forfatter, i samme år, min inden for to forskellige emner, så vil man <i>ikke</i> automatisk få tilføjet bogstaver til årstallet, idet behandlingen af de to emner ikke kender noget til hinanden.

Sidebemærkning 8.10. En ulempe ved alle konstruktionerne til at lave emneopdelte litteraturlister er dog at hvis, man anvender forfatter-år systemet og har to forskellige citeringer af samme forfatter i samme år men i forskellige emner, så vil der *ikke* automatisk blive sat et bogstav på årstallet. Dette skyldes naturligvis at hver litteraturliste behandles hver for sig og kender ikke til hinandens indhold. Heldigvis er det de færreste som løber ind i dette problem.

Man starter med at definere sine emner via

```
\newcites      \newcites{<liste af prefixer>}margliste af overskrifter
```

Et eksempel på dette kunne være:

Eksempel 8.40 `\newcite{sec}{{Sekundær litteratur}}`

Dette laver så nogle nye makroer

```
\citesec{<key>}
\nocitesec{<keys>}
\bibliographystylesec{<stil>}
\bibliographysec{<bib-fil>}
```

Som man skal anvende i stedet for `\cite` eller `\nocite` for at fylde materiale i den sekundære litteraturliste. Den primære litteraturliste kan så samtidigt fyldes via `\cite` og `\nocite`. Lad os se et eksempel.

Kort om bibtopic

Eksempel 8.41



```
\usepackage{natbib}
\usepackage{multibib}
\newcites{sec}{Sekundær litteratur}
```

— slut preamble —

```
bla bla jf. \cite{Dror00b} og \cite{Ahr02}.
Alternativt kan man se \citesec{Lawler76}.
\bibliographystyle{abbrvnat}
\bibliography{testbib}
\bibliographystylesec{abbrvnat}
\bibliographysec{testbib}
```

bla bla jf. Dror and Haouari [2000] og Ahr and Reinelt [2002]. Alternativt kan man se Lawler [1976].

Litteratur

D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

Sekundær litteratur

E. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

Sidebemærkning 8.11. Der er den lille hage at de genererede udgaver af `\bibliographystyle`, her `\bibliographystylesec`, skal anvendes i selve teksten. Anvendes de i preambelen giver det en B_WT_EX-fejl på hoveddokumentet.

Med pakken `multibib` er proceduren altså:

```
latex master
bibtex master
bibtex X  ∀ X ∈ {⟨liste af prefixer⟩}
latex master
latex master
```

For yderligere information, se [Hansen \(2004\)](#).

Kort om bibtopic

Pakken `bibtopic` går helt en anden vej end `multibib`. Her har man i stedet: (a) En specifik `«.bib«`-fil for hvert emne. (b) Hver `«.bib«`-fil skal indlæses gennem et specielt environment hver for sig. (c) Hvert environment laver en ny fil (`master1`, `master2`,...) hvorpå man kan køre B_WT_EX. (d) I hvert environment skal man selv specificere hvad man ønsker at få trykt:

bibtopic

- (i) De ting fra denne `«.bib«`-fil som faktisk er citeret.

8. Konstruktion af litteraturliste

- (ii) De som ikke er (disse to kan gøres separat fra hinanden således man kan sætte en overskrift i mellem).
- (iii) Eller, printe hele indholdet af ».bib«-filen.

For detaljer henvises man til Ulrich (2006). Men her er et eksempel.

Eksempel 8.42



```
\usepackage{bibtopic}
----- slut preamble -----
\cite{Ahr02}, \cite{Lawler76} og noget som er alternativt \cite{texbytopic}.
\begin{btSect}[alpha]{testbib}
\section{Citerede værker}
\btPrintCited
\section{Ikke citerede værker}
\btPrintNotCited
\end{btSect}
\begin{btSect}[plain]{minbib}
\section{Alternativt}
\btPrintCited
\end{btSect}
```

[AR02], [Law76] og noget som er alternativt [1].

1 Citerede værker

[AR02] D. Ahr and G. Reinelt. New heuristics and lower bounds for the min-max k -Chinese postman problem. In R. Möhring and R. Raman, editors, *Lecture Notes in Computer Science*, volume 2460 of *Lecture Notes in Computer Science*, pages 64–74. Springer Verlag, Berlin, 2002.

[Law76] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

2 Ikke citerede værker

[DH00] M. Dror and M. Haouari. Generalized Steiner problems and other variants. *Journal of Combinatorial Optimization*, 4:415–436, 2000.

3 Alternativt

[1] Victor Eijkhout. *T_EX by Topic—A T_EXnician’s Reference*. Addison-Wesley, 2001.

Makroerne af interesse er:

```
\btPrintCited
\btPrintNotCited
\btPrintAll
\begin{btSect}[\langle bib stil \rangle]{\langle bib-fil \rangle}
\langle indhold som i Eksempel 8.42 \rangle
\end{btSect}
```

Forklaringer findes i Ulrich (2006).

Har man brug for f.eks. at tage alle artikler ud fra sin ».bib«-fil (og filen er lidt for lang til at man kan overskue det i sin editor) så kan programmet jabref anbefales, se afsnit 8.8 på næste side.

8.8 Håndtering af referencer databaser

Normalt har man oftest ikke så vildt mange indgange i sin BIBTEX -databasefil, så man kan sagtens styre denne med en almindelig tekst editor. Men er man i den situation at man har rigtigt mange indgange eller måske har brug for at konvertere fra et andet bibliografi format til BIBTEX -formatet, så får man brug for lidt andre programmer.

Jabref

Det program de fleste anbefaler er jabref, som er et java program som derfor kører på alle platforme bare man har Java installeret. JabRef kan downloades (gratis) fra <http://jabref.sourceforge.net/>. Personligt har jeg ikke den store erfaring med programmet, men de gange jeg har testet det, har mit indtryk været at det er meget nemt og intuitivt at anvende. Desuden kan programmet importere fra andre forskellige citeringssystemer.

jabref

RefWorks

RefWorks er et (kommercielt) webbaseret, platformsuafhængigt produkt til at håndtere referencer og lignende. Selve værktøjet består af en konto hos RefWorks og her importerer man så sine artikler, bøger etc. som man finder i forskellige databaser. Produktet indeholder også forskellige features som så kan anvendes til citeringer i Word.

Det vi, som $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -brugere er mest interesseret i er:

- (i) Man kan importere bibliografisk data fra alverdens forskellige databaser, specielt behøver det ikke være matematik relaterede databaser.
- (ii) De fleste ikke matematiske databaser giver naturligvis ikke publikations data ud i BIBTEX -format, men ofte i f.eks. EndNote format eller lignende. RefWorks har intet problem med at importere dette.
- (iii) Ingen problemer i at importere eksisterende BIBTEX -filer, eller copy'n'paste BIBTEX -kode direkte ind i RefWorks.

Og så *piece de resistance*:

RefWorks kan eksportere til BIBTEX -format!

Det vil sige, er man $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -bruger og arbejder men i et fag hvor der traditionelt ikke tilbydes bibliografisk data i et format $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ kan håndtere, så kan RefWorks anvendes til at konvertere denne data til BIBTEX -format, således man kan anvende data i $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

RefWorks er som sagt et kommercielt produkt, men mange højere danske læreanstalter tilbyder deres studerende adgang til RefWorks, det gør AU for eksempel (gennem statsbiblioteket). Spørg hos dit lokale bibliotek om der er adgang til RefWorks. Man skal så registrere fra en computer på institutionens netværk, herefter sker adgangen bare via brugernavn + password samt et group-id (en ekstra kode) og kan ske fra en hvilken som helst computer med en internet forbindelse og en browser.

Her er et link til Statsbibliotekets informationer vedr. RefWorks

http://www.statsbiblioteket.dk/publ/RefWorks_In_general.html

8.9 Biblatex

Fordelen ved brugen af BibTeX er selve opdelingen af data i databasefilen, man kan stort set tilføje al den data man kan forestille sig, det er dog op til stilene at gøre brug af denne data. Ulempen ved brugen af BibTeX er at det er meget svært at ændre noget. Hvis outputtet fra BibTeX -stilen ikke helt passer med den konvention man ønsker, så er det svært at lave om på.¹¹ Det er ikke noget man bare lige retter ved at tilføje noget i preamblen.

biblatex Denne problematik er der mange som har kigget på, bl.a. Philipp Lehman som i 2006 påbegyndte arbejdet med biblatex, som er en pakke, konfigurationsfiler samt en enkelt konfigureringsstil til BibTeX .¹² Ideen bag biblatex er at udnytte styrkerne ved BibTeX (dvs. formatet) og kombinere dette med styrkerne ved LaTeX (makroerne). På denne måde har Lehman skabt et system som kombinerer funktionaliteten af en masse pakker til LaTeX , samt en masse BibTeX -stile, men nu samlet i en enkelt enhed som kan konfigureres i let tilgængelige konfigurationsfiler eller via preambelen. BibTeX anvendes nu bare til at sortere data, samt hive data ud af databasefilen (data omformes til LaTeX -makroer).

Pr. februar 2010 udkom biblatex version 0.9. Forfatteren mener nu at interfacet er stabilt nok til at folk kan bruge det. Pakken er derfor blevet indlemmet i opdateringerne til MiKTeX 2.8 og TeX Live 2009 og senere. Senere er deres afløser til BibTeX programmet – Biber nu også kommet til og i i hvertfald inkluderet i TeX Live .

I en senere version af bogen vil vi begrænse behandlingen af den generelle bibtex en del og i stedet fokusere mere på biblatex for der er ikke nogen tvivl om at dette er fremtiden for citeringer i LaTeX .

¹¹ Sproget stilene er skrevet i, har man slet ikke lyst til at arbejde med. Se evt. [Markey \(2009\)](#).

¹² Man arbejder på en variant af bibtex programmet, specielt til biblatex, kaldet biber. Den skulle være brugbar allerede nu, men er pt. meget svær at få installeret på Windows platformen.

Mere om tekst

I dette kapitel går vi tilbage og kigger på forskellige tekstrelaterede emner, lige fra overskrifter, over skrifttyper og lister.

9.1 Lidt mere om afsnitoverskrifter samt indholdsfortegnelsen

I afsnit 2.1 på side 23 fortalte vi lidt om makroer til afsnitoverskrifter samt til starten af kapitler. Samtidigt fortalte afsnit 2.6 på side 36 om indholdsfortegnelsen. Vi vil lige komme med et par kommentarer.

Det er ofte en fordel at man i alle overskrifter undgår orddeling så vidt det er muligt. Nogle klasser gør dette som standard (f.eks. *memoir*) eller kan nemt konfigureres til det, mens andre ikke gør (f.eks. *article*). I det sidste tilfælde kan man få hjælp via `sectsty` eller `titlesec`.

På andre tidspunkter vil man gerne have delt en overskrift efter et bestemt ord. Dette er meget nemt via (man kan også anvende `»\«` i stedet for `»\newline«`)

```
\section{Tekst\newline mere tekst}
```

Eksempel
9.1

Ulempen ved dette er hvis man desuden også har en indholdsfortegnelse. Al teksten i de ustjernede makroer `\section`, `\chapter` etc. bliver også anvendt til TOCen (samt evt. anvendt til et sidehoved). Hvor det jo naturligvis bliver grimt hvis der pludselig er et lineskift. Løsningen på dette er allerede indbygget i overskriftkommandoerne, de har alle et valgfrit argument som så bliver skrevet til sidehovedet samt TOCen, i stedet for det obligatoriske argument, så vores eksempel fra før bliver:

```
\section[Tekst mere tekst]{Tekst\newline mere tekst}
```

Eksempel
9.2

Mere formelt gælder for alle makroerne nævnt i Tabel 2.1 på side 23 at den fulde syntaks er

```
\section[⟨til indholdsfortegnelse⟩]{⟨Titel⟩}
```

Memoir. I *memoir*-klassen har man endda to valgfri argumenter, som så styrer hhv. teksten til sidehovedet samt teksten til indholdsfortegnelsen.

9.1.1 Om appendiks/bilag

Ordene appendiks og bilag henviser i denne sammenhæng til det samme begreb.

Appendikser kan laves på forskellig vis, men det mest almindelige sker ved at man indsætter

```
\appendix
```

```
\appendix
```

i sin kode på det sted appendikset starter. Dette gør følgende

- (a) Nulstiller hovedafsnitstælleren (alt efter klasse, chapter eller section).

9. Mere om tekst

- (b) Laver om på udseendet af hovedafsnitstælleren. Normalt kommer den nu til at fremstå som store bogstaver.
- (c) Anvendes der kapitler, erstattes ordet »Kapitel« med »Bilag«. Hvilket dog kan konfigureres, se afsnit 9.4 på side 229.

Har man justeret tingene korrekt så kommer formelnumre til at fremstå som f.eks. (A.3).

Memoir. I indholdsfortegnelsen (se f.eks. afsnit 2.6 på side 36) markeres kapitler og appendices bare som numre og bogstaver. Ofte er det mere læsevenligt hvis der står »Kapitel« og »Bilag«. Anvendes *memoir*-klassen (hvilket er det vi anbefaler) så kan man heldigvis nemt konfigurere den til det. Se [Wilson \(2007a\)](#) kapitel 3.¹

Eksempel 9.3

```
\renewcommand\cftchaptername{\chaptername~}  
\renewcommand\cftappendixname{\appendixname~}
```

Hvis man samtidigt gerne vil have en adskillelse i TOCen så man kan se at nu starter bilagene så kan man i ovennævnte scenarie (i.e. med *memoir*) anvende

Eksempel 9.4

```
\appendix  
\appendixname
```

Navnet som indsættes styres via `\appendixpagename` og `\appendixtocname`.

Da `\appendix` laver om på den måde `\chapter` nummereres på, så kan det være et problem at vende tilbage med et almindeligt kapitel efter appendikserne/bilagene. I dette tilfælde kan man i stedet for `\appendix` anvende environmentet *appendices* omkring alle bilagene som i

Eksempel 9.5

```
\begin{appendices}  
  \include{bilag1}  
  \include{bilag2}  
  \include{bilag3}  
\end{appendices}  
  
\chapter{Noget andet}
```

Appendices til et kapitel

subappendices

Memoir-klassen har et *subappendices* environment som man kan pakke om nogle sektioner med under sektioner. Disse sektioner bliver så nummereret med store bogstaver, dvs. f.eks. sektion 1.A og undersektionen 1.A.1.

Bilag og appendiks

Sidebemærkning 9.1. Husk i øvrigt at `\appendix` kun bør optræde én gang. Den nulstiller jo kapitel tælleren hver gang man anvender den, hvilket kunne betyde at man fik op til flere »Bilag A«.

Man kan dog sagtens argumentere for at anvende både appendikser samt bilag. Vi har lavet en længere forklaring vedrørende dette i afsnittet »[Appendix OG bilag](#)« på side på side 328.

¹ Pakken `tocloft` skulle kunne gøre det i forbindelse med standardklasserne.

9.2 Fodnoter og endnotes/slutnoter

9.2.1 Mere om fodnoter

Fodnoter laves som tidligere nævnt (afsnit 2.5 på side 35) med makroen `\footnote{⟨tekst⟩}`. Desuden vil vi indskyde at skal man have ‘fodnoter’ i f.eks. titel eller forfatter delen af `\maketitle` så anvender man ikke `\footnote`, men i stedet `\thanks` (som bare er en specielt konfigureret fodnote).

Der er dog forskellige situationer hvor fodnoter enten slet ikke kan indsættes direkte, eller hvor man bør overveje om fodnoter er nødvendige.

- (i) Hvis man absolut ønsker at have en fodnote i en overskrift (`\chapter`, `\section`, etc.) så bør man anvende følgende syntaks (ellers ville fodnoten komme med i en eventuel header samt med i indholdsfortegnelsen).

```
\chapter[titel]{titel\footnote{en fodnote}}
```

Eksempel
9.6

Syntaksen er desuden omtalt i afsnit 9.1 på side 223.

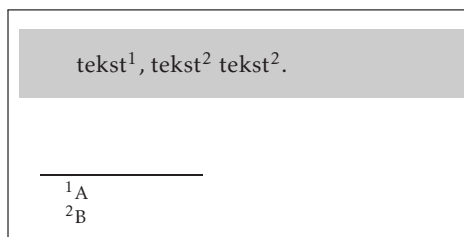
- (ii) I visse konstruktioner kan fodnoter slet ikke anvendes (teknisk set kan fodnoten ikke slippe ud). Som oftest kommer der slet ingen advarsel om at fodnoter ikke kan anvendes, de bliver bare væk, der kommer et nummer, men ikke nogen note. Som et eksempel har vi de grå kasser vi anvender om f.eks. bemærkninger. Disse er lavet med environmentet *shaded* (fra *memoir*-klassen/*framed*-pakken).

Hvis man absolut skal have en fodnote (i mange tilfælde er fodnoter overflødige eller direkte irriterende for læseren), så kan man dele fodnoten op i en markør og selve teksten.

```
\footnotemark{}          ({} for at den ikke skal spise et evt. efterfølgende mellemrum) \footnotemark
\footnotetext{⟨tekst⟩}   \footnotetext
```

`\footnotemark` placeres inden i konstruktionen hvori fodnoter ikke kan placeres direkte, og `\footnotetext{⟨tekst⟩}` placeres *efter* konstruktionen. `\footnotetext` vil anvende den nuværende værdi af fodnotetælleren `footnote`, derfor skal man tage sine forholdsregler, se [Eksempel 9.7](#). Bemærk at man desuden kan give begge makroer et valgfrit argument, hvor man kan angive nummeret der henvises til.

```
\begin{shaded}
  tekst\footnotemark, tekst\footnotemark{}
  tekst\footnotemark[\value{footnote}].
\end{shaded}
\addtocounter{footnote}{-1}
\footnotetext{A}
\addtocounter{footnote}{1}
\footnotetext{B}
```



Eksempel
9.7

Grunden til at pille ved `footnote`-tælleren på denne måde, er for at mindske muligheden for menneskelig-fejl.

- (iii) Et tredje yndet sted at placere fodnoter hvor de slet ikke hører hjemme er i billedteksten eller teksten til en tabel, som flyder. Dette bør man slet ikke gøre, så her vil vi slet ikke give nogen løsning. Skriv i stedet teksten om.

Til enten konfiguration a fodnoternes udseende eller for at få langt mere avancerede fodnoter henvises til pakkerne `footmisc` og `bigfoot`. Især den sidste giver mange nye muligheder, f.eks. muligheden for at anvende `\verb` (se afsnit 9.10 på side 249) direkte i fodnoter, noget som normalt ikke er muligt. Klassen *memoir* har nogle simple

footmisc
bigfoot

9. Mere om tekst

footmisc
bigfoot

muligheder for konfigurering af udseendet af fodnoterne, se [Wilson \(2010\)](#). Husk dog at anvender man f.eks. *footmisc* eller *bigfoot* så overskriver dette *memoir*-klassens egne fodnote metoder og konfigureringer, således at disse ikke fungerer mere.

Sidebemærkning 9.2 (Fodnoterne i denne bog).

Konfigurationen til at få fodnoter som i denne bog er

Eksempel

9.8



```
\setlength \footmarkwidth {3em}
\setlength \footmarksep {0em}
\setlength \footparindent {0em}
\addtolength\footnotesep {2pt}
\addtolength{\skip\footins}{10pt}
\footmarkstyle{\textsuperscript{#1}\hspace{0.25em}}
\renewcommand*\footnoterule{}
\renewcommand\foottextfont{\scriptsize\RaggedRight}
\renewcommand*\footfudgefiddle{80}
```

9.2.2 Endnotes/slutnoter

Hvis man anvender fodnoter rigtigt meget (f.eks. til at give ekstra information om et specielt emne) så kan det være en bedre ide at anvende *endnotes* i stedet. Det er så at sige det samme som fodnoter, men teksten bliver i stedet bare opsamlet og skrevet til slut (lidt på samme måde som indholdsfortegnelsen er lavet). Til dette anvendes pakken *endnotes* og man får de to mest almindelige makroer fra pakken:

endnotes

```
\endnote      \endnote{<tekst>}
\theendnotes  \theendnotes
```

Den første makro anvendes som man ville med `\footnote`, mens den anden anvendes på det sted hvor man ønsker at placere listen af endnotes. [Lavagnino \(2003\)](#) forklarer lidt mere om de makroer man kan anvende til konfigurering.

Nu er det jo sådan at man oftest anvender fodnoter, så man har sikkert skrevet sin tekst med `\footnote` og så er det jo er stort bøvl at skulle rette hele teksten for at ændre det hele til *endnotes*. Men frygt ej, der er en nemmere måde. Omdefiner `\footnote` til at være lig `\endnote`:

Eksempel

9.9

```
\let\footnote\endnote
```



Advarsel 9.3. Hvis man desuden anvender `\maketitle` og `\thanks` i sit dokument, da bør konstruktionen fra [Eksempel 9.9](#) først laves *efter* `\maketitle`. Gør man ikke brug af `\maketitle` kan man sagtens placere konstruktionen i *preamble*.

Memoir-klassen har sine egne indbyggede slutnoter, kaldet *pagenotes*, se [Wilson \(2010\)](#).

9.3 Mere om orddeling og sprog

Orddelingen i $\text{T}_{\text{E}}\text{X}$ tager udgangspunkt i visse orddelingsmønstre som så bliver kompileret ind i $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ -formaterne. Man vil normalt kunne se hvilke mønstre som er installeret i starten af ».log«-filen. Da mønstrene skal kompileres ind i formaterne er det vigtigt at man husker at få installeret de rigtige sprog når man installerer $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$.

Babel og orddeling

Sidebemærkning 9.4. Tidligere inkluderede MiKTeX ikke »*danish*« som standard. Men det er ret nemt at fikse. Gå til MiKTeX-menuen (Start → Programmer → ...) og vælg »*MiKTeX Options*«. Der vil man kunne finde en sprogopsætning og formaterne skal så lige genkompileres, men det skulle MiKTeX gerne selv spørge om.

I MiKTeX 2.9 inkluderes dansk som standard.

Man kan ret nemt se hvis de danske orddelingsmønstre mangler, bare oversæt et dokument som anvende dansk babel og kig i loggen. Hvis L^AT_EX er begyndt at tale om at den anvender »*dummy lang*« så har man ikke installeret de danske orddelingsmønstre.

Sidebemærkning 9.5 (Orddelingspunkter).

For en sikkerhedsskyld minder vi om at man via indsættelse af udvalgte »\-«'er i et ord kan markere hvor man vil tillade at L^AT_EX orddeler dette ord. Husk *altid* at markere *alle* lovlige delepunkter i ordet på denne måde, så har L^AT_EX noget at vælge i mellem. I standardkonfigurationen af L^AT_EX vil f.eks. `orddelings\-algoritme` betyde at der *kun* kan ske orddeling af dette ord mellem »orddelings« og »algoritme«. Husk derfor at markere alle orddelingspunkterne.

Sidebemærkning 9.6 (Få \- til at tilføje i stedet for at overskrive).

Ved at pille lidt ved definitionen af \- kan man godt på den til at *tilføje* tilføje delingspunkter ud over de som L^AT_EX selv finder via orddelingsmønstrene.

Følgende konstruktion skyldes *Jørgen Larsen* og er taget fra hans IMFUFA-L^AT_EX, se [Larsen \(2010\)](#).

```
\addto\extrasdanish{\babel@save\-}
\addto\extrasdanish{\def\-\{\nobreak\discretionary{-}{-}\bbl@allowhyphens}}
```

Som det ses er det noget som tilføjes den danske babel-opsætning.

Kodetip
9.1



Babel og orddeling

I afsnit 2.2 på side 24 forklarede vi om brugen af `\hyphenation`, for at rekapitulere så tilbyder `\hyphenation` en metode hvor man globalt kan specificere hvordan man ønsker at et ord skal orddeles. Man markerer stavelser via »-« (*ikke* med »\-«) og separerer listen med mellemrum. Desuden må man gerne have mange af disse kommandoer. Husk desuden at angiver man et ord i listen helt uden »-« så bliver dette ord altså slet ikke orddelt.

`\hyphenation`

Men hvad gør man når man anvender flere sprog i samme dokument? Dette er faktisk meget nemt idet `\hyphenation` tilføjer orddelinger til det nuværende sprog, så man skal i preamblen bare gøre noget lignende dette:

```
\selectlanguage{danish}% bare for en sikkerhedsskyld
\hyphenation{ord-de-lings-al-go-rit-me}
\selectlanguage{english}
\hyphenation{theo-rem-style}
\selectlanguage{danish}
```

Eksempel
9.10

Når man har større tekststykker hvor man ønsker at anvende et andet sprog (herunder dette sprogs orddelingsmønstre) så kan man som allerede nævnt anvende `\selectlanguage{<sprognavn>}`, som så skifter til et andet sprog, og man kan så skifte tilbage på samme måde. En anden mulighed er at anvende environmentet *otherlanguage* som har følgende syntaks

`\selectlanguage`
otherlanguage

9. Mere om tekst

```
\begin{otherlanguage}{\langle sprog \rangle}
  her er der orddeling efter sproget \langle sprog \rangle
\end{otherlanguage}
```

Indenfor dette environment får man så samme effekt som ved `\selectlanguage`, og ændringen gælder kun frem til slutningen af environmentet. Der er også et *otherlanguage**-environment som gør det samme, men her vil man ikke få ændret standardnavnene ved f.eks. tabeller, figurer eller kapitler.

Hvis man har en dansk tekst hvori det indgår nogle ord fra et andet sprog som man gerne vil have ord delt efter dette andet sprog så kan man anvende følgende makro

```
\foreignlanguage \foreignlanguage{\langle sprog \rangle}{\langle tekst \rangle}
```

Hvor `\langle sprog \rangle` naturligvis er et af de sprog man har angivet som options til enten dokumentklassen eller til selve babel-pakken.

For yderligere informationer om babel-systemet se [Braams \(2005\)](#).

Orddeling ved bindestreg og skråstreg

I ord hvor der optræder bindestreg til L^AT_EX *ikke* lave automatisk orddeling. Som forklaret i afsnit 2.2 på side 27 er løsningen til dette, under dansk babel, at man i stedet for »-« anvender »"-«.

Man har et tilsvarende problem ved »/«. Dette har Jørgen Larsen løst i [Larsen \(2010\)](#) via følgende konstruktion:

Eksempel

9.11



```
\makeatletter
\declare@shorthand{danish}{"/}{\nobreak\discretionary{/}{/}{\bbl@allowhyphens}}
\makeatother
```

Hvorefter man tilsvarende kan anvende »"/« og dermed få orddeling i begge dele af et ord med skråstreg.

Lad os lige gentage koden der skal til for at få featuren med »"-« til at virke med engelsk babel.

Eksempel

9.12



```
\usepackage[english]{babel}
\makeatletter
\useshorthands{"}
\defineshorthand{"-}{\nobreak-\bbl@allowhyphens}}
\makeatother
```

Mærkelige orddelinger

På grund af en dårlig konfiguration af den danske opsætning af babel kan man godt risikere at få orddelinger så som »h-vor«. Man kan afhjælpe dette ved at placere følgende i sin preamble efter man har indlæst babel.

Eksempel

9.13



```
\renewcommand{\danishhyphenmins}{22}
```

Dette betyder at der ved deling skal være mindst to tegn før delingen og mindst to i slutningen. Det er disse værdier de danske orddelingsmønstre er lavet med. (Babel-pakken sætter værdien »12«, hvilket derfor ikke passer sammen med de danske orddelingsmønstre.)

Undgå orddeling

Sidebemærkning 9.7. Værdien af `\danishhyphenmins` skulle nu være blevet ændret til 22 i selve L^AT_EX-installationerne. Tjek selv i `danish.ldf`.

Undgå orddeling

Har man et ord som orddeles, men det ønsker man ikke, så kan man pakke den ind i en `\mbox{⟨tekst⟩}`. Man kan desuden anvende `\foreignlanguage` sammen med sproget »*nohyphenation*«.

`\mbox`
`\foreignlanguage`

```
tekst tekst tekst tekst tekst
orddelingsalgoritme

tekst tekst tekst tekst tekst
\foreignlanguage{nohyphenation}%
{orddelingsalgoritme}
```

```
tekst tekst tekst tekst tekst orddelingsal-
goritme
    tekst  tekst  tekst  tekst  tekst
orddelingsalgoritme
```

Eksempel
9.14

Se desuden *flushleft*, *flushright*, `\raggedleft`, `\raggedright`, `\RaggedLeft` og `\Ragged-Right`.

Teste efter sprog

Laver man sine egne makroer så kan det være en fordel at kunne teste efter hvilket sprog som er taget i brug lige nu (forudsat dokumentet anvender `babel`). Til dette kan man anvende følgende konstruktion

```
\iflanguage{⟨sprognavn⟩}{⟨Ja-kode⟩}{⟨Nej-kode⟩}
```

Hvis `⟨sprognavn⟩` er aktivt lige nu så udføres `⟨Ja-kode⟩`.

Lokalt skift af sprog

Brug følgende konstruktion

```
\begin{otherlanguage}{⟨sprog⟩}
⟨indhold⟩
\end{otherlanguage}
```

Så skiftes til `babel`-opsætningen samt orddelingsmønstrene for `⟨sprog⟩` indenfor dette environment. Nyttigt omkring et engelsk abstrakt eller citat.

9.4 Ændring af standardnavne kontrolleret af babel

Normalt lader man pakken `babel` om at styre standardnavnene som anvendes i dokumentet, så som figur, tabel, Kapitel, Bilag osv. Det er dog ikke altid man er helt tilfreds med det valg som er gjort. Eksempelvis ønsker mange at appendikser hedder »Appendiks« i stedet for »Bilag«. Navnene er listet i Tabel 9.1 på næste side.

babel

Disse navne kan man så ændre via `\renewcommand`, men pga. brugen af `babel` skal man gøre det på en speciel måde (bemærk det første »s« i `\captionsdanish`).

`\addto`
`\captionsdanish`

```
\addto\captionsdanish{
  \renewcommand\appendixname{Appendiks}
  \renewcommand\contentsname{Indholdsfortegnelse}
}
```

Eksempel
9.15



9. Mere om tekst

Tabel 9.1: Standardnavne i L^AT_EX (engelske)

<code>\abstractname</code>	Abstract	<code>\indexname</code>	Index
<code>\alsoname</code>	see also ^a	<code>\listfigurename</code>	List of Figures
<code>\appendixname</code>	Appendix	<code>\listtablename</code>	List of Tables
<code>\bibname</code>	Bibliography ^b	<code>\partname</code>	Part
<code>\chaptername</code>	Chapter ^c	<code>\refname</code>	References ^d
<code>\contentsname</code>	Contents	<code>\seename</code>	see ^e
<code>\figurename</code>	Figure	<code>\tablename</code>	Table

^a Kommer normalt fra `makeidx`-pakken eller `memoir`-klassen. Anvendes i indekser.

^b Anvendes til overskriften af litteraturlisten i klasser som har kapitler, f.eks. `report`, `book` eller `memoir`.

^c Kun defineret i klasser som understøtter `\chapter`. Dvs. f.eks. `report`, `book` eller `memoir`.

^d Overskriften til litteraturlisten i klassen `article`.

^e Kommer normalt fra `makeidx`-pakken eller `memoir`-klassen. Anvendes i indekser.

Man skal naturligvis anvende `\captions<prog>` når man anvender andre sprog og ovenstående skal naturligvis placeres i preambelen *efter* `babel`-pakken.

Tip 9.8. Man kan sagtens tilføje andre ting til [Eksempel 9.15](#) på forrige side som så vil blive udført når man skifter til det pågældende sprog.

9.5 Tekstfyld – Lorem ipsum

Indenfor typografien anvendes *Lorem ipsum* som en form for fyldtekst når man f.eks. skal præsentere et grafisk design (hvor indholdet ikke er vigtigt), se

<http://en.wikipedia.org/wiki/Lipsum>

I L^AT_EX kan vi f.eks. anvende en af Lorem ipsum afsnittene til at fremvise hvordan en font ser ud, eller man kan anvende den i sin tekst som *fyld* indtil man får skrevet det indhold som skal stå lige der, f.eks. indledningen til sit speciale. Pakken `lipsum` giver adgang til 150 Lorem ipsum afsnit, syntaksen er

lipsum

```
\lipsum <start nummer>-<slut nummer>
```

eller

```
\lipsum <nummer>
```

`\lipsum` er det samme som `\lipsum[1-7]`.

Eksempel 9.16



```
\usepackage{lipsum}
```

— slut preamble —

```
\lipsum[31]
```

Sed consequat tellus et tortor. Ut tempor laoreet quam. Nullam id wisi a libero tristique semper. Nullam nisl massa, rutrum ut, egestas semper, mollis id, leo. Nulla ac massa eu risus blandit mattis. Mauris ut nunc. In hac habitasse platea dictumst. Aliquam eget tortor. Quisque dapibus pede in erat. Nunc enim. In dui nulla, commodo at, consetetuer nec, malesuada nec, elit. Aliquam ornare tellus eu urna. Sed nec metus. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Man skal naturligvis ikke forvente at orddelingen fungerer for denne tekst.



En speciel ting ved lipsum er at alle teksterne afsluttes med hvad der svarer til en blank linie, i.e. de er afsnit hver for sig. Det er ikke altid at man ønsker dette når man er ved at teste et design (denne ‘blanke linie’, kan undgås via »\lipsum*«). Et alternativ til lipsum er at anvende pakken blindtext som giver en lignenden funktionalitet, men på en anderledes og lidt mere konfigurerbar måde.

9.6 Fontrelaterede emner

Det er ret vigtigt at man forstår at der i L^AT_EX faktisk er to fontsystemer i spil. Et til den almindelige tekst og et til matematikken. Udskifter man den ene, har man altså *ikke* udskiftet den anden. Der findes forskellige pakker som kan skifte tekstfonten, men anvender man en af disse og samtidigt skriver noget matematik, så vil denne matematik stadigvæk være i standardskrifttypen – Computer Modern. Dette er ofte ikke videre pænt, så det anbefales derfor, at når man skifter fonten og man ved man har brug for matematik i dokumentet, så vælg en skrifttype som inkluderer en passende fontopsætning til matematikken.

9.6.1 Fontvalg med matematiksupport

Som allerede nævnt deler L^AT_EX fontene op i to dele: Tekst og matematik. Hvilket betyder at man kan skifte tekstfonten men ikke matematikfonten. Dette kan have ret negative konsekvenser, hvis f.eks. den nye tekstfont ikke harmonerer særligt godt sammen med Computer Modern (som jo er vores standardfont).

En anden ting man skal være opmærksom på er at der ved mange af fontene er tale om såkaldte *kloner*, dvs. fonte som er kraftigt inspirerede af en given font, men som har et andet navn i virkeligheden. Et eksempel på dette er at Times fonten i L^AT_EX faktisk hedder Nimbus.

Fontene vi enten vil vise eksempler på, eller henviser til er alle frittilgængelige. Dvs. de kan legalt hentes på nettet uden betaling. En smart oversigt over frittilgængelige fonte (visse med matematiksupport) kan ses på <http://www.tug.dk/FontCatalogue/>. Stor hyldest til Palle Jørgensen for at have samlet kataloget.

Sampletæksten som er anvendt til alle eksemplerne er:

```
almindelig tekst, \textit{kursivt}, \textsf{sans serif},
\textbf{fed skrift}, \textsc{Small Caps}
\l
  x\in\mathbb{R}, c\in\mathcal{C} \quad \quad \int_{-\infty}^{\infty} f(x)\, dx
\r
\alpha\beta\eta\Theta,\zeta,\xi$, $Y$ vs. $\Upsilon$, $v$ vs. $\nu$, $\mathbf{ABC}$,
$\mathbf{abc}$, $\mathbf{\alpha\beta\gamma}$
```

**Eksempel
9.17**

Pakkerne som er anvendt til alle eksemplerne er inputenc, babel, amsmath, amssymb og bm, for overskueligheden vises disse ikke i eksemplerne. Bemærk desuden at for at spare plads er visse længder blevet ændret.

Sidebemærkning 9.9. Prøveteksten er valgt for at vise nogle forskellige ting: (a) Forskellige udgaver til almindelig tekst, (b) lidt matematik og nogle græske bogstaver, (c) nogle fede symboler, samt (d) nogle symboler som kan give problemer i visse skrifttyper. I forbindelse med det sidste punkt så er det kursivt ν versus det græske symbol ν (ν), idet disse ofte ligner hinanden. Det er ζ (ζ) og ξ (ξ) samt kursivt Y versus det store græske symbol Υ (Υ).

9. Mere om tekst

Sidebemærkning 9.10. Visse af de fonte vi viser her, er måske ikke med i en standard L^AT_EX-installation. Fontene vil så kunne fås via CTAN. Med hensyn til f.eks. T_EX Live (som er det jeg bruger) visse fonte ikke med i distributionen, hvilket skyldes licensregler. Se afsnittet »*Andet vedr. T_EX Live* på side 383, hver forklares hvordan f.eks. Garamond og LuxiMono installeres i T_EX Live.

Computer Modern

Dette er standardskrifttypen i L^AT_EX.

Eksempel 9.18

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS
$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$
 $\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC**, *abc*, $\alpha\beta\gamma$

Latin Modern

Latin Modern er en ret ny font, som er bygget over Computer Modern. Det er meningen at denne font med tiden skal afløse CM som standardfonten i L^AT_EX. Den umiddelbare fordel ved at avnende Latin Modern frem for f.eks. almindelig Computer Modern, er at filerne kommer til at fylde mindre. De fleste L^AT_EX-distributioner anvender den såkaldte CM-super skrifttype som standard, denne font er fremstillet på en utraditionel måde og fylder derfor en hel del. Latin Modern er lavet fra bunden og fylder derfor langt mindre.

Eksempel 9.19

```
\usepackage{lmodern}
```

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS
$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$
 $\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC**, *abc*, $\alpha\beta\gamma$

Tip 9.11. Sidder man på MAC og anvender editoren TeXShop, så vil lmodern-pakken også hjælpe. Så vil teksten nemlig blive meget klarere i TeXShops indbyggede fremviser. Det skyldes vist noget med de installerede fonte i MacTeX eller lignende.

Times

Bemærk jf. diskussionen ovenfor at man *ikke* anvender pakken *times* (den skifter kun tekstfonten, ikke matematikfonten), men i stedet mathptmx og også indeholder definitioner til matematikken.

Eksempel 9.20

```
\usepackage{mathptmx}
```

Displaydelen af eksempel 9.20 findes på næste side

Palatino

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS

$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$

$\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC, abc, $\alpha\beta\gamma$**

**Eksempel
9.20
fortsat**

Times matematikfonten har den ulempe at den har slet ingen fede symboler så her må alt simuleres via `\bm`.

Sidebemærkning 9.12 (Vedr. brug af Times).

Det anbefales generelt at med mindre man er tvunget til det (og så bør man protestere højlydt) så bør man ikke anvende Times.

Times er overbrugt i alle mulige sammenhænge, sikkert fordi det er (har været?) standardfonten i Word. Denne overbrug gør det altså ikke til en pænt font.

Desuden passer de anvendte matematiksymboler (i `mathptms`-pakken) ikke sammen med Times, hvilket giver en anden uregelmæssighed.

Palatino

Ønsker man palatino som tekst- og matematikfont, så har man umiddelbart to muligheder, enten via pakken `mathpazo` som anvender Pazo fontene til matematik.² Bemærk iøvrigt den ret så bizarre konvention ved »Y« versus »\Upsilon«.

mathpazo

```
\usepackage{mathpazo}
```

**Eksempel
9.21**

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS

$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$

$\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC, abc, $\alpha\beta\gamma$**

Den anden mulighed er Euler-fontene (`eulervm`) som egentlig er matematikfont, så her skal man selv huske at skifte tekstfonten.

eulervm

```
\usepackage{palatino,eulervm}
```

**Eksempel
9.22**

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS

$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$

$\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC, abc, $\alpha\beta\gamma$**

Utopia og fourier

Utopia er tekstfonten mens Fourier anvendes til matematikken. Der er tale om en mørkere font i slægt med Palatino eller Times. Fourier fonten er frit tilgængeligt mens man kan hente Utopia fra CTAN.

```
\usepackage{fourier}
\usepackage[T1]{fontenc}
```

**Eksempel
9.23**

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS

$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$

$\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC, abc, $\alpha\beta\gamma$**

² FixMe Note: `mathpazo` har desuden nogle options man måske kunne bruge til noget

Sidebemærkning 9.13. Den nuværende version af Fourier opsætningen indeholder en spøjs lille fejl som gerne skulle blive fikset på et eller andet tidspunkt. Fejlen er at `\Big(` og `\bigg(` giver den samme størrelse parentes.

Et muligt fix er at tilføje følgende efter man indlæser `fourier`-pakken (fix fra Morten Høgholm):

Eksempel 9.24



```
\makeatletter
\AtBeginDocument{%
  \ifpackageloaded{amsmath}{
    \def\bigg@#1#2{%
      {\@mathmeasure\z@\nulldelimiterspace\z@}%
      {\delimiterfactor=1000\relax \left#2\vcenter to#1\bigg@size{}\right.}%
      \box\z@}}{
\PackageError{master}{Are you kidding, you're not using amsmath?!}{}
}}
\makeatother
```

New Century Schoolbook og fourier

Mere eller mindre som ovenfor, her bare med New Century Schoolbook som tekstfont suppleret med Fourier til matematik.

Eksempel 9.25

```
\usepackage{fouriernc}
\usepackage[T1]{fontenc}
```

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS

$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$

$\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC**, *abc*, $\alpha\beta\gamma$

Garamond

mathdesign

Pakken `mathdesign` inkluderer en større samling af matematikfonte samt support for forskellige fonte, f.eks. en Garamond variant. Fontene skal igen hentes fra CTAN.

Eksempel 9.26

```
\usepackage[urw-garamond]{mathdesign}
\usepackage[T1]{fontenc}
```

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS

$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$

$\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC**, *abc*, $\alpha\beta\gamma$

Kp-fonts

Eksempel 9.27

```
\usepackage{kpfonts}
\usepackage[T1]{fontenc}
```

Displaydelen af eksempel 9.27 findes på næste side

Concrete

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS

$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$

$\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC**, *abc*, $\alpha\beta\gamma$

Eksempel
9.27
fortsat

Concrete

Dette er ligesom Computer Modern en font designet af Donald Knuth suppleret med nye matematikfonte. Bemærk at fonten i sig selv ikke findes i fed. For yderligere information se Vieth (1999).

```
\usepackage{concmath}
\usepackage[T1]{fontenc}
```

Eksempel
9.28

almindelig tekst, *kursivt*, sans serif, **fed skrift**, SMALL CAPS

$$x \in \mathbb{R}, c \in \mathcal{C} \quad \sum \int_0^1 f(x) dx$$

$\alpha\beta\eta\Theta, \zeta, \xi, Y$ vs. Υ, v vs. ν , **ABC**, *abc*, $\alpha\beta\gamma$

Tip 9.14. Står man lige og skal bruge et billede af en formel til en hjemmeside, så kan denne font anbefales sammen med programmet `dvipng` (som i det mindste følger med L^AT_EX-distributionen T_EX Live).

dvipng

```
dvipng -M0 -T tight -x 1200 fil.dvi
```

Dette vil tage hver af siderne i `fil.dvi` og lave den om til et PNG-billede.

9.6.2 Et par sans-serif skrifttyper

Følgende er et par muligheder når det angår sans-serif fonte, dvs. fonte uden fodtegn (de såkaldte seriffer). Igen kan man finde eksempler via Palles oversigt, <http://www.tug.dk/FontCatalogue/>.

Fælles for disse pakker er at de kun ændrer ved sans serif fonten, dvs. `\textsf` og `\sffamily`.

Computer Modern sans serif

```
% standard sans serif font
----- slut preamble -----
\sffamily Noget tekst \textbf{tekst}
\textit{tekst} \textbf{\textit{tekst}}
```

Noget tekst **tekst** *tekst* **tekst**

Eksempel
9.29

Denne samplettekst anvendes desuden i de to næste eksempler.

Helvetika klon

Visse fonte har en ret stor x -højde i forhold til den tekstfont man har besluttet sig for at gøre brug af. Her har pakkerne ofte en option som hedder »scaled« som kan anvende til at skalere fonten med.

```
\usepackage[scaled=0.85]{helvet}
```

Noget tekst **tekst** *tekst* **tekst**

Eksempel
9.30

9. Mere om tekst

Bera sans

**Eksempel
9.31**

```
\usepackage[scaled]{berasans}  
\usepackage[T1]{fontenc}
```

Noget tekst **tekst** tekst **tekst**

9.6.3 Fast-bredde skrifttype (skrivemaskinefont)

Fast-bredde skrifttype kaldes desuden ofte *monospace*- eller *typewriter*-font. En sådan skrifttype anvendes oftest til at fremvise programkode eller når man vil lave ordret (verbatim) fremvisning af noget eksternt tekst som f.eks. en udskrift fra SAS.

Computer Modern (som er L^AT_EXs standard font) har en pæn fast-bredde font, som desværre ikke findes i fed, hvilket er lidt en skam. Man har i stedet visse andre muligheder.

Computer Modern Typewriter

**Eksempel
9.32**

```
% standard font  
----- slut preamble -----  
\ttfamily  
Tekstilfabrikant \\  
\textbf{tekstilfabrikant}\  
\textit{tekstilfabrikant}
```

Tekstilfabrikant
tekstilfabrikant
tekstilfabrikant

Vi vil anvende denne tekst i de følgende eksempler.

TX Typewriter

Denne kan være et godt alternativ til Computer Modern Typewriter, som faktisk også har en fed udgave.

**Eksempel
9.33**

```
\renewcommand\ttdefault{txtt}
```

Tekstilfabrikant
tekstilfabrikant
tekstilfabrikant

LuxiMono

Denne font er anvendt som fastbredde font i denne bog. Den passer godt sammen med mørke fonte så som Palatino, Times eller Utopia.

**Eksempel
9.34**

```
\usepackage[scaled]{luximono}  
\usepackage[T1]{fontenc}
```

Tekstilfabrikant
tekstilfabrikant
tekstilfabrikant

Det er meget vigtigt at man husker »\usepackage[T1]{fontenc}«. På grund af licensen skal LuxiMono installeres direkte fra CTAN.

BeraMono

**Eksempel
9.35**

```
\usepackage[scaled]{beramono}  
\usepackage[T1]{fontenc}
```

Tekstilfabrikant
tekstilfabrikant
tekstilfabrikant

Vilkårlig fontstørrelse

9.6.4 Vilkårlig fontstørrelse

Vi har allerede set (afsnit 2.3 på side 30) at L^AT_EX har nogle forskellige makroer som lokalt kan ændre fontstørrelsen. Der kan dog være situationer (som f.eks. på forsider) hvor man kan have brug for at kunne anvende helt brugerspecifikke fontstørrelser.

I L^AT_EX styrer man dette med følgende konstruktion

```
\fontsize{<størrelse>}{<baselineskip>}\selectfont
```

```
\fontsize  
\selectfont
```

hvor *<baselineskip>* er linieafstanden man ønsker i denne størrelse.

Sidebemærkning 9.15. Af tekniske årsager, vil det ofte være nødvendigt, at man sørger for at anvende pakken `fix-cm` når man ønsker vilkårlige fontstørrelser.

`fix-cm`

³ Desuden vil det være en god ide at anvende Type 1 fonte (også kaldet outline-fonte eller PostScript fonte). De fleste L^AT_EX-installationer anvender i dag denne type fonte som standard. For Computer Moderns vedkommende anvendes fonten CM-Super, men man kan jo være så uheldig at denne ikke er en del af ens egen L^AT_EX-installation, i så fald kan man anvende pakken `lmodern` som erstatning.

`lmodern`

```
\usepackage{fix-cm}  
----- slut preamble -----  
\fontsize{3cm}{4cm}\selectfont  
Stor font
```

Stor font

Eksempel
9.36


Denne fontstørrelsesændring vil så være gældende lokalt som ved de mere almindelige makroer, som f.eks. `\large`, eller frem til næste `\normalsize`. At ændre tekstfontstørrelsen (og tilsvarende for matematik) er ikke helt nemt, og det anbefales ikke, med mindre det er bydende nødvendigt.

9.6.5 Større linieafstand

Da de fleste dokumentklasser er designet til Computer Modern fonten (L^AT_EXs standardfont) kan det ved visse tekstfonte være en god ide at man forøger linieafstanden en lille smule. Dette gøres ved, i preamblen, at placere

```
\linespread{<faktor>}
```

```
\linespread
```

F.eks. `\linespread{1.033}`.

Sidebemærkning 9.16. Hvis man anvender `\linespread` i selve teksten, så skal det ske via

```
\linespread{<faktor>}  
\selectfont
```

den ekstra linie skal med for at aktivere ændringen.

³ FixMe Note: bør skrives om, begrundelsen er ikke helt korrekt

9. Mere om tekst

Til korrekturlæsning kan det desuden være en rar ting at kunne printe værket med dobbelt linieafstand (så man nemmere kan skrive rettelser over teksten). Dette kan hurtigt opnåes med

Eksempel 9.37 `\linespread{1.4}`

i preamblen. Ved indlevering af artikler til *peer review*, kan dobbelt linieafstand også være et krav fra forlaget.

Eksempel 9.38 `\linespread{1.3}`



```
----- slut preamble -----  
  
\small  
\lipsum[13]
```

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Man bør *aldrig* aflevere en tekst i dobbeltlinieafstand. Det er ikke særligt læsevenligt og så er det spild af papir. Se evt. diskussionen på <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=linespace>. Er man kommet i den uheldige situation, at man *er tvunget til* at have en stor og grim linieafstand anbefales det at man anvender pakken `setspace`. Yderligere behandling af denne pakke ligger udenfor rammerne af denne bog, se [Tobin \(2000\)](#) for yderligere information.

Memoir. Igen har *memoir*-klassen direkte indlejret funktionaliteten af denne pakke. Dog under lidt andre navne og med en lidt anden (bedre?) implementation, se [Wilson \(2010\)](#) afsnit 3.3.2.

9.6.6 At pille ved afstanden mellem bogstaver (*letterspacing*)

`letterspace` Indenfor typografien har man det engelske begreb *letterspace* (aner ikke hvad dette hedder korrekt på dansk – måske *bogstavafstand*?), som omhandler det at det til tider kan være en god ide at forøge afstanden mellem bogstaverne i et ord. Dette kan f.eks. anvendes til at fremhæve ord, til overskrifter eller specielt til forsider.

`soul` Til dette kan man anvende pakken `soul`. Vi vil ikke komme nærmere ind på brugen af pakken eller de faldgruber der er, henviser i stedet til [Franz \(2003\)](#). Vi anvender senere denne pakke til at lave *Louises forside*, se figur 12.2 på side 317. Pakken `soul` kan ud over at pille ved mellemrummet mellem bogstaver også tilbyde understregning af tekst som faktisk kan fortsætte over en linieombrydning.⁴ Her er dog et simpelt eksempel.

Eksempel 9.39 `\usepackage{soul}`



```
----- slut preamble -----  
  
\so{dette er en tekst}  
  
\scshape{\so{dette er en tekst}}
```

dette er en tekst
DETTE ER EN TEKST

⁴ Understregning af tekst er dog generelt noget som bør forblive i håndskrevne noter. De har ikke noget at gøre i trykt tekst.

Lister

En ting som det kan være en god ide at anvende fra `soul` er muligheden for selv at definere en `\so`-lignende makro (de tre sidste argumenter skal alle være længder):

```
\sodef<makronavn>{}{<mellem bogstaver>}%  
{<indre ordmelletrum>}{<ydre ordmelletrum>}
```

`\sodef`

Så kan man selv justere på længderne. Det er en god ide at specificere de tre obligatoriske længder i enheden `em`, thi denne skaleres automatisk sammen med den nuværende fontstørrelse. At lave en makro via ovenstående konstruktion kan være nødvendig f.eks. til forsider, thi selve `\so`-makroen vil efterlade mellemrum på steder hvor det ikke er sjovt (efter argumentet), hvorved centering ser underlig ud.



Advarsel 9.17 (UTF8 og soul-pakken).

`Soul` pakken kan pt. ikke håndtere æøå eller accent bogstaver skrevet under UTF8 encoding. Den fungerer dog fint med Latin1.

Løsningen på dette er at anvende \LaTeX -koder til netop disse tegn når man vil anvende det inden i noget som køres gennem `soul`, dvs. skriv »`h\o tyv`« i stedet for »`høtyv`«.

9.7 Lister

Vi har tidligere set de tre standard lister i \LaTeX : *itemize*, *enumerate* og *description*. Vi har desuden set at med pakken `enumerate` kan man selv få lov til at styre udseendet af nummereringen. I dette afsnit vil vi først se lidt på pakken `paralist` som fortsætter stilen fra `enumerate`-pakken med yderligere lister. Dette efterfølges af et kig på en nyere pakke (`enumitem`) som egentlig går tilbage til de tre basale lister, men her gøres disse ekstremt konfigurerbare.

itemize
enumerate
description

Sidebemærkning 9.18. Et par småting:

- (a) Lister kan generelt kun nestes fire lag. Dvs. man kan starte lister i op til fire niveauer, hvilket er nok for de fleste.
- (b) Hver af de fire lag i *enumerate* har deres egen tæller, i rækkefølge er de: `enumi`, `enumii`, `enumiii` og `enumiv`. Dvs. i øverste niveau anvendes `enumi`. Denne viden kan f.eks. anvendes til at justere på hvilket nummer en given liste skal starte ved.

Memoir. `Memoir`-klassen har et par smarte makroer til at formindske den vertikale afstand mellem punkterne i en liste. De anvendes indeni en given liste, eller kan angives globalt (med et lidt anderledes navn), se makroerne `\firmlist`, `\tightlist`, `\firmlists` og `\tightlists`.

Personligt anbefaler jeg dog brugen af `enumitem` hvis man ønsker at konfigurere listerne, denne pakke beskriver vi i afsnit 9.7.2 på side 241.

9. Mere om tekst

Sidebemærkning 9.19. Lige et godt råd: Når man f.eks. er ved at skrive et bevis med en masse længere punkter, så bør man lade være med at skrive dette som en liste. Det er fint nok hvis listen ikke er så lang, men hvis den fylder flere sider, så er det spild af plads (husk at punkterne i standardlisterne rykkes et stykke ind, så de har altså en større venstre margin).

Her er det bedre simpelt hen at anvende nye afsnit. Enten starter man et nyt afsnit, med noget lignende: *I tilfældet ...*, eller hvis hvert punkt har et bestemt nummer/overskrift så kan man med fordel anvende `\paragraph{<tekst>}`. Denne makro har den fordel at den efterlader lidt afstand til den førliggende tekst og selve teksten efter starter lige efter `{<tekst>}`.

`\paragraph`

Eksempel 9.40



```
\begin{proof} For at bevise (1) og (2)
tekst tekst tekst tekst tekst tekst tekst
\paragraph{(1)} bla bla bla bla
bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla
\paragraph{(2)} bla bla bla bla
bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla
\end{proof}
```

Bevis. For at bevise (1) og (2) tekst tekst tekst
tekst tekst tekst tekst

(1) bla bla bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla

(2) bla bla bla bla bla bla bla bla bla bla
bla bla bla bla bla bla bla bla □

Personligt ville jeg nok lige pille lidt ved afstanden over `\paragraph`. I *memoir*-klassen er standard afstanden over `\paragraph` mindre end den er i *article*-klassen (som er anvendt til at lave eksemplet herover).



Det skal lige nævnes at en version af `enumitem` (fra 2011), nu nemt kan lave ‘fuldbredde lister’ altså lister som kommer til at ligne resultatet fra [Eksempel 9.40](#). Så man kan godt bygge tingene op som en liste, bare sørg for at man ikke spilder en masse plads til venstre.

9.7.1 Flere lister med *paralist*-pakken

I afsnit [2.8 på side 39](#) så vi at man med pakken `enumerate` kunne få en udvidet syntaks for *enumerate*-environmentet. En anden mulighed for at få samme ekstra funktionalitet er pakken `paralist`. Der er nøjagtigt samme syntaks som ved *enumerate* men nu kan man også give *itemize* et valgfrit argument.⁵ Man får desuden flere ekstra environments.

`paralist`

For mere information om `paralist`, se [Schandl \(2002\)](#).

Enumerate og itemize

Bare for at rekapitulere, så har begge af disse environments nu et valgfrit argument.

⁵ Denne funktionalitet sammen med funktionaliteten fra `enumerate`-pakken, findes allerede i *memoir*-klassen.

Kompakte lister

<pre>\usepackage{paralist} ----- slut preamble ----- \begin{enumerate}[(a)] \item tekst \item mere tekst \end{enumerate} \begin{itemize}[\ensuremath{\ast}] \item tekst \item mere tekst \end{itemize}</pre>	(a) tekst (b) mere tekst * tekst * mere tekst
--	--

Eksempel

9.41



Kompakte lister

Men pakken giver desuden ultra kompakte versioner af listerne, kaldet hhv. *compactenum* og *compactitem*. Meget anvendelige hvis man lider af pladsmangel, se [Eksempel 9.42](#).

compactenum
compactitem

<pre>\usepackage{paralist} ----- slut preamble ----- \begin{compactenum}[(a)] \item tekst \item mere tekst \end{compactenum} Tekst i mellem \begin{compactitem}[\ensuremath{\ast}] \item tekst \item mere tekst \end{compactitem} Tekst efter.</pre>	(a) tekst (b) mere tekst Tekst i mellem * tekst * mere tekst Tekst efter.
--	--

Eksempel

9.42



En løbende liste

Her menes en liste af typen (a) punkt 1, (b) punkt 2, ... Dette kan man selvfølgelig skrive pr. håndkraft, men fordelene er igen at man kan henvise direkte til nummeret. Man anvender her listen *inparaenum*, se [Eksempel 9.43](#).

inparaenum

<pre>\usepackage{paralist} ----- slut preamble ----- Noget tekst gik sig en tur, og mødte \begin{inparaenum}[(i)] \item en hest, \item en nisse og \item noget den ikke vidste hvad var. \label{ukendtitem} \end{inparaenum} (\ref{ukendtitem}) viste sig senere at være en Ubuntu installation.</pre>	Noget tekst gik sig en tur, og mødte (i) en hest, (ii) en nisse og (iii) noget den ikke vidste hvad var. (iii) viste sig senere at være en Ubuntu installation.
--	---

Eksempel

9.43



9.7.2 Mere kontrol over lister med *enumitem*-pakken

Der er et par kritikpunkter til de almindelige listekonstruktioner.

9. Mere om tekst

- (i) Da afstandsoplysningerne er hardkodet ind i definitionen af listerne, er det meget besværligt efterfølgende at rette disse parametre. Man kan rette på den vertikale afstand (se f.eks. `\firmList` og `\tightList` i *memoir*-klassen) mellem punkterne undervejs, men så heller ikke ret meget mere.
- (ii) Det er ret besværligt at en reference til f.eks. »(4)« kun kommer ud som »4«, man skal selv huske formateringen (og huske at rette alle steder hvis man senere ændrer format).
- (iii) Det er også ret forvirrende at man skal huske på at beskytte »a«, »I« etc. hvis man ikke ønsker at de skal fortolkes i forbindelse med tællerformatering.
- (iv) Desuden har mange brug for muligheden for at slutte en nummereret liste, skrive noget tekst og så fortsætte listen.

enumitem Svaret på disse problemer kan man sige kom med Javier Bezos' *enumitem*-pakke.

Sidebemærkning 9.20. Før vi går videre er der lige et par ting som vi gerne vil nævne.

- (a) Som vi skal se anvender *enumitem* en helt anden syntaks til at angive hvordan teksten til f.eks. *enumerate* skrives, i værste fald kunne dette betyde at man skulle skrive sit dokument om hvis man ønskede at anvende *enumitem*.

Det er dog ikke så slemt, anvend

Eksempel 9.44

```
\usepackage[shortlabels]{enumerate}
```

så understøttes også `\begin{enumerate}[(i)]`.

enumerate
itemize
description

- (b) Bare for god ordens skyld minder vi desuden om at *enumitem* (indtil videre) *kun* berører *enumerate*, *itemize* og *description*. Man kan også definere sine egne lister af forskellig type. Man kan endda indlæse pakken uden at den rører ved standardlisterne, herefter man så anvende pakken til at definere sine egne ekstra lister.

Vi starter med et eksempel hvor man kan se hvad man f.eks. skal erstatte syntaksen »[(I)]« med.

Eksempel 9.45



```
\usepackage{enumitem}
```

_____ slut preamble _____

```
\begin{enumerate}[label=(\Roman*)]
\item Punkt 1\label{i}
\item Punkt 2
\end{enumerate}
reference: \ref{i}
```

(I) Punkt 1
(II) Punkt 2
reference: (I)

Bemærk at formateringen af punktet automatisk bevares i referencen. I stedet for *enumerate*-pakkens syntaks hvor man skulle anvende »1«, »a«, »A«, »i« eller »I«, skal man nu i stedet anvende henholdsvis `\arabic*`, `\alph*`, `\Alph*`, `\roman*` eller `\Roman*` (bemærk brugen af *). Hvilket både giver bedre mening, men også fjerner behovet for at skulle 'beskytte' tællerrepræsentationen, sammenlign med [Eksempel 2.33](#) på side 40.

`\arabic*`
`\alph*`
`\Alph*`
`\roman*`
`\Roman*`

Som nævnt vil referencen som standard gemme formateringen af tælleren. Dette kan være uheldigt i visse sammenhænge, f.eks. hvis labelen skal skrives i fed, så ønsker man sikkert ikke at referencer skal være i fed. Dette styres af nøglen »ref«, som så kan anvendes til helt at om konfigurere udseendet af referencen, se de lettere urealistiske forslag i [Eksempel 9.46](#) på næste side.

ref

Mere kontrol over lister med enumitem-pakken

<pre>\usepackage{enumitem} ----- slut preamble ----- \begin{enumerate}[label=\textbf{\Alph*}, ref=(--\Alph*--)] \item Punkt 1\label{a} \item Punkt 2 \end{enumerate} reference: \ref{a}</pre>	<p>(A) Punkt 1</p> <p>(B) Punkt 2</p> <p>reference: (–A–)</p>
--	---

Eksempel
9.46


Dvs. både klagepunkt (ii) og (iii) på modstående side er der nemt taget hånd om.

Tip 9.21. Nester man lister og man gerne vil have referencer til den indre liste til at afspejle den ydre, så er her et trick:

<pre>\usepackage{enumitem} ----- slut preamble ----- \begin{enumerate}[label=\arabic*] \item Bla \begin{enumerate}[label=(\alph*), ref=\arabic{enumi}{\alph*}] \item \label{item:1} bla \item bla \end{enumerate} \item bla \end{enumerate} En reference: \ref{item:1}</pre>	<p>1 Bla</p> <p style="padding-left: 2em;">(a) bla</p> <p style="padding-left: 2em;">(b) bla</p> <p>2 bla</p> <p>En reference: 1(a)</p>
---	---

Eksempel
9.47


Tælleren `enumi` holder værdien af tælleren i niveau 1, `enumii` i niveau 2, osv.

Man kan selv bestemme ved hvilket nummer en liste skal starte (nøgleordet »start«), man kan endda styre justeringen af tælleren via nøgleordet »align« (disse er normalt højrestillet, sammenlign med [Eksempel 9.46](#)).

start
align

<pre>\usepackage{enumitem} ----- slut preamble ----- \begin{enumerate}[label=\Roman*., start=5,align=left,] \item Punkt 1 \item Punkt 2 \end{enumerate}</pre>	<p>V. Punkt 1</p> <p>VI. Punkt 2</p>
--	--------------------------------------

Eksempel
9.48


Man kan desuden genstarte en liste via nøgleordet »resume« (igen version 2), hvis lige man skal skrive en bemærkning før man går videre med listen. Der er også en version uden »*«, men den bevarer ikke formateringen. Nøgleordet »noitemsep« kan anvendes til at fjerne den vertikale spacing.

*resume**

noitemsep

<pre>\usepackage{enumitem} ----- slut preamble ----- \begin{enumerate}[label=(\alph*), noitemsep] \item Punkt 1 \item Punkt 2 \end{enumerate} Noget tekst imellem \begin{enumerate}[resume*] \item Punkt 3 \item Punkt 4 \end{enumerate}</pre>	<p>(a) Punkt 1</p> <p>(b) Punkt 2</p> <p>Noget tekst imellem</p> <p>(c) Punkt 3</p> <p>(d) Punkt 4</p>
--	--

Eksempel
9.49


9. Mere om tekst

Lad os for god ordens skyld også komme med et eksempel hvor vi ændrer ved den horisontale spacing.

Eksempel 9.50



```
\usepackage{enumitem}
```

```
----- slut preamble -----
```

```
\begin{enumerate}[leftmargin=*
```

```
\item Punkt 1 \item Punkt 2
```

```
\end{enumerate}
```

Ny paragraf

```
\begin{enumerate}[leftmargin=*,
```

```
labelindent=\parindent,
```

```
label=\Roman*,widest=III, align=left
```

```
] \item 1 \item 2 \item 3
```

```
\end{enumerate} Tekst efter
```

1. Punkt 1

2. Punkt 2

Ny paragraf

I 1

II 2

III 3

Tekst efter

Bezos (2011) forklarer mere om hvordan den horisontale konfiguration fungerer, bemærk især eksemplerne i afsnit 5.

Lad os lige tage et eksempel med omkonfigurering af *description*.. Vi vil gerne have at hvis teksten vi beskriver, er for lang, så skal den stå på en linie for sig selv, således at beskrivelsen begynder på linien under, hvis teksten er kort nok, skal beskrivelsen starte på samme linie. Dette sker via nøgleordet *style*.

Eksempel 9.51



```
\usepackage{enumitem}
```

```
----- slut preamble -----
```

```
\begin{description}[style=nextline,
```

```
leftmargin=3.5em]
```

```
\item[kort] tekst tekst
```

```
\item[noget længere] tekst tekst
```

```
\end{description}
```

kort tekst tekst

noget længere

tekst tekst

Det bliver naturligvis ikke sjovt i længden hvis man for hver anvendelse af *enumerate*-environmentet skal skrive en hel masse konfiguration. Dette er der også tænkt på.

```
\setenumerate      \setenumerate[⟨niveau⟩]{⟨format⟩}
\setitemize        \setitemize[⟨niveau⟩]{⟨format⟩}
\setdescription    \setdescription{⟨format⟩}
\setlist           \setlist[⟨niveau⟩]{⟨format⟩}
```

Disse kan anvendes til at angive standard konfigurationer til de enkelte lister. `\setlist` vil påvirke alle tre lister, mens `\setNavn` kun påvirker den angivne liste. Dvs. ønsker man at fjerne afstanden mellem punkterne i *alle* lister så skal man bare anvende:

Eksempel 9.52

```
\usepackage{enumitem}
```

```
\setlist{noitemsep}
```

Da både *enumerate* og *itemize* kan nestes opstår behovet for *niveauer*. Man kan så angive konfigurationen af et givet niveau via det valgfri argument (angives via 1, 2, 3, 4).

Eksempel 9.53

```
\usepackage{enumitem}
```

```
\setenumerate[1]{label=\arabic*}
```

```
\setenumerate[2]{label=(\emph{\alph*})}
```

Man kan jo så anvende disse konfigurationsmuligheder til at lave sine egne lister med.

Tip 9.22. I denne bog har vi bl.a. anvendt

```
\setlist{
  listparindent=\parindent,
  parsep=0pt,
}
```

**Eksempel
9.54**

således at hvis et punkt i en liste indeholder mere end et afsnit, så anvendes der afsnitsindentering samt inden luft mellem afsnit (dermed passer dette med resten af designet).

Kort oversigt over enumitem parametre

For en liste over de mange forskellige nøgleord man anvender til konfigurationen vedrørende `enumitem`, henvises til [Bezoz \(2011\)](#).

9.7.3 Egne listekonstruktioner

Man kan desuden selv lave liste konstruktioner ved brug af indbyggede makroer som ikke behøver yderligere pakker. Dette ligger dog udenfor rammerne af denne bog, så læseren henvises i stedet til [Mittelbach og Goossens \(2004\)](#) afsnit 3.3.4 eller Memoir manualen [Wilson \(2010\)](#). Dog kan man oftest nøjes med at lave nye lister via `enumitem`-pakken.

9.8 Brug af farver

Vi har i et tidligere afsnit ([7.8.3](#) på side 193) snakket lidt om farver i tabeller. Vi vil her kort fortælle lidt om den generelle brug af farver i L^AT_EX. Da farvetryk er dyrt er det ikke noget vi vil gøre så meget ved, men man ved i det mindste hvor man skal kigge efter mere information.

Man anbefales at anvende pakken `xcolor`, den har en meget større fleksibilitet end standardpakken `color`. Man henvises til [Kern \(2006\)](#) for yderligere brugerinformation, helt specifikt om de mange muligheder for at definere nye farver, justere på eksisterende farver samt de forskellige farvemodeler. Man kan f.eks. blande farver, eller bede om komplementærfarver.

xcolor

Der er to basale makroer til at definere nye farvenavne

```
\definecolor{<navn>}{<farvemodel>}{<værdi>}
\colorlet{<nyt navn>}{<tidligere navn>}
```

Hvad `<farvemodel>` (f.eks. `rgb`, `RGB`, `gray`) eller `<værdi>`, er, kan man se i [Kern \(2006\)](#).

Tip 9.23. `\colorlet` er ganske smart, den angiver et nyt navn for en eksisterende farve (eller en blanding af en eksisterende farve). Hvis `<nyt navn>` findes i forvejen vil den blive overskrevet (lokalt). Dette kan være ganske nyttigt til lokalt at ændre på en farve. Vi bruger det f.eks. i bemærkning [9.24 på den følgende side](#), hvor baggrundsfarven af koden ændres så den passer bedre med farven i kassen.

Der er fire basale makroer man kan anvende til at få farver i sin tekst.

9. Mere om tekst

```
\color{<farvenavn>}
\textcolor{<farvenavn>}{<tekst>}
\colorbox{<farvenavn>}{<tekst>}
\fcolorbox{<randfarvenavn>}{<farvenavn>}{<tekst>}
```

`\color{<farve>}` opfører sig som f.eks. `\bfseries`, dvs. den ændrer tekstfarven herfra og frem, så her vil det være en god ide at huske at vide hvordan man beskytter sig. `\textcolor{<farve>}{<tekst>}` svarer til »`\color{<farve>}` tekst«.

`\colorbox` vil i stedet farve baggrunden af den angivne tekst – bør kun anvendes til korte tekster da makroen ikke tillader linieskift.

Har man et større område (flere linier) som skal have en farvet baggrund, så skal det ind i en boks, f.eks. en `\parbox`. `\fcolorbox` er som `\colorbox`, men bare med en ramme omkring. Tykkelsen af rammen styres normalt af `\fboxrule` (som også styrer `\fbox`) og luften mellem rammen og teksten styres af `\fboxsep`. Pakken `xcolor` giver desuden yderligere kontrol over disse.

Som en slutbemærkning kan siges at den røde farve som er anvendt i af denne bog er defineret via

Eksempel 9.55

```
\definecolor{nicered}{rgb}{.647,.129,.149}
```

Koden stammer fra Emil Hedevang Lohse (anvendt i hans utroligt flotte bachelor projekt, [Hedevang \(2004\)](#), som er skrevet med plain- \TeX).

Sidebemærkning 9.24. Som nævnt bliver man nødt til at skulle bruge en boks konstruktion (`\parbox` eller *minipage*) for at kunne få baggrundsfarve på flere linier med tekst. Problemet med dette er at det ikke kan deles over flere sider.

`framed`
`shaded` Skulle dette være nødvendigt kan man i stedet anvende pakken `framed` (direkte indbygget i *memoir*-klassen). Denne giver bl.a. environmentet *shaded* som giver en baggrundsfarve til teksten, og denne kasse deler fint over flere sider. Det er denne konstruktion vi anvender til f.eks. baggrunden i denne bemærkning.

`shadecolor` Environmentet *shaded* kræver at brugeren *selv* har defineret farven `shadecolor`. I denne bog har vi anvendt

Eksempel 9.56

```
\definecolor{shadecolor}{gray}{.87}
```

Det største problem med `framed` (alene og i *memoir*) er at det er svært at konfigurere for en almindelig bruger. I 2010 dukkede en ny pakke op på CTAN: `mdframed`. Den gør det samme men med et langt mere konfigurerbart interface. Det er den konstruktion vi nu anvender til at lave baggrund og ramme omkring kasser som denne vi er i nu. En anden fordel er at når kassen har ramme, og den deles over en side, så lukkes rammen ikke. Se [Daniel og Schubert \(2010\)](#) for mere info.

Lad os slutte dette afsnit med et farveeksempel.

Eksempel 9.57

```
\raggedright
\textcolor{nicered}{\bfseries Dette
bliver rødt},
\colorbox{blue}{\color{white} dette
er hvidt på blå baggrund},
\fcolorbox{red}{cyan}{ramme og baggrund}
```

Dette bliver rødt,
dette er hvidt på blå baggrund,
ramme og baggrund

9.9 Udsende mange breve

Vi har tidligere nævnt at man til breve kan anvende klasserne *letter* og *scr1tr2*. Vedrørende brugen af den sidste henvises til [Kohm og Morawski \(2006\)](#). (Bemærk iøvrigt at breve normalt ikke understøtter `\section` etc.)

I [Eksempel 9.58](#) finder man en meget simpel skabelon til *letter*.

```
\documentclass[a4paper]{letter}
% normal preamble
\usepackage[danish]{babel}
\begin{document}
\address{
Lars Madsen\
Institut for Matematiske Fag}
\signature{Lars Madsen}
\begin{letter}{En eller anden \
                Et eller andet sted}
\opening{Til rette vedkommende}

Noget tekst

\closing{M.v.h.}
\end{letter}
\end{document}
```

Eksempel

9.58



Ville egentlig bare nævne breve idet man med L^AT_EX, nemt kan lave mange næsten ens lydende breve. Meget nyttigt hvis man f.eks. er en forening som skal ansøge om fondsmidler til en studietur.

Dette kan løses via pakken *mailing*. De to ting man kan ændre er adressen man sender til (argumentet til `\begin{letter}`) samt argumentet til `\opening`. Resten af brevet forbliver ens for alle.

mailing

Man starter med at lave en adresse fil som skal have et bestemt format. Et eksempel kan ses i [Eksempel 9.59](#). Dvs. normalt består hver indgang af fire linier, »modtager

```
Navnet på modtageren
Vej\By\Land ...
Åbningslinie

Navnet på modtageren
Vej\By\Land ...
Åbningslinie

% end of records
```

Eksempel

9.59



navn«, »modtager adresse« (skal være på én linie hver), »åbningslinie« samt en »*blank*« linie. Det er en almindelig fejl at man glemmer den blanke linie efter sidste indgang, så husk den (et trick er at gøre som ovenfor og indsætte »% end of records« til slut så glemmer man den ikke). (Der sker ikke noget ved at man har mange blanke linier mellem indgangene.)

Så er resten faktisk simpelt.

9. Mere om tekst

Eksempel

9.60



```
\documentclass[a4paper]{letter}
% normal preamble
\usepackage{mailing}
\addressfile{adressefil.txt}
\begin{document}
% opsætning af brev variable
\address{Mit navn\ Min adresse}
\signature{Mit navn}
% teksten som anvendes i brevet, incl. mvh. linien (\closing)
\mailingtext{
  Dette er teksten til brevet

  \closing{M.v.h.}
}
\makemailing
\end{document}
```

Man kan desuden tilføje ekstra ting til adresse formatet, se evt. [Braams \(1994\)](#).



Har man brug for mere fleksibilitet, så kan man tage et kig på pakken `datatool` som kan håndtere CSV-filer. Den kan f.eks. loope over alle data linierne i CSV-filen og lade brugeren gøre noget ved data for hvert sæt.

I DK-TUG skulle vi lave et følgebrev til girokortet (som også er lavet ud fra behandling af CSV-data), her ville vi bede brugerne om at tjekke det emailadresse vi har til dem. Dette brev blev lavet nogenkunde som det ses i [Eksempel 9.61](#).

Eksempel

9.61

```
% oevrige preamble
\usepackage{datatool,url}
\pagestyle{empty}
\begin{document}
% format i CSV fil:
% "navn", "email"
% "Lars Madsen", "daleif@imf.au.dk"
% ...

\DTLloaddb{data}{mydata.csv}

% makro til at generere indhold af brev
\newcommand\mybrev[2]{%
% indhold af brevet
\begin{center}
  \begin{tabular}{l !{\quad} r}
    \toprule
    Navn & Emailadresse \\
    \midrule
    #1 & \expandafter\url\expandafter{#2} \\
    \bottomrule
  \end{tabular}
\end{center}
% resten af brevet
}

% loop over data (det er vigtigt med =email}, eller =email%, ellers
% fanger den ikke denne søjle rigtig
\DTLforeach{data}{%
  \navn=navn,\email=email}{%
  \mybrev{\navn}{\email}
  \newpage
}
\end{document}
```

Dette genererede 96 følgebrev, man kunne naturligvis også have lavet hele brevet inden i løkken. Mere information findes i [Talbot \(2009a\)](#).

9.10 Fremvisning af tekst uden L^AT_EX-fortolkning

Fremvisning af tekst uden L^AT_EX-fortolkning kan man også betegne som ordret (engl.: *verbatim*) fremvisning, som betyder at den anvendte tekst fremvises nøjagtigt som den er skrevet, alle mellemrum tæller, der er ingen automatiske lineskift, ingen automatisk orddeling, \ bliver vist etc.

Verbatim tekst i L^AT_EX kan faktisk være overraskende vanskeligt, men vi har heldigvis nogle redskaber ved hånden. Hvis man ønsker at fremvise en kort tekst i ordret form kan man anvende:

```
\verb<separator-tegn><tekst uden separator-tegn><samme separator-tegn> \verb
\verb*<separator-tegn><tekst uden separator-tegn><samme separator-tegn> \verb*
```

Her betyder `<sep.-tegn>` (eks.: `+`) at `\verb` skal skrive teksten fra `<sep.-tegn>` til næste forekomst af `<sep.-tegn>`, ordret. Se [Eksempel 9.62](#).

Sidebemærkning 9.25. De to `<sep.-tegn>` skal stå på samme linie som `\verb` ! og de to tegn skal naturligvis være ens.

Den *-ede udgave vil markere mellemrum som `»_«`.

Sidebemærkning 9.26. `\verb` må *ikke* anvendes i definitionen af en makro eller som argument til en makro. Dog kan den oftest sagtens anvendes inden i et environment.

Det vil sige at `\verb` må f.eks. ikke anvendes i en fodnote. Se pakken `fancyvrb` for lidt hjælp til hvad man så kan gøre hvis man skulle få brug for det. Dog skal det nævnes at pakken `bigfoot` tillader brugen af `\verb` i fodnoter.

Større tekststykker kan fremvises verbatimt via de to environments ***verbatim*** og ***verbatim****.

Sidebemærkning 9.27. `\begin` og `\end` delene for begge environments skal stå på linier for sig selv og der må ikke forekomme mellemrum før `\begin/end`. Desuden bør disse ikke anvendes inden i definitioner af andre environments.⁶

Her er et eksempel på brugen af både den korte form og environment udgaven.

```
\LaTeX skrives via \verb+\LaTeX+\
\verb*|tekst med mellemrum|
\begin{verbatim}
noget verbatim tekst
over flere linier
\end{verbatim}
```

```
LATEX skrives via \LaTeX
tekst_med_mellemrum
noget verbatim tekst
over flere linier
```

**Eksempel
9.62**



⁶ Vi nævnte jo at verbatim tekst involverer meget skumle konstruktioner.



Farlig (læs teksten) 9.28. Det man oftest får brug for at kunne, er at inkludere indholdet af en fil og fremvise dette ordret. Den naive skribent ville sikkert forsøge sig med:

Eksempel 9.63

```
\begin{verbatim}
\input{filnavn}
\end{verbatim}
```

– hvilket man så selv kan overveje hvorfor ikke vil virke.⁷

verbatim Den simpleste måde at inkludere en fil på denne måde er via pakken *verbatim*, hvorefter man anvender⁸

\verbatiminput `\verbatiminput{⟨filnavn⟩}`

Det inkluderede bliver så skrevet med typewriter-fonten, hvilket er tradition vedrørende den slags tekst, den kan lokalt tilgås via `\texttt{⟨tekst⟩}`. Opsætningen af fonten vil normalt være sådan at alle bogstaver og mellemrum fylder det samme. Man skal dog bemærke at teksten ofte bliver ret så bred når man anvender typewriter-fonten, derfor kan det være en god ide at lave sin egen inkluderings makro.

Kodetip 9.2

```
\newcommand\sasoutputfont{\small}
\newcommand\sasoutput[1]{%
\begin{flushleft}
\sasoutputfont
\verbatiminput{#1}
\end{flushleft}
}
```

flushleft Den eneste grund til at anvende *flushleft*-environmentet i dette tilfælde er at den så automatisk får noget luft før og efter inkluderingen. En anden konstruktion kan laves via *listings*-pakken. I afsnit 9.10.2 på side 252 omtales en pakke undertegnede har lavet specifikt til inkludering af SAS-kildekode og *-output*, den gør netop brug af *listings*-pakken.

9.10.1 Kildekode fremvisning

fancyvrb listings

Kildekode fremvisning kan selvfølgelig laves via en *verbatim*-konstruktion, men det er findes endnu smartere løsninger. De mest interessante er sandsynligvis pakkerne *fancyvrb* og især *listings*. Begge pakker ligger udenfor rammerne af denne bog så vi vil i stedet bare komme med et kort *listings* eksempel og så henvise til de respektive manualer (Van Zandt (1998) og Heinz (2004)).

Sidebemærkning 9.29. Eksemplerne i denne bog er lavet via *listings*. Til at starte med var de dog lavet med både *listings* og *fancyvrb*, men seneste version af *listings* gjorde *fancyvrb* overflødig (hvis man lige var en smule snedig).

Listings er så, kombineret med en masse hjemmebrygget kode, blevet til den pakke vi anvender til at lave eksemplerne.

⁷ Har faktisk set folk undre sig over hvorfor dette ikke virker, og en vred bruger skrev en mail om at man ikke kunne tillade sig at skrive eksempler som er forkerte. Derfor er disse eksempler nu markeret.

⁸ Makroen er direkte inkluderet i *memoir*-klassen.

9.10.2 Lidt om listings

Pakken `listings` virker lidt som *verbatim* men kan nu konfigureres ud i det ekstreme. Den er specielt god til at tage sig af fremvisnings af kode fra forskellige programmeringssprog, inkluderer bl.a. muligheder for automatisk at indeksere nøgleord etc. Den er specielt god til at hente koden ind fra eksterne filer og så f.eks. kun vise visse linier.

listings

I [Eksempel 9.64](#) følger et simpelt eksempel på fremvisning af et Perl-script. Vi har skiftet fonten i eksemplet til Computer Modern for at kunne understrege en *pointe* som følger senere i teksten.

```
\usepackage{listings}
\lstset{extendedchars=true,          basicstyle=\ttfamily,
        keywordstyle=\normalfont\bfseries, columns=flexible,
        numbers=left,                numberstyle=\tiny,
        breaklines=true,             breakatwhitespace=true,
        language=Perl}

----- slut preamble -----

\begin{lstlisting}
#!/usr/bin/perl
# epstopdf kan ikke klare mere end en fil ad gangen derfor denne wrapper
# anvendelse: EpsToPdf *.eps
foreach $l (@ARGV) {
    system "epstopdf $l";
}
\end{lstlisting}
```

Eksempel 9.64



```
1  #!/usr/bin/perl
2  # epstopdf kan ikke klare mere end en fil ad gangen derfor denne wrapper
3  # anvendelse: EpsToPdf *.eps
4  foreach $l (@ARGV) {
5      system "epstopdf $l";
6  }
```

[Eksempel 9.64](#) viser forskellige ting:

- Man kan styre hvordan forskellige dele af teksten skal skrives (prettyprint), se `#!`-linien samt se to nøgleord, `foreach` og `system`.
- Man har mulighed for at sætte linienumre på.
- Man kan specificere hvilket programmeringssprog som kodestumpen er skrevet i, på denne måde ved L^AT_EX hvilke ord der skal betragtes som nøgleord.
- Systemet kan automatisk bryde linierne hvis de bliver for lange, uden at dette påvirker linienumrene.

Pakken `listings` kan ufatteligt meget mere og er yderst konfigurerbar. Den interesserede læser henvises til [Heinz \(2004\)](#). F.eks. kan man i stedet for at skrive det hele, bare inkludere en fil, via `\lstinputlisting{<filnavn>}` og man kan endda få vist kun nogle af linierne i denne fil (meget nyttigt hvis man er ved at forklare forløbet af implementeringen af et program).

Som det ses af [Eksempel 9.64](#) har vi fremhævet visse nøgle ord i det lille PERL-script. Vi blev her nødt til at anvende den almindelige fede font idet Computer Modern Typewriter ikke findes i en fed version. Har man anvendt en mørkere tekstfont, eksem-

9. Mere om tekst

pelvis Palatino eller Utopia kan man med held anvende LuxiMono⁹ som også er en typewriter-font men som også findes i f.eks. fed, se også afsnit 9.6.3 på side 236. Det er vigtigt at man husker fontenc med T1-option. Se [Eksempel 9.65](#).

Eksempel 9.65



```
\usepackage{mathpazo}
\usepackage[scaled=0.8]{luximono}
\usepackage[T1]{fontenc}
\usepackage{listings}
\lstset{extendedchars=true,          basicstyle=\ttfamily,
        keywordstyle=\normalfont\bfseries\ttfamily, columns=flexible,
        numbers=left,                numberstyle=\tiny,
        breaklines=true,             breakatwhitespace=true,
        language=Perl}

----- slut preamble -----

\begin{lstlisting}
#!/usr/bin/perl
# epstopdf kan ikke klare mere end en fil ad gangen derfor denne wrapper
# anvendelse: EpsToPdf *.eps
foreach $l (@ARGV) {
  system "epstopdf $l";
}
\end{lstlisting}
```

```
1  #!/usr/bin/perl
2  # epstopdf kan ikke klare mere end en fil ad gangen derfor denne wrapper
3  # anvendelse: EpsToPdf *.eps
4  foreach $l (@ARGV) {
5    system "epstopdf_$l";
6  }
```

Sidebemærkning 9.30. Anvender man justeringen

Eksempel 9.66

```
\lstset{
  extendedchars=true,
  basicstyle=\ttfamily,
  columns=fixed
}
```

så får man hvad der svarer til *verbatim*-environmentet, i.e. man behøver kun angive sprog hvis man ønsker det.

Opsætning til SAS-udskrifter via listings

Vi har allerede nævnt at programmører bør kigge i [Heinz \(2004\)](#) for mere information omkring listings. Indenfor matematik vil det sikkert ofte være til sas man får brug for at kunne fremvise noget kode eller output. Til dette formål har jeg til en af IMF's VIP medarbejdere lavet en pakke (SASdisplay) som er lidt listings konfiguration for sas samlet til en pakke. Pakken er offentligt tilgængelig via

SASdisplay

<http://www.imf.au.dk/system/latex/lokale-pakker/>

⁹ Pakken luximono samt selve fontfilerne kan findes på CTAN, MiKTeX-brugere kan installere den direkte via *Package Manager*, TeX Live-brugere kan anvende programmet *getnonfreefonts*, se <http://tug.org/fonts/getnonfreefonts/>, siden forklarer hvordan man henter og installerer *getnonfreefonts*, herefter kan man installere via f.eks. »*getnonfreefonts-sys -a*«.

Stikordsregister

hvor man vil finde selve pakken, en konfigurationsfil med flere keywords samt en manual. Man skal selv sørge for at installere pakken, nemmeste er bare at lægge ».sty« og ».cfg« filen i samme bibliotek som filen hvori man indlæser SASdisplay. Se [Eksempel 9.67](#), som kræver SASdisplay version 0.81.

```
\usepackage[frame=lineswithbackground]{SASdisplay}
\renewcommand\ttdefault{txtt}
```

— slut preamble —

```
\begin{SAScode}
  * hest  $a^2+1$  ;#
  PROC INSIGHT DATA data=fisk;
  SCATTER x1 x2 x3 x4 x5 * dosis vgt;
  RUN;
  * dette er en kommentar ;#
  OUTPUT
  QUIT; /* en kommentar i SAS */
\end{SAScode}
```

Eksempel

9.67



SAS-kode 1

```
* hest  $a^2+1$  ;
PROC INSIGHT DATA data=fisk;
SCATTER x1 x2 x3 x4 x5 * dosis vgt;
RUN;
* dette er en kommentar ;
OUTPUT
QUIT; /* en kommentar i SAS */
```

Bemærk hvad der sker mellem hvert par af #-er. Pakken inkluderer desuden forskellige konfigurationsmuligheder, alt sammen forklaret i [Madsen \(2007\)](#). I [Kodetip 10.3 på side 281](#) kan man se hvordan man kan anvende SASdisplay konfigurationen til at lave noget tilsvarende til R-kode.

Sidebemærkning 9.31. Er i gang med at skrive en erstatning til SASdisplay, som skal understøtte både SAS- og R-kode. Hvornår denne pakke bliver klar vides endnu ikke.

9.11 Stikordsregister

Et godt stikordsregister/indeks kan være en stor gavn til læseren samt for forfatteren, når han/hun ikke lige kan huske hvor en bestemt ting er beskrevet i bogen. Men et indeks er heller ikke bedre end forlæggert, det kræver disciplin og viden for at lave et godt indeks. Der er faktisk firmaer som lever af at lave fornuftige indekser til andres bøger. I dette afsnit vil vi beskrive hvordan man laver et indeks i L^AT_EX.

9.11.1 Generere indeksindgangene

At skrive indeksindgange i L^AT_EX, er egentlig ret problemfrit og skriver man på engelsk vil der sikkert ikke opstå (alvorlige) problemer. Desværre forholder det sig ikke sådan med andre sprog. Dette kommer vi tilbage til senere.

Hvis man *ikke* anvender *memoir*-klassen (eller lignende), skal man lige lægge ud med at inkludere pakken `makeidx`. I alle tilfælde skal man starte med at placere følgende makro i sin preamble

`makeidx`

9. Mere om tekst

`\makeindex` `\makeindex`

Dette sørger for at åbne filen $\langle masterfil \rangle$.idx hvor alle indekxinformationerne skrives til, uden `\makeindex` får man ikke indsamlet nogen indekseringsdata (brug af `\index`-kommandoen ignoreres helt).

Selve indekxindgangen skrives (meget naturligt) med kommandoen

`\index` `\index{\langle tekst \rangle}`

denne vil så sørge for at $\langle tekst \rangle$ samt det aktuelle sidetal skrives til $\langle masterfil \rangle$.idx.

Der gælder forskellige regler for fortolkningen af indholdet af $\langle tekst \rangle$:

»!« angiver at det som kommer efter ! er et *underpunkt* af det som kom før »!«. Man har normalt kun ned til to underniveauer, i.e. anvend maksimalt to !-er.

Eksempel `»\index{Dyr!Hest}«`, her indekseres »Hest« som et underpunkt af »Dyr«.

»@« Til tider ønsker man at en indekxindgang skal enten sorteres anderledes end det skrives, eller man ønsker at formatere indgangen anderledes. Til dette formål anvendes syntaksen $\langle til\ sortering \rangle @ \langle formateret\ indgang \rangle$. Eksempel: `\index{tekst@textit{tekst}}`. I denne bog anvender vi dette til at indekxere L^AT_EX-makronavne mens vi ignorerer »\«.

Hvis frasen, man ønsker at indekxere, bare skal stå som almindelig tekst, så skriver man naturligvis bare `»\index{frase}«` uden brug af »@«.

»|« anvendes i formen `\index{...|textit}` (bemærk `textit` uden »\«), og bevirker i dette tilfælde at *sidetallet* som optræder i indekset nu kommer ud i kursiv. Man kan også anvende andre markoer efter »|« f.eks. `\see`.¹⁰

`\see`

(speciel)

Har man brug for at skrive »!«, »@« eller »|« i en indekxindgang skal kan de *escapes* via »"!«, »"@« eller »"|« (altså anførselstegn fra tastaturet foran det man ønsker at escape).

Sidebemærkning 9.32. Er det et specielt ord man ønsker at indekxere ved at placere en `\index{\langle ... \rangle}` efter det, så skal man huske at det *ikke* bør være mellemrum før `\index{\langle ... \rangle}`, ellers kan man risikere at få et forkert sidetal.

Man kan sagtens lave mange indekxinger efter hinanden ved bare at skrive flere `\index` kommandoer, altså `»ord\index{...}\index{...}...«`. Til at hjælpe med indekx i denne bog anvendes pakken `dlfltxmarkup` og min specifikke konfigurationspakke til denne `dlfltxmarkupbookkeys`. Dette giver mig mulighed for f.eks. at skrive `\markup[nomk,sty]{pakke}` som så bl.a. vil indekxere »pakke« som underpunkt til en liste over pakker, samt som et punkt for sig selv.

På det sted hvor man ønsker at få indsat sit indekx, skal man placere makroen

`\printindex` `\printindex`

(det kan sammenlignes med `\bibliography` til litteraturlister lavet med B_IB_TE_X). Denne makro vil så inkludere indholdet af `»\langle masterfil \rangle.ind«` hvis den findes. `» .ind«`-filen er

¹⁰ Denne feature virker pt. ikke ordentligt hvis man anvender hyperref i kombination med indekxeringsprogrammet xindy.

Sortere selve indekset

den sorterede udgave af den tidligere omtalte $\langle masterfil \rangle$.idx-fil, og skal fremstilles af et specielt eksterne program, se afsnit 9.11.2.

Tip 9.33. I indekset er data i bogstavgrupper, først kommer symboler (dvs. ting som i indekseringsnøglen ikke starter med a–z), så A, B, ... Mellem hver gruppe bliver der indsat noget luft, via $\backslash indexspace$, hvis størrelse er defineret af dokumentklassen. Vi vil gerne have at der mindst skal komme to punkter efter overskriften, hvis der ikke er plads til det, er det bedre at lave søjle skift før overskriften.

Dette man vi gøre ved at pille lidt ved $\backslash indexspace$:

```
\let\oldindexspace\indexspace
\renewcommand\indexspace{\oldindexspace\needspace{3\baselineskip}}
```

hvor $\backslash needspace$ kommer fra *needspace*-pakken (eller *memoir*-klassen). I dette tilfælde beder vi om at der skal være plads til tre linier efter $\backslash indexspace$, overskriften samt de to første punkter.

$\backslash indexspace$

Eksempel 9.68

$\backslash needspace$
needspace

9.11.2 Sortere selve indekset

L^AT_EX kan ikke selv sortere og sammensætte selve indekset, det anvender man et eksternt program til at gøre for sig.

Der findes forskellige programmer til at generere indekset med:

makeindex

dette er det klassiske indekseringsprogram til L^AT_EX.

Fordele: Programmet følger med alle L^AT_EX-installationer.

Ulemper: Kan ikke sortere på dansk. Kræver at man i $\backslash index\langle tekst \rangle$ ikke har noget lineskift i $\langle tekst \rangle$ (for så får man i stedet to forskellige indgange). Skal omkonfigureres for at få overskrifter til hver bogstavgruppe.

xindy

giver gennem programmet *texindy* et mere moderne indekseringsprogram.

texindy

Fordele: Kan sortere efter sprogreglerne på mange forskellige sprog, endda på flere måder på enkelte sprog (eks. tysk til tyskere og tysk til ikke-tyskere). I standardopsætningen vil den automatisk lave overskrifter til bogstavgrupperne.

I modsætning til *makeindex*, så er denne ikke sårbar over for multiple mellemrum i det man ønsker at indekse. Dette sørger den selv for at fikse således at »hest_nisse« og »hest_nisse« er den samme indgang i indekset.

Ulemper: Har indtil fornyligt kun været tilgængelig på Linux, men findes nu til alle platforme, dog kun hvis man anvender T_EX Live 2008+. Understøtter ikke konfigurationsfilerne til *makeindex*, og kan være en smule underlig selv at konfigurere. Er ikke altid velegnet til at sortere indeks til bøger om L^AT_EX (thi den sorterer bl.a. \$ og \ \$ til \ \$, som er fint til almindelige bøger, men ikke altid velegnet til bøger om L^AT_EX).

mkind

Program skrevet af Jørgen Larsen og kan downloades fra <http://dirac.ruc.dk/imfufalateX/>. Programmet findes til Windows og Linux og udemærker sig ved at være et indekseringsprogram som tager hensyn til de danske regler.

Personligt har jeg ikke nogen erfaring med programmet.

9. Mere om tekst

Ulempe: Anvender en lidt anderledes syntaks (som man dog også kan konfigurere `makeindex` til at anvende), se [Larsen \(2010\)](#).



Hvornår skal man så køre indekseringsprogrammet? Tja, det skal man når der er tilføjet nye indeksindgange, eller når sidetallet er ændret. Det nemmeste er at man sørger for at huske at køre indekseringsprogrammet når man er færdig med at skrive på dokumentet, husk også at oversætte L^AT_EX-dokumentet nogle gange bagefter.

latexmk

Alternativt kan man anvende programmer som selv kan finde ud af at køre f.eks. `makeindex` og `bibtex` det gange det er nødvendigt. Jeg anvender selv anvendes Perl-programmet `latexmk` som også fungerer på Windows og som nu følger med T_EX Live. Her kigger man på `.idx`-filen, mellem hvert gennemløb, hvis den er ændret (f.eks. anderledes sidetal), så skal der køres `makeindex` og `latex` igen.

Sortering via `makeindex`

For at få den sorterede indeksfil med `makeindex` er syntaksen:

```
makeindex <options> masterfil
```

Bemærk der skal ikke angives nogen filtype. `<options>` kan f.eks. anvendes til at konfigurere hvordan indekset skal se ud, det er dog ikke noget vi til gå i detaljer med her, se i stedet [Mittelbach og Goossens \(2004\)](#) eller [Wilson \(2007a\)](#), se dog [Eksempel 9.69 på næste side](#).

De fleste moderne L^AT_EX-editorer har en knap eller lignende¹¹ til at køre `makeindex` med (skal man bruge ekstra `options`, skal man sikkert omkonfigurere sin editor en smule). Alternativt kan man udføre `makeindex` via kommandolinien eller en terminal.

Sidebemærkning 9.34. I afsnit [9.11.2 på forrige side](#) beskrev vi nogle problemer vedr. `makeindex`, her er et hint til hvad man så kan gøre.

- (i) `makeindex` kan *ikke* sortere på dansk! Den opfatter de danske vokaler som makroer i stedet for bogstaver, dermed bliver »sø« sorteret før »sa«. Man har umiddelbart tre alternativer hvis man løber ind i problemer af denne type:
- Skriv en alternativ sorterings nøgle (dette kræver god disciplin), f.eks. `\index{sz@sø}`.
 - Man kan anvende programmet `mkind` af Jørgen Larsen. Det findes til både Linux og Windows, via <http://dirac.ruc.dk/imfufalates/>, dog anvender den en lidt anderledes syntaks (som man dog også kan konfigurere `makeindex` til at anvende), se [Larsen \(2010\)](#). Eller man kan anvende `xindy`.
- (ii) For `makeindex` har *alle* tegn betydning. Dvs. har man anvendt `\index{tekst_}`, `\index{ _tekst}`, `\index{tekst}` eller `\index{tekst <linieskift> tekst}` og `\index{tekst_tekst}`, så giver dette fem(!) indeks indgange i stedet for de naturlige to indgange. Både `mkind` og `xindy` har ikke dette problem.

mkind

Kigger man på indekset i denne bog, så vil man se at der er overskrifter når en ny bogstavsgruppe starter. Dette vil `makeindex` *ikke* gøre som standard (det gør `xindy`

¹¹ Editoren TeXnicCenter vil som default køre både B_IB_TE_X og `makeindex` hver gang man oversætter.

Sortering via texindy

til gengæld), men man kan nemt konfigurere `makeindex` til det. Gem følgende som `minstil.ist`:

```
headings_flag 1
heading_prefix "{\\bfseries "
heading_suffix " }\\nopagebreak\\n"
```

Eksempel
9.69

Ønsker man at anvende syntaksen fra `mkind` sammen med `makeindex` så skal man tilføje linierne

```
actual '='
quote ''
level '>'
```

Eksempel
9.70

Nærmere forklaring kan findes i [Mittelbach og Goossens \(2004\)](#), afsnit 11.2.4, samt [Wilson \(2007a\)](#), afsnit 4.5. En `makeindex`-konfigurationsfil anvendes så som

```
makeindex -s minstil.ist masterfil
```

eller hvad nu din `«.ist«`-fil hedder.

Sortering via texindy

Skulle man være så heldig at have adgang til `texindy` (f.eks. via `TEX Live 2008+`), så anbefales (kraftigt) det at man anvender den i stedet for `makeindex`.

For at sortere at indeks på dansk¹² er det nu bare at køre

```
texindy -L danish masterfil.idx
```

Bemærk at endelsen `«.idx«` skal med. `texindy` vil automatisk sørge for at hver ny bogstavgruppe i indekset får en overskrift. *texindy*

Personligt har jeg anvendt `texindy` i flere år, denne bog er naturligvis også indekseret med den (via en omkonfigureret `latexmk`). Desuden tvinger jeg rutinemæssigt de skribenter jeg yder `LATEX`-support for til at anvende `texindy` netop for at undgå problemer med indekset.

I denne bog har vi dog fusket en del med indekset, vi ville gerne have et ordbogsagtigt sidehoved, samt at kunne se når et hovedord fortsætter fra den forrige side.

9.12 Terminologiforklaring

Til længere projekter som kunne forventes læst af andre end de som er direkte inden i emnet, så er det en god ide at lave sig en terminologiforklaring et eller andet sted i dokumentet. Terminologi kan i dette tilfælde være både symbolterminologi samt specielle ord eller vendinger som man gerne ville have at læseren hurtigt kunne så op hvis han/hun blev i tvivl om betydningen.

Vedrørende symbolforklaringer vil det især være en ide at medtage en sådan hvis man anvender symbolsammensætninger som ikke er helt standard, f.eks. brugen af `»~«` i statistik.

Sådanne oversigter kan laves på forskellig vis, lige fra det meget simple håndlavede til det meget avancerede hvor man anvender indekseringsprogrammer til at holde styr på tingene og sortere data.

¹² Gæt selv hvad der skal ændres for at sortere på et andet sprog.

9. Mere om tekst

9.12.1 Meget simpel symbolliste

Dette vil vi lave med en *longtable* hvor vi sætte første søjle til automatisk at være i matematik-mode. Skulle man få brug for at skrive et ord ind på listen kan den skrives med `\text`.

Eksempel 9.71



```
\usepackage{longtable,array}
\usepackage{amssymb,amsmath}

----- slut preamble -----

\section*{Symboloversigt}
% brug \chapter* i f.eks. memoir
%\phantomsection% hyperref
\addcontentsline{toc}{section}{Symboloversigt}
\begin{longtable}@{} >{${}l<{${} p{10cm} @{} }
% konfigurering
\multicolumn{2}{r}{\textit{fortsætter på næste
side}}\}
\endfoot
\endlastfoot
\mathbb{N} & De naturlige tal & \\\
\mathbb{Z} & De hele tal & \\\
\mathbb{Q} & De rationelle tal & \\\
\mathbb{R} & De reelle tal & \\\
\sim & Fordelt som, i.e. & \\
& $X\sim N(0,1)$ & \\\
\text{hvis} & \emph{Hvis og} & \\
& kun hvis & \\\
\end{longtable}
```

Symboloversigt

\mathbb{N}	De naturlige tal
\mathbb{Z}	De hele tal
\mathbb{Q}	De rationelle tal
\mathbb{R}	De reelle tal
\sim	Fordelt som, i.e. $X \sim N(0,1)$.
hvis	<i>Hvis og kun hvis</i>

Ulempen med [Eksempel 9.71](#) er blandt andet at man ikke har mulighed for at se hvor er er vigtig information vedr. dette ord.

9.12.2 Ikke helt så simpelt

Ønsker man f.eks. sidetal til siden hvor man har defineret eller forklaret noget specifikt, så skal vi ud i noget mere avanceret, se næste afsnit.

9.13 Terminologi- eller tegnforklaring (glossaries-pakken)

På engelsk kaldes dette ofte *glossary* eller *nomenclature*. Der er flere pakker som kan hjælpe, f.eks. har *memoir*-klassen sine egne indbyggede features til at lave et glossary (se [Wilson, 2010](#)). Et alternativ er pakken *nomencl* (se [Netherton og Radhakrishnan, 2005](#)). Fælles for begge løsninger er at de ligner meget metoden bag indekser. Altså at man skriver forklaringerne som en del af teksten, og så samler data via `makeindex`.

Dette har én stor ulempe. Tit vil man gerne flere gange kunne henvise til de (vigtige) steder hvor et forklaret koncept er anvendt i teksten, hvilket er bøvlet når forklaringen selv er en del af teksten. Man skal passe meget på hvad man gør.



glossaries

Vi vil i stedet anbefale at man anvender pakken *glossaries*. Denne pakke angriber opgaven på en anderledes, mere overskuelig, måde. Indenfor *glossaries*-pakkens univers lader man sig inspirere af metoderne bag litteraturlisten. Altså at man har sine data liggende ét sted, og dette henviser man så til på forskelligvis, f.eks. ved at hver gang man skriver et bestemt nøgleord i teksten, så gør man det via en speciel kommando. Denne kommando sørger så for at brugen bliver markeret således at terminologilisten kan skrive at dette nøgleord er anvendt på den og den side.

Data til ordlisten

Tip 9.35. Har man problemer med glossaries, får man normalt hurtig og kompetent hjælp på enten nyhedsgruppen `comp.text.tex`, <http://www.latex-community.org> eller <http://tex.stackexchange.com>. På CTT er det ofte forfatteren (Nicola Talbot) som svarer.

Anbefaling 9.36. For at få selve terminologilisterne, skal man, som ved indekser, anvende et indekseringsprogram, som så skal køres med forskellige options. Da man kan lave flere forskellige ordlister (hvilket vi ikke kommer nærmere ind på), kan man komme ud for at skulle køre indekseringsprogrammet på flere filer, hvilket hurtigt kan blive ret uoverskueligt.

Heldigvis har forfatteren til glossaries skrevet et program `makeglossaries` som kan hjælpe. Dette program er skrevet i sproget Perl. Er man på Mac eller Linux, så har man allerede Perl, og kan derfor anvende `makeglossaries` uden problemer. Er man på Windows og anvender T_EX Live 2009 eller nyere, har man heller ikke problemer, for disse versioner af T_EX Live kommer med Perl.

Er man derimod MikTeX-bruger, så har man ikke nødvendigvis Perl. I så fald skal man enten selv installere Perl, eller køre indekseringsprogrammerne manuelt. Vi anbefaler at man selv installerer Perl, se <http://www.perl.org/get.html>. Det er ret nemt, og bare Perl er installeret skulle man kunne anvende `makeglossaries` uden problemer. Syntaksen er så bare

```
makeglossaries <masternavn>
```

makeglossaries

9.13.1 Data til ordlisten

Man starter med at skrive sine ordlisteindgange. Vi anbefaler at man anvender en ekstern fil til dette. Denne fil skal så indlæses og dette *skal gøres før* man beder dokumentet om at forberede sig på en terminologiliste. Man skriver

```
\loadglsentries{glossary}  
\makeglossaries
```

**Eksempel
9.72**

hvor vi her har lagt indgangene i den eksterne fil `glossary.tex`.

Det man oftest vil skrive som data kommer i to typer: Ordforklaringer samt akronymer (forkortelser, som f.eks. HTTP). Det sidste er normalt meget nemt

```
\newacronym{<nøgle>}{<kort>}{<lang>}
```

Hvor man så vil få `<lang>` (`<kort>`) første gang man anvender `<nøgle>` og `<kort>` de efterfølgende gange. Normalt kommer disse bare med blandt punkterne i terminologilisten, men har man mange, kan man også bede om at få dem for sig selv, se Talbot (2009b). En mere kompliceret opsætning kan opnåes med den generelle makro `\newglossaryentry` med en bestemt type, se Talbot (2009b), afsnit 2.10.

Som et eksempel på en forkortelse kunne vi anvende

```
\newacronym{nato}{NATO}{North Atlantic Treaty Organisation}
```

**Eksempel
9.73**

Hvordan det anvendes i teksten forklarer vi i afsnit 9.13.2 på side 261.

¹² Anvend evt. Google Groups.

9. Mere om tekst



I de fleste tilfælde vil man i stedet anvende den mere generelle macro

```
\newglossaryentry{⟨nøgle⟩}{⟨data⟩}
```

Hvor *⟨nøgle⟩* skal ses på samme måde som ved citeringer. *⟨data⟩*-delen er bygget op som en liste af `key = {value}` punkter (man bør huske `{}` omkring værdierne). Her lister vi nogle af dem, en længere liste kan ses i (Talbot, 2009b, afsnit 2.2).

name = *⟨data⟩*

Dette er teksten som kommer til at stå i selve ordlisten. Hvis ikke **text**-nøglen er angivet, så er det også denne tekst som skrives i teksten når dette *ord* anvendes.

description = *⟨data⟩*

Er selve beskrivelsen af hvad dette nu er for noget. Hvis punktet af en eller anden grund ikke skal have nogen beskrivelse så skriv »\nopostdesc«.

parent = *⟨data⟩*

Man kan have underpunkter til punkter i ordlisten. Man skal blot huske at forældren *skal* være defineret før børnene. Det er så *⟨nøgle⟩* for forældren man angiver her.

text = *⟨data⟩*

Angiver den tekst som skal skrives i teksten når *⟨nøgle⟩* anvendes (sammenlign med det som skrives når man anvender `\cite`-kommandoen). Hvis punktet som *⟨nøgle⟩* henviser til ikke har angivet **text** anvendes teksten fra **name** i stedet.

first = *⟨data⟩*

Angiver hvad der skal stå første gang vi anvender dette ord i teksten, hvis det skal være anderledes end det fra **text**. Anden gang får man så teksten fra **text**. Denne feature kan *nulstilles* undervejs.

symbol = *⟨data⟩*

Et ord kan være associeret med et symbol. (Det afhænger af konfigurationen af ordlisten om symbolet vises.)

sort = *⟨data⟩*

Angiver hvordan dette skal sorteres. Antag f.eks. vores ord er »*p*-adisk«, her skrives *p* i matematik, men vi vil gerne have den sorteret under »*p*«. Her ville man så anvende

Eksempel 9.74

```
text = {$p$-adisk},  
sort = {p-adisk}
```

Nøglen **sort** skal desuden anvendes i forbindelse med underpunkter til at styre rækkefølgen af disse (se Talbot, 2009b, afsnit 2.2.2).

Som sagt er der også andre ting man kan angive, bl.a. diverse flertalsformer.

Sidebemærkning 9.37. Pakken `glossaries` har forskellige makroer til at anvende et ord i teksten (hvilket vi forklarer i det følgende afsnit), en af mulighederne er at man kan bede om at få angivet et ord med stort begyndelsesbogstav selv om ordet er defineret med små bogstaver (hvilket de fleste ord bør).

Dette kan give problemer når man anvender ord som starter med f.eks. æ, ø, å eller bogstaver med accent. Her anbefaler manualen at man husker at skrive f.eks.

```
name = {\æ}ble}
```

altså at man husker at sætte `{}` omkring dette startbogstav. For så kan makroerne fange det korrekt og gøre det til et stort bogstav.¹³

Eksempel
9.75

Som et eksempel på en indgang i en ordliste kan vi lave en for \LaTeX :

```
\newglossaryentry{latex}{
  name = {\LaTeX},
  sort = {LaTeX},
  description = {typesettingssystem byggende på \TeX}
}
```

Eksempel
9.76

9.13.2 Brug i teksten

Der er en hel del makroer man kan anvende i forbindelse med `glossaries`-pakken, vi forklarer kun nogle få her, og henviser ellers til manualen (Talbot, 2009b).

`\gls[options]{nøgle}`

Dette er den primære makro til *anvendelse* af et nøgleord. Den skriver så den tekst man har valgt for nøgleordet, samt markerer at ordet er anvendt på netop denne side.

`\glslink[options]{nøgle}{tekst}`

Dette minder mere om et link i HTML. Her skrives teksten `<tekst>` og `<nøgle>` markeres som anvendt på denne side. Hvis man anvender `hyperref` vil `<tekst>` blive et link til `<nøgle>`s indgang i ordlisten. Der er en stjernet version som ikke laver et link.

Bemærk at `\glslink` ikke påvirker første brug af `<nøgle>`. Ønsker man med denne lejlighed at markere `<nøgle>` som anvendt, skal man anvende `\glsdisp` i stedet.

`\addgls[options]{nøgle}`

Markerer at ordet er anvendt på denne side, men skriver ikke noget.

`\Gls[options]{nøgle}`

Som `\gls` men starter med stort bogstav.

`\glspl[options]{nøgle}`

Flertalsversion, hvis der ikke er defineret en flertalsversion, så forsøger den selv at lave en.

`\glsaddall`

Dette markerer alle punkter i listen, således at man kan tjekke formateringen af sin liste uden at skulle have anvendt alle nøgler.

¹³ Hvad man lige skal gøre vedr. *p*-adisk, ved jeg faktisk ikke.

9. Mere om tekst

9.13.3 Indsættelse af den formaterede terminologiliste

Med indekser in mente, indsættes den formaterede liste med kommandoen

Eksempel
9.77

```
\printglossary
```

Vi nævnte i anbefaling 9.36 på side 259 at man bør gøre brug af programmet `makeglossaries` når man skal have formateret sin ordliste, så når man vil formatere sin ordliste anvendes

```
makeglossaries <master-navn>
```

– man bliver enten nødt til at lære hvordan man bruger kommandolinien, eller hvordan man konfigurerer sin editor til at give en knap til denne kommando.



Selve listen kan formateres på vidt forskellig vis. Sammen med `glossaries` finder man en masse eksempler på brugen af `glossaries` og formatering af listerne. Man kan f.eks. anvende formelnumre som henvisning, eller kapitelnumre. Man kan anvende tabeller eller forskellige former for lister. Se mere i Talbot (2009b) samt i `<sti>/texmf-dist/doc/latex/glossaries/samples/` (på en \TeX Live installation).

9.13.4 Eksempel

Eksempel
9.78



```
\usepackage{glossaries,amssymb}
\AtBeginDocument{\renewcommand\glossaryname{Ordliste}}
\makeglossaries
\newglossaryentry{ordliste}{name={ordliste},
description={Beskrivelse af forskellige termer og udtryk}}
\newglossaryentry{nattal}{name=\ensuremath{\mathbb{N}},
description={de naturlige tal}, sort={N}}
\newglossaryentry{http}{name={http}, text=\textsc{http},
description={Hypertext Transfer Protocol},
first = {\textsc{http} (Hypertext Transfer Protocol)}}
----- slut preamble -----
```

```
I denne \gls{ordliste} tester vi et par ting. Forkortelsen \gls{http}
og igen \gls{http}. Noget matematik:
\[\forall n \in \mathbb{N}: n > 0\]

\printglossary
```

I denne ordliste tester vi et par ting. Forkortelsen HTTP (Hypertext Transfer Protocol) og igen HTTP. Noget matematik:

$$\forall n \in \mathbb{N}: n > 0$$

Ordliste

http Hypertext Transfer Protocol. 1

N de naturlige tal. 1

ordliste Beskrivelse af forskellige termer og udtryk. 1

Tiende Kapitel

Diverse

Dette kapitel er en mindre opsamling på en del ting som ikke rigtigt passer ind andre steder. Her vil man blandt andet kunne læse om hvordan enheder skal behandles, samt meget mere.

10.1 Kode til at tjekke for L^AT_EX eller pdfL^AT_EX

Den anbefalede metode er at anvende pakken `ifpdf` med følgende syntaks:

```
\usepackage{ifpdf}

----- slut preamble -----

\ifpdf
% pdflatex
\else
% alm latex
\fi
```

Eksempel
10.1



Funktionaliteten er direkte inkluderet i *memoir*-klassen.

10.2 Formatering af tal, enheder og navngivne variable

Enheder og navngivne enheder kræver samme omhyggelighed som vi tidligere gav navngivne matematiske funktioner. Men faktisk bør man også give formatering af tal en tanke. Dansk typografi siger f.eks. at man skal anvende komma som decimalindikator, hvorimod langt de fleste andre lande anvender punktum. I dette afsnit vil vi forklare hvordan man håndterer dette nærmere.

10.2.1 Tal og enheder via *siunitx*

Sidebemærkning 10.1. I det følgende antager vi at man anvender *siunitx*, version 2. I version 2 blev det meste af konfigurationen omdøbt. Hvis man fortsat anvender version 1, burde man kunne få det meste til fortsat at virke hvis man anvender

```
\usepackage[load-configurations=version-1]{siunitx}
```

Se desuden afsnit 8 i *siunitx*-manualen (Wright, 2010).

Eksempel
10.2

Gennem flere år har der været mange forskellige pakker til at håndtere enheder på en korrekt måde. SI (det internationale enhedssystem) har faktisk en masse regler for hvordan enheder skal skrives, hvilket mellemrum der skal være mellem enheden og værdien etc.¹ Funktionaliteten af alle disse pakker er nu blevet samlet i én pakke kaldet *siunitx*. Da pakken desuden tager sig af formatering af tal, er det naturligt at

siunitx

¹ Faktisk har de også en mening om »d«-et vi anvender når vi integrerer, men det taler man ikke højt om i matematikkredse... reglen gælder vist kun for fysikere.

10. Diverse

siunitx også giver metoder som formaterer søjlerne i tabeller. Dette har vi allerede kigget på i afsnit 7.4.3 på side 179.

Til selve teksten giver siunitx os tre primære makroer vi kan bruge

```
\num[⟨options⟩]{⟨værdi⟩}
\si[⟨options⟩]{⟨enhed⟩}
\SI[⟨options⟩]{⟨værdi⟩}{⟨enhed⟩}
```

De to første kommandoer er til formatering af tal henholdsvis enheder, mens `\SI` er til at skrive værdi og enhed i ét hug (den anvender internt `\num` og `\si`). Vi skal ikke sige så meget om `⟨options⟩`-delen, men i stedet henviser til [Wright \(2010\)](#) for de nærmere detaljer.

Formatering af tal

Makroen `\num` kan mange sjove ting som er rigt illustreret i manualen, lad os komme med et par eksempler:

Eksempel 10.3



```
\usepackage{siunitx}
----- slut preamble -----
\num{1234.56}
\num[output-decimal-marker={,}]{1234.56}
$\num[separate-uncertainty]{12.34(5)}$
```

1234.56 1234,56 12.34 ± 0.05

Bemærk hvordan vi i det andet eksempel erstatter punktum med komma, som decimal tegn. Dette kan vi også gøre globalt via

Eksempel 10.4

```
\usepackage{siunitx}
\sisetup{output-decimal-marker={,}}
```

Det sidste eksempel med den korte angivelse af afvigelsen er nyttig inden for mange forskellige fag.

Man kan også angive intervaller, med

```
\numrange[⟨options⟩]{⟨fra⟩}{⟨til⟩}
```

Her kan man så specificere hvordan man skal angive at der er tale om et interval:

Eksempel 10.5



```
\usepackage{siunitx}
----- slut preamble -----
\numrange{5}{17}
\quad\numrange[range-phrase={ til }]{5}{17}
\quad\numrange[range-phrase=-]{5}{17}
```

5 to 17 5 til 17 5–17

Observer hvordan man bliver nødt til at anvende `{_til_}` for at få interval ordet skrevet til noget andet. Se også [Eksempel 10.8 på side 268](#). Det er måske ikke lige til at se hvad man skal bruge dette til, »man kan jo bare skrive det i hånden«, det interessante ved denne konstruktion er faktisk den storebror `\SIrange{⟨fra⟩}{⟨til⟩}{⟨enhed⟩}`, hvor det så begynder at blive sjovt, for nu kan siunitx selv finde ud af at gentage enheden eller sætte parenteser om værdi-delen.

Eksempel 10.6



```
\usepackage[load-configurations=abbreviations]{siunitx}
----- slut preamble -----
\SIrange[range-phrase=\dots]{15}{35}{\kg}
```

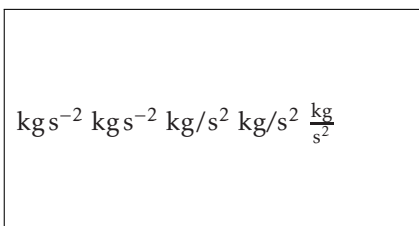
15 kg... 35 kg

Sidebemærkning 10.2. Bemærk at det anses for ukorrekt brug af SI enheder at skrive noget i retningen af 1–5 kg, enheden bør gentages for at undgå forvirring, det samme gælder 2×2 m. Se desuden Wright (2010), afsnit 9.

Formatering af enheder alene

Ved brugen af enheder skal man huske det ikke er alle som ved hvad forkortelserne står for. Dette kan være et problem, både hos skribenten samt som læseren. Den sidste kan vi ikke gøre så meget ved,² men den første (som ofte gør sig gældende i samarbejde med forfattere fra andre felter), kan vi fikse ved brug af `siunitx`.

Med pakken kan man anvende to forskellige metoder til at skrive sine enheder. En kort syntaks hvor L^AT_EX ikke selv ved hvad der er enheder (og man derfor selv skal markere når man går til en ny enhed) og en mere omstændig (engl: *verbose*) hvor navnet på hver enhed skrives helt ud, og hvor prefixer, eksponenter og lignende skrives med ord. Den lange notation gør det meget nemmere for medforfattere at forstå hvad der stå, således der ikke kommer tvivlsspørgsmål.³ Lad os illustrere med et eksempel

<pre style="background-color: #ffffcc; margin: 0; padding: 5px;">\usepackage{siunitx} ----- slut preamble ----- \si{kg.s^{-2}} \si{\kilo\gram\per\second\squared} \si{kg/s^{2}} \si[per-mode=symbol]{\kilo\gram\per\second\squared} \si[per-mode=fraction]{\kilo\gram\per\second\squared}</pre>	
---	---

Eksempel
10.7



Begge indtastninger giver det samme, bemærk hvordan man bliver nødt til at indsætte et punktum for at adskille de to enheder i den første konstruktion. Den lange type er meget mere læsbar.

I de to foregående eksempler kan vi også se den generelle opbygning af en enhed: Der er et prefix (f.eks. kilo), en enhed (f.eks. gram) samt en potens.

Man kommer nok ikke til at anvende makroen `\si` direkte videre tit, men den anvendes internt i `\SI`. Dog kan `\si` være ganske anvendelig når man f.eks. i overskrifterne i en datatabel skal angive hvilken enhed der er anvendt i en søjle/række.

Her følger nu nogle tabeller med enheder, prefixer, forkortelser og nogle potenser skrevet ud i ord, se tabellerne 10.1 til 10.4 på side 267.

`Siunitx` understøtter mange flere enheder end dem vi har listet her. Nogle er direkte tilgængelige, andre kræver at man beder om dem, se Wright (2010) afsnit 5.9. Her finder man enheder til syntetisk kemi, højenergi fysik o.a. Afsnittet forklarer også hvordan man beder om at få adgang til dem.

Det er også muligt at definere sine egne enheder, enten som forkortelser som man ofte anvender, eller helt nye enheder. Dette er forklaret i Wright (2010) afsnit 5.9.

Værdier med enhed

Nu har vi allerede set hvordan tal og enheder skrives. Den mest almindelige måde at anvende syntaksen er så at anvende `\SI{<værdi>}{<enhed>}` med den åbenlyse syn-

² De fleste enhedsforkortelser kan slås op i Wikipedia.

³ En anden fordel med den lange syntaks er for blinde! Der laves pt. forsøg med at udvide den traditionelle 6-punkt blindeskrift til 8-punkt (se <http://www.sensus.dk/latex1>), således at man får plads til at beskrive matematik og enheder via L^AT_EX-kode.

10. Diverse

Tabel 10.1: Nogle enheder

Enhed/makro	Symbol	Enhed/makro	Symbol	Enhed/makro	Symbol
<i>De syv SI grundenheder</i>					
<code>\kilogram</code>	kg	<code>\metre^a</code>	m	<code>\second</code>	s
<code>\mole</code>	mol	<code>\kelvin</code>	K	<code>\ampere</code>	A
<code>\candela</code>	cd				
<i>Afledte enheder</i>					
<code>\becquerel</code>	Bq	<code>\lux</code>	lx	<code>\celsius^b</code>	°C
<code>\newton</code>	N	<code>\ohm</code>	Ω	<code>\coulomb</code>	C
<code>\pascal</code>	Pa	<code>\farad</code>	F	<code>\radian</code>	rad
<code>\gray</code>	Gy	<code>\siemens</code>	S	<code>\sievert</code>	Sv
<code>\hertz</code>	Hz	<code>\steradian</code>	sr	<code>\henry</code>	H
<code>\tesla</code>	T	<code>\joule</code>	J	<code>\volt</code>	V
<code>\katal</code>	kat	<code>\watt</code>	W	<code>\lumen</code>	lm
<code>\weber</code>	Wb				
<i>Fra fysiske eksperimenter</i>					
<code>\electronvolt</code>	eV	<code>\atomicmassunit</code>	u		
<i>Accepterede enheder</i>					
<code>\bel</code>	B	<code>\day</code>	d	<code>\degree</code>	°
<code>\hour</code>	h	<code>\litre</code>	l	<code>\liter^c</code>	L
<code>\minute</code>	min	<code>\arcminute</code>	'	<code>\neper</code>	Np
<code>\percent^d</code>	%	<code>\arcsecond</code>	"	<code>\tonne</code>	t
<i>Midlertidige enheder^e</i>					
<code>\angstrom</code>	Å	<code>\decibel</code>	dB	<code>\bar</code>	bar
<code>\barn</code>	b	<code>\hectare</code>	ha	<code>\mmHg</code>	mmHg
<code>\radian</code>	rad				

NOTE 1: Bemærk at `siunitx` desuden definerer en del forkortelser, a la `\kg`, eller `\MHz`, se tabel 10.3 på næste side.

NOTE 2: Husk at enheder så som fahrenheit *ikke* er en SI-enhed og derfor ikke understøttet af `siunitx`. Man kan sikkert nemt selv definere den hvis man har brug for den i en undervisningssituation.

^a Alias: `\meter`

^b Alias: `\degreeCelsius`

^c Bemærk forskellen i forhold til `\litre`

^d Ja, % er faktisk en enhed og bør behandles som sådan.

^e Betegnelsen *midlertidig* går vist på at disse ikke helt er ratificeret af SI, se Wright (2010) afsnit 20.

Værdier med enhed

Tabel 10.2: Enheds prefixer

Makro	Værdi	Symbol	Makro	Værdi	Symbol
<code>\yocto</code>	10^{-24}	y	<code>\deca</code>	10^1	da
<code>\zepto</code>	10^{-21}	z	<code>\hecto</code>	10^2	h
<code>\atto</code>	10^{-18}	a	<code>\kilo</code>	10^3	k
<code>\femto</code>	10^{-15}	f	<code>\mega</code>	10^6	M
<code>\pico</code>	10^{-12}	p	<code>\giga</code>	10^9	G
<code>\nano</code>	10^{-9}	n	<code>\tera</code>	10^{12}	T
<code>\micro^a</code>	10^{-6}	μ	<code>\peta</code>	10^{15}	P
<code>\milli</code>	10^{-3}	m	<code>\exa</code>	10^{18}	E
<code>\centi</code>	10^{-2}	c	<code>\zetta</code>	10^{21}	Z
<code>\deci</code>	10^{-1}	d	<code>\yotta</code>	10^{24}	Y

NOTE: Makronavnet angiver også navnet på prefixet.

^a Bemærk hvordan denne automatisk giver os et opretstående μ som er det korrekte tegn for mikro. Udseendet af denne kommer an på fonten som man anvender. Se [Wright \(2010\)](#).

Tabel 10.3: Nogle sammensatte og forkortede enheder
(kræver `load-configurations=abbreviations`)

Lang	Kort	Symbol	Lang	Kort	Symbol
<code>\kilogram</code>	<code>\kg</code>	kg	<code>\milligram</code>	<code>\mg</code>	mg
<code>\nanometre</code>	<code>\nm</code>	nm	<code>\micrometre</code>	<code>\um</code>	μm
<code>\millimetre</code>	<code>\mm</code>	mm	<code>\centimetre</code>	<code>\cm</code>	cm
<code>\kilometre</code>	<code>\km</code>	km	<code>\second</code>	<code>\s</code>	s
<code>\microsecond</code>	<code>\us</code>	μs	<code>\millisecond</code>	<code>\ms</code>	ms
<code>\milliampere</code>	<code>\mA</code>	mA	<code>\kiloampere</code>	<code>\kA</code>	kA
<code>\millilitre</code>	<code>\ml</code>	ml			
<code>\hertz</code>	<code>\Hz</code>	Hz	<code>\millihertz</code>	<code>\mHz</code>	mHz
<code>\kilohertz</code>	<code>\kHz</code>	kHz	<code>\megahertz</code>	<code>\MHz</code>	MHz
<code>\gigahertz</code>	<code>\GHz</code>	GHz	<code>\millivolt</code>	<code>\mV</code>	mV
<code>\kilovolt</code>	<code>\kV</code>	kV	<code>\kilojoule</code>	<code>\kJ</code>	kJ
<code>\millielectronvolt</code>	<code>\meV</code>	meV	<code>\kiloelectronvolt</code>	<code>\keV</code>	keV
<code>\megaelectronvolt</code>	<code>\MeV</code>	MeV			

NOTE: Flere forkortelser kan ses i [Wright \(2010\)](#) tabel 20.

Tabel 10.4: Predefinerede potenser

Makro	Eksempel	Resultat	Før/efter
<code>\squared</code>	<code>\si{\metre\squared}</code>	m^2	efterstillet
<code>\square</code>	<code>\si{\square\metre}</code>	m^2	foranstillet
<code>\cubic</code>	<code>\si{\cubic\metre}</code>	m^3	efterstillet
<code>\cubed</code>	<code>\si{\metre\cubed}</code>	m^3	foranstillet

NOTE: I ([Wright, 2010](#), afsnit 7.3) kan man finde makroer til enten generelt at definere flere potenser, eller makroer til hurtigt at skrive en ny potens inline.

10. Diverse

taks. Man kan se `\SI{<værdi>}{<enhed>}` som værende nogenlunde det samme som `\num{<værdi>}\si{<enhed>}` (hvilket dog ikke er helt rigtigt).

Eksempel 10.8



```
\usepackage{siunitx}
\sisetup{per=slash}
```

— slut preamble —

```
\SI{35}{\mole\per\litre}
```

35 mol/l

10.3 Komma som separator i decimaltal

Egentlig siger dansk retskrivning at på dansk skal decimaltegnet i decimaltal være *komma* og ikke punktum som anvendes på engelsk.⁴ Det går jo fint så længe at man skriver decimaltallet i almindelig tekst »23,95«. Gør man det derimod i matematik så er kommaet jo et 'tegnsætningstegn' som automatisk indsætter noget mellemrum efter kommaet: »23,95«.

Den bedste løsning på dette er at anvende `\num` fra `siunitx` (beskrevet på side 264), man omkranser simpelthen hver af disse talværdier med makroen, og kan så styre formateringen af disse globalt i dokumentets preamble.

Eller man kan vælge helt at gå over til at anvende punktum som decimaltegn, dette kan man så skrive i sin opgave (f.eks. med begrundelsen om at dette er hvad mange af ens programmer gør brug af).

10.4 Navngivne variable og enheder

I et tidligere kapitel nævnte vi hvordan navngivne funktioner skulle skrives med en ikke-kursiv opretstående font. Der gælder tilsvarende anbefalinger vedr. variable indiceret via et ord, samt mere vigtigt ved fysiske enheder. Dette vil vi kigge lidt nærmere på i dette afsnit.

Navngivne variable

Det vi her henviser til er hvordan man f.eks. navngiver den maksimale radius R_{\max} på den mest fornuftige måde.

Vi har to regler som skal overholdes: På den ene side skal vi have adgang til alle tegn, også æøå, og teksten skal skrives ikke-kursivt. Resultatet man så kommer frem til er »`\textup`«. ⁵ Vi ender altså op med `\textup` som slet og ret betyder »ikke-kursiv«.

Det man skriver er bliver noget lignende $X_{\textup{\max}}$. Med mindre man har en god editor (som giver nemt adgang til `\textup`, så bliver man hurtigt træt af altid at skulle skrive `_{\textup{\max}}`). Via Enrico Gregorio, Donald Arseneau og sluttelig Morten Høgholm, er vi derfor kommet frem til følgende tip:

⁴ Der er dog mange som i dag konsekvent anvender punktum, da det så er nemmere at anvende samme data mellem flere sprog.

⁵ Nogle ville måske have anvendt `\text`, `\mathrm` eller `\textrm`, men de har hver især fejl og mangler som gør dem irrelevante. `\text` bliver kursiv hvis den omkringliggende tekst er kursiv, `\mathrm` understøtter ikke æøå, mens `\textrm` skifter til »romansk/antikvask« tekst, hvilket kan give problemer hvis nuværende tekst er sans-serif.

Versionskontrol

```
\makeatletter
\begingroup
\catcode`\_=\active
\protected\gdef_{\@ifnextchar|\subtextup\sb}
\endgroup
\def\subtextup|#1|{\sb{\textup{#1}}}
\AtBeginDocument{\catcode`\_ =12 \mathcode`\_ =32768 }
\makeatother
```

Kodetip

10.1



Dette udvider funktionaliteten af subscript i matematik-mode således at X_{\max} giver det almindelige (og forkerte) X_{max} , mens hvis vi erstatter $\{$ 'erne med et par af $|$ 'er så få man subscript med \textup , vi forudsætter naturligvis at man anvender `amsmath`. Se [Eksempel 10.9](#).

```
\[ X_{\max} \quad X_{|max|} \]
```

X_{max}	X_{\max}
-----------	------------

Eksempel

10.9



Man skal huske at nu kan man så ikke længere anvende $X_{|}$ her skal man i stedet skrive $X_{\{|}}$ for at få $X_{|}$.

Man skal være opmærksom på at tricket ikke kan anvendes sammen med alle pakker. Der kan sandsynligvis blive problemer med pakken `breqn`.

10.5 Versionskontrol

Som nævnt i indledningen har $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, ikke indbygget versionskontrol. Men dette betyder *ikke*, at man ikke har brug for versionskontrol til sine projekter, det er en rigtig god ide at man gør. Det mest gængse system til versionskontrol pt. er Subversion (et endnu mere moderne alternativ er Git). Fordelen ved versionskontrol er at man kan følge udviklingen af projektet, kan ses hvad der er ændret mellem hver revision og hvorfor (hvis man sørger for at skrive sine kommentarer ordentligt). En anden fordel (i hvertfald hvis man har adgang til f.eks. et subversion depot/repository) er at man nemt kan være flere om det samme projekt, så en gruppe kan arbejde på samme dokument samtidigt og så synkronisere sine ændringer med depotet. Mange institutter/universiteter tilbyder deres studerende adgang til versions kontrol.

De fleste versionskontrollsystemer kan konfigureres til at placere revisionsinformationer i de filer som ligger under versionskontrol. Der findes forskellige $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -pakker som herefter kan hente denne information ud af en given fil og stille den til rådighed i teksten. Dette kunne være: revisionsnummer, seneste revisionstidspunkt, hvad hed brugeren som lavede den sidste ændring.

The $\text{PracT}_{\text{E}}\text{X}$ Journal, 2007-3 indeholder en masse artikler om brugen af Subversion i forbindelse med $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ projekter, udgaven kan ses via <http://www.tug.org/pracjourn/2007-3/>. Især artiklen, *Version Control of LaTeX Documents with svn-multi* af Martin Scharrer ([Scharrer, 2007](#)) giver en god indføring i brugen af Subversion for både Linux og Windows brugere, samt forklarer hvordan man kan anvende pakken `svn-multi` til at få fat på revisionsnumre etc.

Sidebemærkning 10.3. Når man arbejder med L^AT_EX er der jo en masse autogeneratede filer, .aux, .log etc. Disse behøver ikke at komme under versionskontrol (de bliver jo automatisk regenereret hvis de mangler). De filer som man bør lægge under versionskontrol er de filer som ikke automatisk genereres ud fra andre filer i projektet. Dvs. dette er .tex-filer, egne pakker, konfigurationsfiler, datafiler og programkode (hvis man har brug for dette), figurer.

Har man derimod f.eks. figurer lavet i METAPOST (og man har kildekoden til figurerne), så er der jo ikke nogen grund til at lægge selve METAPOST-tegningerne under versionskontrol, gem i stedet METAPOST-kildekoden, så kan figurerne regenereres derfra.

Dette er hvad der er gjort ved denne bog. Her har vi et subversion depot med .tex-filer, litteraturdata (.bib), konfigurationsfiler til diverse programmer, navnene på nogle biblioteker som skal eksistere (men L^AT_EX fylder dem selv med indhold), METAPOST-filer for alle tegninger i bogen, specielle egne pakker samt nogle scripts som anvendes til at bygge bogen. Resten ignoreres da de kan genereres ud fra de andre filer. I mit depot til bogen er der pt. 93 filer, hvilket vokser til mellem 2200 og 2500 filer når alt er oversat.

Tip 10.4. »*Introduktion til L^AT_EX*« ligger naturligvis i Subversion. Sidder man med et stort projekt som dette, hvor man stiller beta versioner til rådighed for folk, så kan det være en ide at sørge for at læseren kan se hvornår man sidst har ændret noget i de enkelte kapitler.

Dette er ret nemt med subversion og svn-multi. Først sørger man for at slå *subversion keywords* til som beskrevet i [Scharrer \(2007\)](#). Herefter sætter man følgende i toppen af hver side (man kan bare slette det som der står mellem »:« og det sidste »\$«).

**Eksempel
10.10**



```
\svnidlong
{$HeadURL: file:///home/daleif/svn-repos/ltxb/trun/diverse.tex $}
{$LastChangedDate: 2011-09-13 19:03:09 +0200 (Tue, 13 Sep 2011) $}
{$LastChangedRevision: 891 $}
{$LastChangedBy: daleif $}
```

Når man nu lave en *subversion update* af sit projekt, så vil Subversion automatisk opdatere/indsætte relevant information. Via interfacet fra svn-multi kan vi nu få fat på dette og vi kan bruge det i dokumentet, f.eks. ved at pille ved pagestylen (i et *memoir* dokument):

**Eksempel
10.11**



```
\usepackage[filehooks]{svn-multi}
\usepackage{xcolor}
\newcommand\addRevisionData{%
  \begin{picture}(0,0)%
    \put(0,-10){%
      \tiny%
      \expandafter\@ifmtarg\expandafter{\svnfiledate}{}%
      \textcolor{darkgray}{Kapitlet er sidst opdateret
      \svnfileyear/\svnfilemonth/\svnfileday
      \enspace \svnfilehour:\svnfileminute\ (revision \svnfilerev)}
    }%
  }%
\end{picture}%
}
\makeoddfoot{plain}{\addRevisionData}{\thepage}{%}
\makeevenfoot{plain}{\addRevisionData}{\thepage}{%}
```


Normalsider og anslag

Der findes en del alternativer til svn-multi-pakken, f.eks. svninfo, som i modsætning til den nuværende version af svn-multi ikke gør brug af en ekstra write (skrivning til ekstern fil, L^AT_EX har en begrænset antal af disse).

10.6 Normalsider og anslag

Når man arbejder med L^AT_EX, især med matematik, tabeller og figurer, så er det min påstand at normalsider samt antal anslag ikke uden videre giver mening. Hvordan skal matematikken eller tabellerne tælles med? Skal der tælles i outputtet (DVI, PostScript eller PDF) eller skal der tælles i (dokumentdelen af) kildekoden (hvilket i hvertfald slet ikke giver mening)?

Vedrørende normalsider så er disse ofte normalt defineret som et vist antal anslag eller ord pr. side (justeret for fontstørrelse og marginer). Men hvordan tæller matematikken med? Hvis vi nu har følgende formel:

```
\[
\sum_{\substack{i \in K \\ j > i}} U_{ij}
\]
```

$$\sum_{\substack{i \in K \\ j > i}} U_{ij}$$

Eksempel
10.12

Tæller dette så som to tegn (en sum og det summerer), ti tegn (alle tegnene i outputtet) eller skal man tælle det som ord? Umiddelbart giver det slet ikke mening at tage matematikken med i udregningen af normalsiderne, men har man masser af matematik så får man jo temmelig få normalsider. Ulrich Diez opsummerer det meget fint på CTT vedrørende automatisk ordoptælling i L^AT_EX:

I think that – beneath other aspects – automatic word-counting requires algorithms for detecting

- *which sequences of bits and bytes belong to a single writeable/drawable "expression for a concept". (Here a question might be what to consider a single writeable/drawable "expression for a concept" —e.g., how to treat hyphen-compound words and the like?)*
- *whether sequences of bits and bytes represent words at all or whether these sequences represent other kinds of writeable/drawable "expressions for a concept" [e.g., acronyms or abbreviations; chinese writing; visible algebraic signs; smileys ;-); other ways of combining visible/writeable/drawable signs for symbolizing concepts].*

Where can I learn more about such algorithms?

Citeret fra <http://groups.google.com/group/comp.text.tex/msg/38fad3b276288843>

Er man underlagt begrænsninger vedr. anslag eller normalsider, så ville jeg anbefale at man udbeder sig en specifikation af hvordan man definerer normalsider og anslag, hvad skal tælles med og hvad skal ikke tælles med. Først derefter kan man begynde at undersøge hvordan man skal tælle dette.⁶ På CTAN kan man finde programmet texcount som ser ud til at kunne lave interessante ting vedr. ordtælling i T_EX-filer.

Faktisk er der meget stor forskel på hvordan man definerer hvad en normalside egentlig er. En artikel i [Universitetsavisen for Aarhus Universitet \(2006\)](#), nummer 10,

⁶ Man kunne f.eks. lave dokumentet om til en PDF-fil og så konvertere denne til ren tekst (det er der programmer som kan gøre), og så man kan tælle på resultatet.

10. Diverse

viser hvor stor forvirringen er, god kommentar fra studievejleder for naturvidenskab, Sven Skyum:

–Normalsider!? Det går vi ikke meget op i her hos os. Vi bedømmer på kvaliteten og går ikke sådan og tæller tegn. <http://www.au.dk/univers/arkiv/2006/arkiv06/#10>

Normalt er normalsider og anslagsangivelser ikke noget man skal være bange for. De er oftest lavet for at folk enten (a) ikke skal skrive en hel bog⁷, eller (b) ikke skal snyde ved at lave stor skriftstørrelse, eller stor linie afstand for nemt at kunne opnå et bestemt antal sider. Der er dog visse steder hvor anslag og normalsider kan være meget vigtigere, f.eks. i humaniora (lignende fag) eller journalistik, hvor det er vigtigt at man kan formulere sig kort og præcist.

10.7 Elektroniske slides

beamer Hvis man skal holde en præsentation og man ønsker at anvende L^AT_EX til at lave slides til PDF, så anbefales dokumentklassen *beamer* (alternativt kan man anvende *powerdot*⁸).

Anbefaling. Klassen *beamer* er, som klassen *memoir*, meget veldokumenteret, og som bruger af *beamer* anbefales man naturligvis at gøre sig bekendt med manualen.

Beamer er et meget stort system, så vi har ikke plads til at gå så meget ned i detaljer, så ud over de basale ting må man altså konsultere manualen. Iøvrigt kan man finde et par af *beamer*-forfatterens egne foredrag (inklusive kildekode) via CTAN: [/macros/latex/contrib/beamer/examples/](http://www.ctan.org/ctan/latex/contrib/beamer/examples/)

Det kan anbefales at man kigger indholdsfortegnelsen for Tantau (2005), igennem, så får man et indtryk af hvad der er forklaret rundt omkring i manualen. Det er en god ide at læse afsnit 3: *Euclid's Presentation*, det er kun otte sider, men man får et indblik i tankegangen bag opbygningen af en præsentation med *beamer*.

En anden god ting ved manualen er at forfatteren også tager sig tid til at skrive en vejledning til hvordan man skriver gode præsentationer. En af de mest almindelige fejl folk gør når de laver præsentationer, er at inkludere alt for meget tekst. De forskellige sider i præsentationen er med til at understrege det man som foredragsholder står og siger, det er ikke meningen at folk skal sidde og læse en masse tekst, for så hører de jo ikke efter hvad det er man har at sige.

Når man arbejder med elektroniske slides er der to koncepter man lige skal kende betydningen af: *frame* og *overlay* (vi vil anvende de engelske betegnelser). En *frame* svarer til en side eller skærmbillede i præsentationen, mens en *overlay* svarer til at man lægger et gennemsigtigt billede ovenpå siden, dette kan så tilføje ekstra information til denne frame. Nogle kalder overlays for layers, altså lag.

Sidebemærkning 10.5. Beamer betegner layers som *slides*, men det er ret forvirrende på dansk så det vil vi ikke gøre.

⁷ Et syndrom som undertegnede klart lider af.

⁸ Som er arvtageren til den tidligere meget anvendte klasse *prospcr*, har dog personligt ikke den større erfaring med nogen af de to klasser.

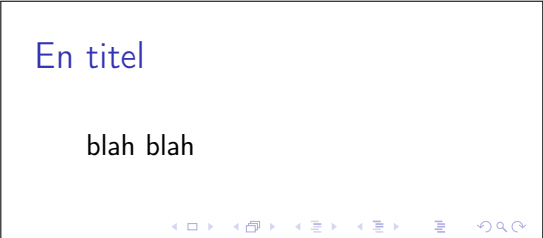

Elektroniske slides

Det klassiske eksempel er en liste, hvor selve punkterne ikke fremstår når man skifter til frame, men hvor de dukker frem et efter et. Denne konstruktion kalder vi her en *frame* med *overlays*.

I *beamer* laves en frame naturligt nok med environmentet *frame* som har følgende syntaks:

```
\begin{frame}[\langle options \rangle]
\frametitle{\langle Titel \rangle}
\langle indholdet af denne frame \rangle
\end{frame}
```

Titel-delen kan udelades. I standard konfigurationen kommer det til at se sådan ud (størrelsesforholdet snyder, vi har pillet lidt for at spare plads):

<pre>\documentclass{beamer} ----- slut preamble ----- \begin{frame} \frametitle{En titel} blah blah \end{frame}</pre>		Eksempel 10.13 
---	---	--

Linien man ser nederst er en navigationslinie man kan anvende i f.eks. Adobe Reader. En anden ting man lægger mærke til at der automatisk anvendes sans serif fonte, det er simpelthen fordi det er det mest tydelige når man forstørre noget op på et lærred. En anden ting man skal bemærke er at de almindelige regler vedrørende god typografi ikke helt er gældende når man laver præsentationer. F.eks. er der ingen orddeling og der er afstand mellem afsnit.

Sidebemærkning 10.6. Man skal ikke blive forskrækket over omfanget af papirstørrelsen som *beamer* anvender, den er på 128 mm × 96 mm, svarende til forholdet 4 : 3 som det er de fleste projektorer arbejder med. Det er så meningen at f.eks. Adobe Reader forstørre materialet op til fuld skærmstørrelse.

Tip 10.7. Hvis man har brug for at fremvise kildekode (f.eks. SAS-kode) i en frame, så skal man lige huske at tilføje optionen »fragile« til *frame*-environmentet, altså

```
\begin{frame}[fragile]
```

så kan man anvende f.eks. `listings` indeni en frame. Man skal også huske »fragile« hvis man anvender `\verb` eller *verbatim*-environmentet, eller nogle konstruktioner fra SASdisplay-pakken.

Layoutet er en smule kedeligt, men kan piftes gevaldigt op med *themes* (temaer):

10. Diverse

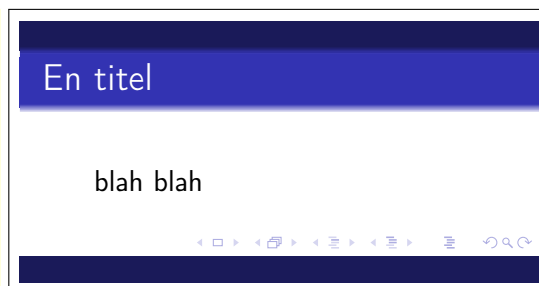
Eksempel 10.15



```
\documentclass{beamer}
\usetheme{Frankfurt}
\useoutertheme[subsection=false,
footline=authortitle]{miniframes}

----- slut preamble -----

\begin{frame}
\frametitle{En titel}
blah blah
\end{frame}
```



De forskellige temaer kan man læse mere om og se eksempler på i [Tantau \(2005\)](#).

På forsiden af præsentationen ønsker man ofte at have oplysninger om hvem man er og hvor man kommer fra. Så i *beamer* skriver man sine oplysninger i preamblen via følgende makroer (se også afsnit 9.1 i [Tantau \(2005\)](#)):

```
\title      \title[⟨kort titel⟩]{⟨lang titel⟩}
\subtitle   \subtitle{⟨undertitel⟩}
\author     \author[⟨kort navn⟩]{⟨langt navn⟩}
\institute  \institute[⟨kort tekst⟩]{⟨lang tekst⟩}
\date      \date[⟨kort dato⟩]{⟨lang dato⟩}
```

De korte titler kan anvendes i automatisk generet sidehoved/fod, dette er noget som sættes op via de forskellige temaer. I de lange titler som man anvender `\` til at skifte linien. Man laver så selve forsiden via kommandoen `\titlepage` som man smider i en *frame*:

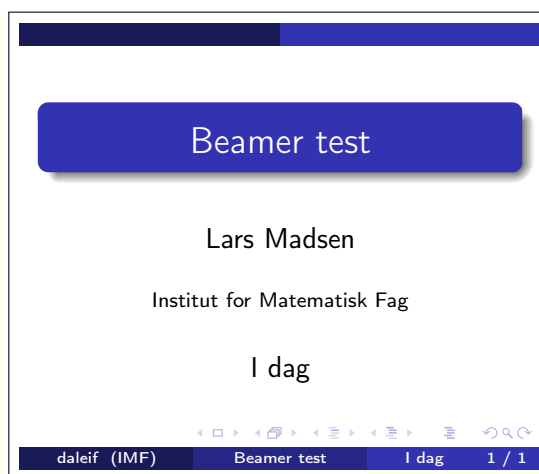
Eksempel 10.16



```
\documentclass{beamer}
\usetheme{Frankfurt}
\useoutertheme{infolines}
\title{Beamer test}
\author[daleif]{Lars Madsen}
\institute[IMF]{Institut for
Matematisk Fag}
\date{I dag}

----- slut preamble -----

\begin{frame}
\titlepage
\end{frame}
```



Bemærk effekten af det anderledes ydre tema, se [Tantau \(2005\)](#), afsnit 15.2.

Layers

9

En trinvis punktopstilling (som er noget af det mest anvendte i en præsentation) kan laves på flere måder. Man kan selv skrive hvilket lag et punkt skal være på:

⁹ FixMe Note: det kan være en ide at nævne noget vedr. effekter med layers, fik en mail om det for lang tid siden

Sætninger

```
\begin{itemize}
\item<1-> Punkt 1
\item<2-> Punkt 2
\item<3-> Punkt 3
\end{itemize}
```

**Eksempel
10.17**

Her betyder f.eks. »<1->« ar dette punkt skal fremvises fra lag 1 og frem (for netop denne frame). Havde man skrevet »<1-2>« så ville punktet kun blive fremvist på lag 1 og 2, men ikke fra lag 3 og frem. Tilsvarende ville hhv. »<-3>« og »<2>« vise punktet frem til og med lag 3 hhv. kun blive fremvist på lag 2. Dette kan dog blive træls i længden så man kan også gøre det automatisk, se [Eksempel 10.18](#).

```
\begin{itemize}[<-+>] % automatisk trinvis fremvisning
\item Punkt 1
\item Punkt 2
\item Punkt 3
\end{itemize}
```

**Eksempel
10.18**

Har man en mere almindelig tekst så kan man anvende `\pause`, alt efter `\pause` kommer først med på det næste layer (makroen er dog ikke så fleksibel som mange af de andre layer makroer).

`\pause`

```
blah blah \pause nisse nisse
```

**Eksempel
10.19**

På første layer står der »blah blah«, på andet layer (aktiveret med `\pause`) står der så »blah blah nisse nisse«. Med makroen

```
\uncover<<layer info>>{\langle tekst \rangle}
```

`\uncover`

kan man indsætte noget `\langle tekst \rangle` som først bliver vist frem på et bestemt layer. Se desuden [Tantau \(2005\)](#), afsnit 8, her kan man også lære med om `\langle layer info \rangle`-syntaksen.

Ønsker man at lave en trinvis fremvisning af en lang udregning i en *align*, så skal man passe lidt på. For det første bliver man nødt til at omkranse hver række i udregningen (inklusiv `\\`) i en `\uncover<...>{\langle kode \rangle}`-konstruktion,¹⁰ desuden anbefales det at man undlader at anvende formelnumre i dette tilfælde.

```
\begin{align*}
\uncover<1->{ f(x) &= g^2(x) + ... \\}
\uncover<2->{      &= -\sqrt{-1}^n + ... \\}
\uncover<3->{      &= 0 }
\end{align*}
```

**Eksempel
10.20**

Se [Tantau \(2005\)](#), afsnit 22.4, hele afsnit 22 handler om hvordan man fremviser ting og sager trinvist. Hvis man har formelnumre med i opstillingen så sker der meget underlige ting, hvilket man selv kan teste.

Sætninger

I *beamer* er der allerede defineret en masse sætningskonstruktioner som man frit kan anvende. Ved at anvende layer instruktionerne (i.e. dette med `<...>`) så kan man nemt bestemme hvordan tingene bliver vist frem. I [Eksempel 10.21](#) på den følgende side skal man bemærke at temaet også påvirker hvordan sætningerne ser ud.

¹⁰ Det har noget med den måde *align* fungerer internt.

10. Diverse

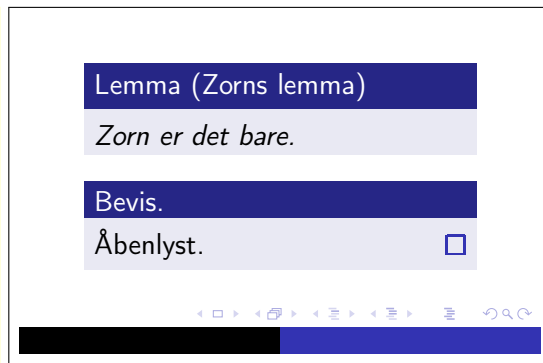
Eksempel 10.21



```
\documentclass{beamer}
\usetheme{Luebeck}

----- slut preamble -----

\begin{frame}
  \begin{lemma}<1->[Zorns lemma]
    Zorn er det bare.
  \end{lemma}
  \begin{proof}<2->
    Åbenlyst.
  \end{proof}
\end{frame}
```



Her vil lemmaet blive vist frem fra layer 1, hvorimod beviset først kommer fra layer 2, bemærk desuden at [Eksempel 10.21](#) viser layer 2!

En anden ting man skal bemærke er at de prædefinerede sætningskonstruktioner (se [Tantau \(2005\)](#), afsnit 11.4) alle står på engelsk. Ønsker man at anvende danske navne gør man som i [Eksempel 10.22](#), bemærk brugen af `notheorem` klasse-option.

`notheorem`

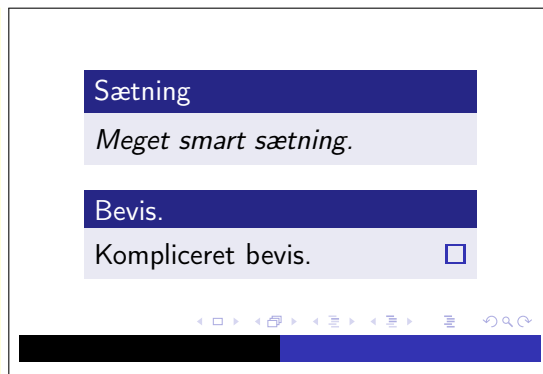
Eksempel 10.22



```
\documentclass[notheorems]{beamer}
\usetheme{Luebeck}
\newtheorem{theorem}{Sætning}

----- slut preamble -----

\begin{frame}
  \begin{theorem}<1->
    Meget smart sætning.
  \end{theorem}
  \begin{proof}<2->
    Kompliceret bevis.
  \end{proof}
\end{frame}
```



Opdeling af præsentationen

Man kan anvende de almindelige `\section` og `\subsection` til at opdele sin præsentation, de placeres *udenfor* `frame`-environmentet. Man kan så få `beamer` til at skrive en mini-indholdsfortegnelse i f.eks. toppen af hver frame, det kommer an på det tema man anvender, den kunne også nemt stå i en sidebar. Dette er også beskrevet i [Tantau \(2005\)](#), se afsnit 15.2. Eksempelvis:

Eksempel 10.23

```
\useoutertheme[subsection=false,
footline=authortitle]{miniframes}
```

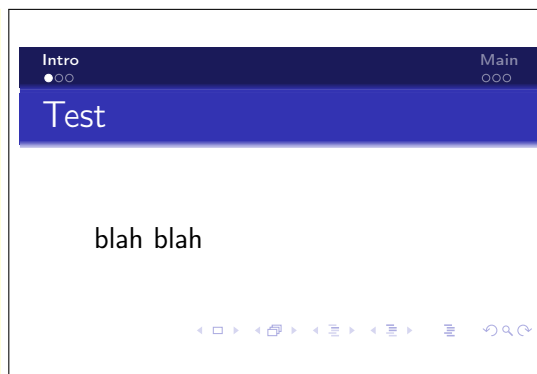
Denne konfiguration viser alle `\section`-titler (indenfor en given `\part`) i en par i toppen af de enkelte frames. Kombinerer man med brugen af `\subsection{<titel>}` så markeres hver frame med en lille cirkel, hvor den aktive er markeret med en fyldt cirkel. Grupper af disse cirkler fremhæves så alt efter hvor man har placeret sine `\subsection{<>}`. Selve `\subsection`-titlen vises ikke (»`subsection=false`«). Bemærk at visse temaer har allerede `miniframes` indbygget/indlæst, dette kan man ofte se af billederne i [Tantau \(2005\)](#), afsnit 14.

Animationer

```
\documentclass{beamer}
\usetheme{Frankfurt}
\useoutertheme[subsection=false,
footline=false]{miniframes}
```

----- slut preamble -----

```
\section{Intro}
\subsection{Test}
\begin{frame}
  \frametitle{Test}
  blah blah
\end{frame}
...
```



Eksempel
10.24



Animationer

Man siger at et billede er bedre en tusind ord. I specielle tilfælde kan animationer også være meget informative. Her kan det anbefales at tage et kig på pakken `animate`, som kan hentes via CTAN, se [Grahn \(2008\)](#). En hurtig animation lavet af en serie af f.eks. METAPOST-billeder kan laves via

animate

```
\usepackage{animate}
```

----- slut preamble -----

```
\animategraphics[1]{with_handles2-}{0}{6}
```

Eksempel
10.25

hvor `with_handles2-` er et filnavnsuffix, således at animationen er lavet via filerne `with_handles2-0.pdf` til `with_handles2-6.pdf`.

Selve `beamer`-klassen har også visse metoder indbygget til at lave animationer med, se [Tantau \(2005\)](#), afsnit 14.

Handouts

Når man laver en præsentation så er det en god ide at lave nogle trykte eksemplarer til uddeling blandt tilhørerne. Det kan endda være smart at efterlade plads på disse print til tilhørens egne noter.

Da de fleste præsentationer anvender layers i en eller anden form, giver det problemer hvis man printer et sådant dokument, hvert layer kommer ud som en side for sig selv. Dette er både dårligt for miljøet og irriterende for læseren. Klassen `beamer` løser dette ved at tilbyde en klasse-option kaldet `handout`. Når denne er aktiveret vil layers helt forsvinde og man vil i stedet se det fulde indhold af hver frame (svarende til det allersidste layer i hver frame). Man skriver altså bare

handout

```
\documentclass[handout]{beamer}
```

og så oversætter man sit dokument igen. Handouts er nærmere beskrevet i [Tantau \(2005\)](#), afsnit 20.

Eksempel
10.26



Næste problem bliver jo så at hver frame har en ret lille størrelse, se bemærkning 10.6 på side 273. Men på den anden side så lægger dette jo op til at man sparer noget papir ved at placere flere slides på en side. Indbygget i `beamer` har man konfigurationsmuligheder således at vi kan lave specielle konfigurationer til når man har aktiveret `handout` og andre konfigurationer som gælder dokumentet i alle `modes`. Rasmus Villemoes har indsendt følgende trick som skal placeres i preamble, den smider 8 frames på en side og sætter et sidetal nederst.

Eksempel 10.27

```

\mode
<handout>
\usepackage{pgfpages}
\pgfpagesuselayout{8 on 1}[a4paper,border shrink=4mm]
\setbeamertemplate{footline}[page number]

\mode
<all>

```

pgfpages Pakken *pgfpages* er en del af PGF som man allerede har, da den er påkrævet af *beamer*. Den er nærmere beskrevet i [Tantau \(2007\)](#), afsnit 36. Ved at kigge lidt på koden til »8 on 1« er vi kommet frem til [Kodetip 10.2 på modstående side](#) som kan sætte fire frames op i en 4 × 2 matrix, hvor højre søjle er tom (skal anvendes til noter). Så skal man tage [Eksempel 10.27](#), erstatte »8_on_1« med »4_on_1_with_space_for_notes« samt placere koden fra [Kodetip 10.2 på modstående side](#) efter `\usepackage{pgfpages}` og før den anvendes i `\pgfpagesuselayout`.

Sidebemærkning 10.8. Konstruktionen med *pgfpages* kan også anvendes sammen med almindelige dokumenter, det behøver ikke være *beamer*-slides.

Omvej

Det kræver stor erfaring at vide præcist hvad man kan nå ved en præsentation eller en forelæsning. I AUs universitetsavis Campus 19/2007 ([Universitetsavisen for Aarhus Universitet, 2007](#)) var der en diskussion for og imod brugen af PowerPoint og lignende, et af problemerne som er påpeget er at ved brugen af slides er det svært at lave omveje i sin undervisning. Men dette kan man jo planlægge sig ud af.

Beamer-klassen stiller forskellige redskaber til rådighed således at man kan lave knapper der sender en på kryds og tværs i præsentationen, dette kan f.eks. være en knap som siger »*Skip proof*« for at hoppe over beviset for en sætning. Det kan være en udflugt ned i et appendix, hvori man så sørger for at lave sig en knap som pege tilbage igen. Dette er nærmere forklaret i [Tantau \(2005\)](#), afsnit 10.

Transparenter

trans Man kan godt lave transparenter med *beamer*, man skal bare tilføje klasse option *trans*, se [Tantau \(2005\)](#), afsnit 19.

Man kan også lave transparenter på mange andre måder. DK-TUG-foredraget om *memoir* ([Madsen, 2003](#)) har slides lavet direkte i *memoir* og efterbehandlet med programmet *ppower4*. Disse slides kunne også nemt printes på transparenter.

10.8 Statistik

11

Der findes forskellige småpakker rundt omkring som samler lidt statistisk notation sammen. Jeg vil dog personligt vurdere at de ofte ikke er besværet værd, de makroer man skal bruge kan man lige så godt lave selv. En ting man skal huske indenfor statistik er at sørge for at få lavet alle funktionsnavne lavet om til operatorer, dette

¹¹ FixMe Note: dette skal gennemgås og drøftes med granfeldt og blæsild



```

\pgfpagesdeclarelayout{4 on 1 with space for notes}
{ \edef\pgfpageoptionheight{\the\paperwidth} % landscaped by default
  \edef\pgfpageoptionwidth{\the\paperheight}
  \def\pgfpageoptionborder{0pt}}
{ \pgfpagesphysicalpageoptions{logical pages=4,%
  physical height=\pgfpageoptionheight,%
  physical width=\pgfpageoptionwidth}
\ifdim\paperheight>\paperwidth\relax
  % put side-by-side
  \pgfpageslogicalpageoptions{1}{%
    border shrink=\pgfpageoptionborder,%
    resized width=.25\pgfphysicalwidth,%
    resized height=.5\pgfphysicalheight,%
    center=\pgfpoint{.125\pgfphysicalwidth}{.75\pgfphysicalheight}}%
  \pgfpageslogicalpageoptions{2}{%
    border shrink=\pgfpageoptionborder,%
    resized width=.25\pgfphysicalwidth,%
    resized height=.5\pgfphysicalheight,%
    center=\pgfpoint{.625\pgfphysicalwidth}{.75\pgfphysicalheight}}%
  \pgfpageslogicalpageoptions{3}{%
    border shrink=\pgfpageoptionborder,%
    resized width=.25\pgfphysicalwidth,%
    resized height=.5\pgfphysicalheight,%
    center=\pgfpoint{.125\pgfphysicalwidth}{.25\pgfphysicalheight}}%
  \pgfpageslogicalpageoptions{4}{%
    border shrink=\pgfpageoptionborder,%
    resized width=.25\pgfphysicalwidth,%
    resized height=.5\pgfphysicalheight,%
    center=\pgfpoint{.625\pgfphysicalwidth}{.25\pgfphysicalheight}}%
\else % stack on top of one another
  \pgfpageslogicalpageoptions{1}{%
    border shrink=\pgfpageoptionborder,%
    resized width=.5\pgfphysicalwidth,%
    resized height=.25\pgfphysicalheight,%
    center=\pgfpoint{.25\pgfphysicalwidth}{.875\pgfphysicalheight}}%
  \pgfpageslogicalpageoptions{2}{%
    border shrink=\pgfpageoptionborder,%
    resized width=.5\pgfphysicalwidth,%
    resized height=.25\pgfphysicalheight,%
    center=\pgfpoint{.25\pgfphysicalwidth}{.625\pgfphysicalheight}}%
  \pgfpageslogicalpageoptions{3}{%
    border shrink=\pgfpageoptionborder,%
    resized width=.5\pgfphysicalwidth,%
    resized height=.25\pgfphysicalheight,%
    center=\pgfpoint{.25\pgfphysicalwidth}{.375\pgfphysicalheight}}%
  \pgfpageslogicalpageoptions{4}{%
    border shrink=\pgfpageoptionborder,%
    resized width=.5\pgfphysicalwidth,%
    resized height=.25\pgfphysicalheight,%
    center=\pgfpoint{.25\pgfphysicalwidth}{.125\pgfphysicalheight}}%
\fi
}

```

10. Diverse

gælder f.eks. »Var«, »Cor« etc. Visse steder er der desuden tradition for at middelværdien »E« også skrives som en operator (altså ikke i kursiv). Vi minder derfor om `\DeclareMathOperator`, se afsnit 3.2.4 på side 59.

I statistik (ihvertfald på IMF ved AU) er der desuden tradition for at anvende visse specielle forkortelser f.eks. (se Blæsild og Granfeldt (2003) side 64)

$$SSD = \sum_{i=1}^n (x_i - \bar{x})^2$$

Som man kan ane ser *SDD* ikke helt optimalt ud. Det mest fornuftige ville være at det blev skrevet enten ikke-kursivt, eller i det mindste uden dette alt for store mellemrum mellem bogstaverne. Det mest fornuftige er at lave sig en ny kommando

Eksempel 10.28 `\newcommand{\SDD}{\textup{SDD}}`

Evt. kunne man have anvendt `\mathit`.

Eksempel 10.29 `\I`
`\textup{SSD} \quad \mathit{SSD}`
`\J`

SSD SSD

Jeg synes ikke at man skal skrive det som bare `SSD`, netop fordi det kan misforstås som $S \cdot S \cdot D$ og fordi teksten (alt efter hvilke bogstaver der anvendes) kommer til at se forkert ud, eksempelvis *diff* (»`\$diff`«).

Når vi nu er ved statistik, så er der jo syntaksen med »X.«, dette betyder at man har »summet« et indeks ud. Men problemet er jo at punktummet let kan drukne. Omvendt er • alt for stor. Men et fedt punktum vil sikkert kunne anvendes:

Eksempel 10.30 `\I X_{\bm{.}} \quad X_{.} \quad X_{\bullet} \quad \J`

X_{\cdot} $X_{.}$ X_{\bullet}

Lav selv genvejen `\bdot`. Om det så skal skrives som subscript eller ej, kan så diskuteres.

Statistikprogrammer

Vi har allerede kigget lidt på hvad man kan gøre med SAS (afsnit 9.10.2 på side 252).

R Et andet Statistik program som vinder større og større udbredelse er »R« (det hedder det faktisk). R er en open source implementation af S-plus (et andet statistik program), se <http://www.r-project.org>.

Sweave Som en del af en R-installation finder man Sweave som både er navnet på en R-funktionalitet samt en pakke til \LaTeX . Den generelle ide er at man kan skrive en speciel fil, som egentlig er en rapport skrevet i \LaTeX men hvor der visse steder står noget R-kode, dette køres så i gennem et program (R med Sweave), hvorefter R-koden fortolkes, f.eks. kunne den lave en simulation hvorefter programmet så indsætter en figur i stedet for R-koden. Ud i den anden ende får man få et \LaTeX dokument/fil som bare skal køres igennem \LaTeX .

For mere information om Sweave se <http://www.r-project.org> samt <http://www.ci.tuwien.ac.at/~leisch/Sweave/>.

Pakken Sweave distribueres sammen med R programmet og er derfor normalt ikke en del af ens \LaTeX -installation. Derfor kan det være nødvendigt at angive fuld sti for at hente pakken. På vores Linux system virker følgende

Eksempel 10.31 `\usepackage{/usr/lib/R/share/texmf/Sweave}`

Selve pakken giver bare nogle environments, bl.a. to verbatim environments til fremvisning af R-kode og R-output. Disse to kan sagtens omkonfigureres til f.eks. at se ud

Lidt fysik og kemi

som SAS-kodestumper kommer til at se ud med SASdisplay. Følgende kan placeres efter SASdisplay og Sweave, se kodetip 10.3.

```
% deaktiver environments fra Sweave
\let\Soutput\relax
\let\Sinput\relax
\newcounter{Routput}
\lstnewenvironment{Soutput}[1][]{%
  \refstepcounter{Routput}%
  \newcommand\mylstname{\small R-udskrift \theRoutput}%
  \lstset{style=SASdisplayBasic, columns=fixed,
    title=\mylstname, basicstyle=\SASOutputFont,
    language={}, #1}%
}{}
\newcounter{Rcode}
\lstnewenvironment{Sinput}[1][]{%
  \refstepcounter{Rcode}%
  \newcommand\mylstname{\small R-kode \theRcode}%
  \lstset{style=SASdisplayBasic, columns=flexible,
    title=\mylstname, basicstyle=\SASCodeFont,
    language=R, keywordstyle=\normalfont\bfseries\ttfamily,
    #1}%
}{}

```

Kodetip
10.3



Sidebemærkning 10.9. Der findes noget ligende for SAS, se <http://www.cs.uiowa.edu/~rlenth/SASweave/>.¹²

10.9 Lidt fysik og kemi

Andre kan sikkert også have glæde af visse dele.

Kemiske forbindelser

Grundstofnavne bør som hovedregel også altid skrives med upright font.¹³ Dette bliver hurtigt bøvlet hvis man skal gøre det pr. håndkraft (eks. $\text{\textup{H}}_2\text{\textup{O}}$). Hvilket der selvfølgelig også er andre som synes. Man kan i stedet anvende pakken mhchem. Vi vil behandle version 3, som aktiveres via

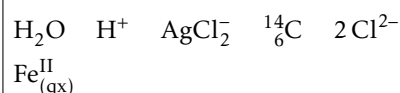
mhchem

```
\usepackage[version=3]{mhchem}
```

Brugen af pakken består af makroen $\text{\ce{\langle tekst \rangle}}$ hvor indholdet af $\langle tekst \rangle$ så fortolkes på forskellig vis. Først et simpelt eksempel.

\ce

```
\usepackage[version=3]{mhchem}
----- slut preamble -----
\ce{H2O} \quad \ce{H+} \quad \ce{AgCl2-} \quad
\ce{^{14}_6C} \quad \ce{2Cl^2-} \quad \ce{Fe^{II}_{(qx)}}
```



Eksempel
10.32



\ce vil anvende upright tekstfont når \ce anvendes i tekstsammenhæng og upright matematikfont når \ce anvendes i matematik. Pakkens features tæller altså (jf. [Eksempel 10.32](#)):

¹² Tak til Jørgen Granfeldt for dette link.

¹³ Her gør det ikke så meget om det er tekst- eller matematikfont, da der kun er tale om bogstaverne a til z.

10. Diverse

- (i) Tal som *ikke* står *først* i argumentet til `\ce` laves automatisk til subscript. Hvis tallet står først i strengen skrives den bare som normalt.
- (ii) I slutningen af argumentet til `\ce` vil tegnene »+« og »-« automatisk skrives som superscript.
- (iii) `\ce` kan sagtens skrive isotoper rigtigt, sammenlign følgende (atomnummeret)

Eksempel 10.33



```
\usepackage[version=3]{mhchem}
----- slut preamble -----
 $\textsuperscript{227}_{90}\text{Th}$  versus
 $\text{}^{\text{227}}_{\text{90}}\text{Th}$ 
```



– hvor vi for sammenligningens skyld har skrevet begge dele i matematikmode.

Man kan dog lave mange flere ting. Her er nogle ekstra ting man bør vide

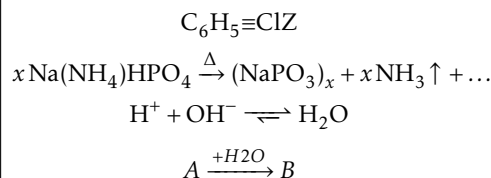
- (a) Tal kan tvinges i superscript og »+«, »-« kan tvinges ned i subscript via den normale »^« hhv. »_« syntaks.
- (b) Punktum (».«) eller stjerne (»*«) laves automatisk om til gangetegn (»·«).
- (c) Skrives man »_+« eller »_-« med mellemrum før og efter så får man et almindeligt +/-.
- (d) Bindinger kan angives via »-«, »=« eller »#« som giver hhv. -, =, eller ≡. Dog giver minus i slutningen af et udtryk stadigvæk minus i superscript. Se desuden [Hensel \(2005\)](#), for mere om bindinger.
- (e) Skrives man »_↑« med mellemrum før og efter får man ↑ tilsvarende giver »_↓« med mellemrum før og efter ↓.
- (f) Man kan lave reaktionspile via forskellige »_>_«-typer, f.eks. »_>_«, »_<->_«, »_<=>_«, »_<=>>_«. Man kan endda skrive over og under via »_>[over][under]_«. Mellemrummene er *vigtige*!
- (g) Da alle bogstaver skrives i upright, kan man anvende `\cmath` til at skrive *almindeligt* matematik.

Igen henviser vi til [Hensel \(2005\)](#) for flere detaljer. Lad os slutte af med et eksempel.

Eksempel 10.34



```
\usepackage[version=3]{mhchem}
\usepackage{amsmath}
----- slut preamble -----
\begin{gather*}
\ce{C6H5#ClZ} \\
\ce{\cmath{x},_Na(NH4)HP04_->[\Delta]
(NaP03)_\cmath{x},_+NH3_
+_\cmath{\dots}} \\
\ce{H+_OH_-<=>_H2O} \\
\ce{\cmath{A}_->[\cmath{+H2O}]_\cmath{B}}
\end{gather*}
```



Der er flere eksempler i manualen. Forfatteren bag `mhchem` har lovet at ville kigge lidt nærmere på muligheden for at man kan lave alignment med labels, hvilket den nuværende version ikke kan, den kan kun uden formelnumre. Bemærk brugen af `\cee` i stedet for den normale `\ce`.

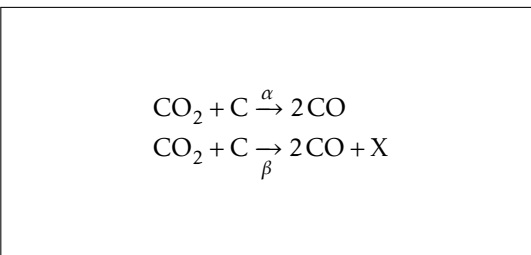
```

\usepackage[version=3]{mhchem}
\usepackage{amsmath}

----- slut preamble -----

\begin{align*}
\cee{
  CO2 + C & \rightarrow[\alpha] 2CO \ \backslash
  CO2 + C & \rightarrow[\beta] 2CO + X }
\end{align*}

```



Eksempel
10.35


Grafik til fysik, kemi og elektronik

Det ligger uden for rammerne af denne bog at gå for meget i detaljer med de L^AT_EX-konstruktioner som kan hjælpe med forskellige typer af tegninger til fysik, kemi og elektronik. Har man adgang til Goosens et al. (1997) så er der en del eksempler i den. Vi vil bare liste nogle ting man kan tage et nærmere kig på.

XYM_TE_X-systemet

Dette er et system af pakker som anvendes til at tegne en masse kemiske forbindelser. Desværre synes forfatteren bag XYM_TE_X ikke at pakken skal være på CTAN. Desuden er pakken ret svært at installere. Dette er ret kedeligt, da pakken faktisk kan lave en masse spændende ting. En ældre version af pakkerne kan findes via CTAN: [/macros/latex/contrib/xymtex/](http://ctan.org/ctan/packages/macros/latex/contrib/xymtex/), man kan også finde lidt information via <http://en.wikipedia.org/wiki/XyMTeX>.

Vi havde i en tidligere version af bogen et eksempel på disse kemiske tegninger, men det er blevet fjernet da pakkerne ikke længere distribueres sammen med standard L^AT_EX-distributioner.

Andre

på <http://texcatalogue.sarovar.org/bytopic.html#charts> kan man se en oversigt over forskellige systemer til f.eks. at tegne Feynman diagrammer eller til at tegne elektronik diagrammer.

10.10 Beskrivelse af algoritmer

Mange får sikkert desuden brug for at skulle beskrive diverse algoritmer, ofte kaldet *pseudokode*. Vi vil beskrive pakken `algorithmicx`.

pseudokode
algorithmicx

10.10.1 Brug af *algpseudocode*-pakken

Sjovt nok anvender man *ikke* pakken `algorithmicx` direkte, man anvender i stedet pakken `algpseudocode` i stedet, den tjener bare som en frontend for `algorithmicx`. Syntaksen som anvendes er meget simpel

algorithmicx
algpseudocode

```

\begin{algorithmic}[\langle num \rangle]
\langle tekst \rangle
\end{algorithmic}

```

algorithmic

– hvor *⟨tekst⟩* så skal skrives på en speciel måde. Argumentet *[\langle num \rangle]* anvendes til at aktivere linienumre i algoritme beskrivelsen, disse kan dog stadigvæk henvises til uden at være synlige. Lad os tage et simpelt eksempel:

Eksempel
10.36


```

\usepackage{algpseudocode,amsmath}
----- slut preamble -----
\begin{algorithmic}[3]
\Procedure{Euclid}{a,b}
\State  $r=a \bmod b$  \label{start}
\While{ $r \neq 0$ }
  \State  $a=b$ 
  \State  $b=r$ 
  \State  $r=a \bmod b$ 
\EndWhile
\State \textbf{return}  $b$ 
\EndProcedure
\end{algorithmic}
Linie \ref{start}\dots

```

```

procedure EUCLID(a,b)
   $r = a \bmod b$ 
3:  while  $r \neq 0$  do
     $a = b$ 
     $b = r$ 
6:     $r = a \bmod b$ 
  end while
  return  $b$ 
9: end procedure
Linie 2...

```

Vi observerer at

- vi har nogle operations punkter som *altid* optræder i par, f.eks. `\While` og `\End-While`,
- hver af disse par eller makroen `\State` starter en ny linie,
- linierne bliver automatisk indenteret indenfor et `\langle X \rangle \dots \End\langle X \rangle`-par.

Rent teknisk er alle disse strukturer faktisk implementeret via lister, hvert makro-par starter (og slutter) en ny liste og indholdet af denne bliver så indenteret. Indeni disse lister svarer `\State` så til `\item`. Bemærk også at kun hvert punkt i disse »lister« nummereres, det er ikke linienumre i gængs forstand.

10.10.2 Struktur konstruktioner

Bemærk at de makroer vi beskriver her passer til `algorithmicx` version 1.2, af 27. april 2005.

Først makroer som ikke har en `\End\langle X \rangle` makker.

`\State`

anvendes til at lave nye punkter i beskrivelsen, tæller desuden linienummeret en op.

`\StateX`

som `\State` men ændrer ikke ved linienummeret, faktisk kommer der slet ikke noget nummer ved dette punkt. Anvendeligt til at indsætte en blank linie.

`\Require`

anvendes foran start betingelser.

`\Ensure`

anvendes foran slutbetingelser.

`\Call{\langle navm \rangle}{\langle argument \rangle}`

anvendes til funktionskald, navnet bliver så formateret som ved `\Procedure` eller `\Function`, se nedenfor.

`\Comment{\langle tekst \rangle}`

til kommentarer. Udseendet kan ændres, se [János \(2005\)](#).

Makroerne ovenfor starter i sig selv ikke nogen ny liste. Vi mangler så de forskellige par-konstruktioner som hver især starter en underliste.

`\For{\langle tekst \rangle} \dots \EndFor`

en for-løkke.

Flydende algoritmer

`\ForAll{⟨tekst⟩}... \EndFor`
en variant.

`\While{⟨tekst⟩}... \EndWhile`
en while-løkke.

`\Repeat... \Until{⟨tekst⟩}`
selvforklarende

`\If{⟨tekst⟩}... \ElsIf{⟨tekst⟩}... \Else... \EndIf`
en if-konstruktion. Bemærk at det er `\ElsIf` og *ikke* `\ElseIf` når man skal lave et betingelsestræ.

`\Procedure{⟨navn⟩}{⟨argumenter⟩}... \EndProcedure`
til at navngive procedurer med.

`\Function{⟨navn⟩}{⟨argumenter⟩}... \EndFunction`
samme ved funktioner.

`\Loop... \EndLoop`
angivelse af en løkke.

Her er så et andet simpelt eksempel:

```
\begin{algorithmic}
\If{ $X \neq \emptyset$ }
  \ForAll{ $x \in X$ }
    \State \textbf{return}  $\tilde{x}$ 
  \EndFor
\Else\Comment{Mængden var tom}
  \State \textbf{return}  $\emptyset$ 
\EndIf
\end{algorithmic}
```

```
if  $X \neq \emptyset$  then
  for all  $x \in X$  do
    return  $\tilde{x}$ 
  end for
else
  return  $\emptyset$ 
end if
```

► Mængden var tom

Eksempel
10.37



Alle makroerne kan konfigureres meget nemt, hvis man f.eks. ønskede en konstruktion med danske ord i stedet for engelske. Dette er nærmere forklaret i manualen til `algorithmicx`, se [János \(2005\)](#). Her forklares det også hvordan man laver sine egne ekstra makroer, blokke etc.

10.10.3 Flydende algoritmer

Anvender man desuden pakken `algorithm` får man et nyt *float* environment **algorithm** som kan pakkes omkring en algoritme beskrivelse, hvorefter det flyder omkring på samme måde som ved *figure* eller *table*.

algorithm

Pakken `algorithm` kan som pakke-options tage `boxed`, `ruled` og `plain` som alle påvirker hvordan den flydende algoritme beskrivelse ser ud.

Algoritmerne i sig selv kan sagtens deles hen over en side, men det kan en float jo ikke så her kan man blive nødt til at dele beskrivelsen i to. Men dette kan `algorithmicx` faktisk også finde ud af, se [János \(2005\)](#), afsnit 2.6.

10.11 Interne hyperlinks

Til manualer og andre dokumenter som publiceres via nettet er hyperlinks i PDF yderst anvendeligt.

Man skal anvende pakken `hyperref` og det anbefales at man sørger for at dette er en af de sidste pakker man loader (så senere pakker ikke kommer til at overskrive noget). I de fleste tilfælde vil man kunne nøjes med

hyperref

Eksempel 10.38

```
\usepackage[colorlinks]{hyperref}
```

Dette vil nu automatisk lave alle krydsreferencer, citeringer, indholdsfortegnelsen og indekset om til at være interne hyperlinks som man nu kan klikke på i PDF.

Bemærk at alt efter hvilken kompilerings proces man anvender (latex plus dvips, latex plus dvipdfm eller pdflatex) så kan det være nødvendigt at angive en driver, se [Rahtz og Oberdiek \(2004\)](#) for mere information.

Hyperref manualen vil desuden forklare om de forskellige konfigurationsmuligheder man har.

Sidebemærkning 10.10. Der er yderligere følgende bemærkninger:

- (a) Anvender man `\url` fra `url`-pakken, så vil disse blive lavet om til eksterne URLer, som PDF læseren så kan sende videre til en browser. Da `\url` vil bryde URLen hvis den bliver for lang er det en god ide at hvis man anvender `hyperref` så skal man bruge optionen `breaklinks`. Hvis man desuden går vejen om `latex + dvips + ps2pdf` skal man desuden anvende pakken `breakurl` efter `hyperref`. Denne vil sørge for at for brudte eksterne URLer vil alle delene af URLen pege på det samme eksterne mål, dette sker automatisk hvis man anvender `pdflatex`, så der skal man altså *ikke* bruge pakken `breakurl`. Et muligt setup vil så være

Eksempel 10.39

```
\usepackage{url}
\usepackage[... ,breaklinks]{hyperref}
\usepackage{ifpdf}
\ifpdf
% pdflatex
\else
\usepackage{breakurl}
\fi
```

- (b) Har man i sit dokument eller i sine makroer anvendt makroen `\addcontentsline` (se afsnit 14.3 på side 345), så kan man få problemer med Hyperref, da hyperlinks skal have noget at »holde fast i«. Her skal man i stedet anvende

Eksempel 10.40

```
\phantomsection
\addcontentsline...
```

`\phantomsection` er normalt bare en tom makro (*memoir* har den f.eks. med), men den omdefinieres passende af `hyperref`.

Der kan også være mange andre problemer, se evt. [Rahtz og Oberdiek \(2004\)](#).

Det er en meget god ide at læse den README fil som følger med `hyperref`, her kan man læse om mulige løsninger vedrørende brugen af `hyperref` i forbindelse med andre pakker, f.eks. `varioref`.

Tip 10.11. Givet et stort projekt i PDF-format, fremvist i Adobe Reader, så har AR et lille vindue hvor den viser et 'sidetal'. Har man en indledning nummereret med romertal, vil man opdage at projektets sidetal og de sidetal AR viser i den lille oversigt, ikke stemmer overens. Dette kan man fikse med pakke-option »plainpages=false« til `hyperref`. Så vil det lille vindue i stedet vise »projektsidetal (fysiskside of totalt antal fysiske sider)«. Meget anvendeligt, hvilket vi også anvender til PDF-udgaven af denne bog.

Interne hyperlinks

Tip 10.12. Som standard laver hyperref bookmarks. Til tider (især hvis man anvender `\part`) vil placeringen i bookmarktræet være en smule forkert. Her kan pakken `bookmark` hjælpe, pakken kan findes på CTAN som en del af oberdiek-bundtet.

bookmark

Dette kan man så anvende således at f.eks. litteraturlisten ikke længere er en del af den sidste `\part` i dokumentet.

Tip 10.13. Når man oversætter med `pdflatex`¹⁴ kan man til tider se følgende warning:

```
Token not allowed in a PDF string (PDFDocEncoding):
```

efterfulgt af noget mere tekst. Dette skyldes at man har skrevet noget i en overskrift som ikke uden videre kan skrives til bookmarklisten i en PDF-fil. Man kan enten vælge at ignorere det, eller hjælpe hyperref på vej med

```
\texorpdfstring{<til teksten>}{<til bookmarks>}
```

omkring de 'besværlige dele'. Eksempelvis

```
\section{\texorpdfstring{ $\chi^2$ }{chi2}-fordeling}
```

Eksempel
10.41



Personligt har jeg det med ikke at anvende hyperref, mens jeg skriver. Men jeg vil gerne fortsat kunne anvende `\texorpdfstring`. Dette gør jeg ved at have følgende placeret *efter* det sted i min preamble hvor jeg senere indlæser hyperref

```
\providecommand\texorpdfstring[2]{#1}
```

Eksempel
10.42

Så vil makroen kun blive tilføjet hvis den ikke allerede findes.

¹⁴ Går ud fra at de fleste anvender `pdflatex`.

Vedrørende større projekter

Håndtering af større projekter

Man vil som studerende komme ud for at skulle lave mindst én større skriftlig opgave, hvilket man naturligvis anvender L^AT_EX til. Det er dog noget værre administrativt rod at have ti kapitler af et speciale på 100 sider liggende i samme, meget lange fil. Vi skal i dette kapitel se på hvordan man kan opdele kildekoden, samt vælge kun at arbejde på dele af den ad gangen. Kapitlet vil også indeholde tips til hvordan man kan holde styr på sine labels, skrive noter til sig selv, eventuelt referere til eksterne labels i eksterne dokumenter.

1

11.1 Opdeling af kildekoden

For ikke at have meget store, uoverskuelige filer er det smartest at opdele kildekoden i mindre filer, hvor man så f.eks. har et hoved/master-dokument som inkluderer de forskellige dele.

Ideen er at man f.eks. placerer hvert kapitel i hver sin .tex-fil.

Sidebemærkning 11.1. Husk at en eventuel `\chapter` skal være i filerne som inkluderes, de skal *ikke* være i masterfilen.

Man inkluderer så hvert kapitel (eller hvad det nu er) med en passende kommando, L^AT_EX tilbyder to forskellige makroer til dette, `\input` og `\include` med hver deres fordele og ulemper.

`\input`
`\include`

Sidebemærkning 11.2 (Ignorerer resten af en inkluderet fil).

For begge konstruktioner vil vi lige bemærke at ønsker man at ignorere den *resterende* del af den inkluderede fil, da skal man blot indsætte `\endinput`. Både `\input` og `\include` inkluderer indholdet af filen enten indtil slutningen af filen, eller til den første »`\endinput`« de møder. *Alt* efter `\endinput` ignoreres.

`\endinput`

Sidebemærkning 11.3. I visse ældre projekter, jeg har haft fornøjelsen af at redigere lidt på, ser man nogle gange forfattere som starter disse inkluderede filer med at sætte kapitelnumre eller manuelt at nulstille formelnumre.

Dette er slet ikke nødvendigt i vore dage. Skulle man have brug for det, så hører den slags til i master-filen, så kan man hurtigere tilrette dem, hvis det bliver nødvendigt.

Lad os nu ser nærmere på `\input` og `\include`.

¹ Fixme Note: Her er sikkert mange andre ting vi kan tilføje, ideer?

11. Håndtering af større projekter

`\input`

Når L^AT_EX når til et `\input{<filnavn>}`-kald vil den læse og fortolke indholdet af `<filnavn>` som om dette slet ikke var i en ekstern fil.

– og hvad så?

Indrømmet, ovenstående er ikke den bedste forklaring, men når vi i det næste afsnit forklarer om `\include` vil man se at der er meget stor forskel på hvad de gør.

Angiver man »`<filnavn>`« uden angivelse af filendelse så vil L^AT_EX automatisk forsøge at inkludere filen »`filnavn.tex`«. Hvis man ønsker at inkludere indholdet af en fil som ikke har `.tex` som filendelse, så skal »`<filnavn>`« altså angives med endelse, f.eks. »`tabel.tbl`«.

Man kan uden problemer indlæse filer inden i andre filer (dette kaldes nesting).

Som eksempel kan nævnes at man kan have hele sin preamble i en ekstern fil som så hentes ind via `\input`. Der er intet problem i at have et hoveddokument som ser ud som dette (dette *er* hele dokumentet):

Eksempel 11.1

```
\input{preamble}
\begin{document}
\input{kapitel1} ...
\input{kapitel5}
\end{document}
```

Det kan dog vise sig at være en smule bøvlet i nogle situationer.

`\include`

Her er så forklaringen på at vi bevidst ikke anvendte ordet *inkludere* i afsnittet om `\input`.

`\include` Makroen `\include{<filnavn>}` svarer til `\input{<filnavn>}` men med en twist. Der gælder følgende regler

- `<filnavn>` skal svare til en `.tex`-fil og `<filnavn>` skal angives uden filendelse.
- en `\include` starter *altid* på en ny side (i modsætning til `\input` som starter nu og her).²
- Ved `\include` vil der blive lavet en `.aux`-fil for hver inkluderet fil. Dette anvendes til features beskrevet nedenfor.
- Man må *ikke* anvende `\include` inden i en fil som allerede er inkluderet via `\include`, men man må godt anvende `\input` inden i en fil som inkluderes via `\include`.

Fordelen ved `\include` er at man nu kan lave betinget inkludering af f.eks. kapitler. Antag man er ved at skrive en bog (eller et langt speciale) og i dag vil man gerne arbejde på kapitlet som ligger i filen »`kapitel9.tex`«. Da projektet er meget langt, tager det en del tid at oversætte det hele, så man vil gerne bare oversætte med `kapitel9.tex` inkluderet alene. Man kunne naturligvis udkommentere alle de andre inkluderede kapitler, men så vil alle krydsreferencer til disse kapitler være udefinerede, hvilket er træls at se på, desuden vil sidetallene være misvisende. Her er i stedet den smarte metode:

² Det er derfor man skal huske at smide `\chapter` kommandoer ned i den inkluderede fil, ellers får man sideskift efter kapiteloverskriften.

Master/hoved-filen

- Start i stedet med at oversætte *hele* dokumentet med alle filer inkluderet via `\include`, dette sørger for at alle labels nu er at finde i `.aux`-filen hørende til hver inkluderet fil.
- Placér herefter `\includeonly{<kommasepareret liste af filnavne>}` i *preamblen* og oversæt igen. Så vil kun de specificerede kapitler blive oversat, men alle krydsreferencer til øvrige kapitler optræder fint og man har de korrekte sidetal.

`\includeonly`

For eksemplet beskrevet ovenfor ville man altså skulle placere

```
\includeonly{kapitel9}
```

**Eksempel
11.2**

i preamblen.³



Advarsel 11.4. Man bør kun anvende `\include` til at inkludere ting som giver side-output. Man bør heller ikke inkludere f.eks. en tabel via `\include`, anvend i stedet `\input`.



Advarsel 11.5 (`\include` giver ikke fejl hvis fil ikke findes).

Hvis filen man ønsker at inkludere *ikke* findes, vil `\include` kun give en advarsel (warning) og ellers bare gå videre, i modsætning til `\input` som giver en fejl. Denne advarsel skal man altså selv opdage. Bemærk at *findes* som sædvanligt betyder at filen ikke kan findes på et af de steder \LaTeX leder efter filer, hvilket normalt er mere end bare det nuværende bibliotek!

Sidebemærkning 11.6. Anvend aldrig `\end{document}` inden i en fil inkluderet via `\include`, der kan ske meget underlige ting. Ønsker man at stoppe inkluderingen (f.eks. ved debugning) så anvend i stedet `\endinput`.

Se desuden [Mittelbach og Goossens \(2004\)](#) for mere information.

Master/hoved-filen

Masterfilen er et hoveddokument, det er her man så inkluderer sine kapitler etc. Det er også denne fil man kompilerer for at få det samlede dokument.

Sidebemærkning 11.7. I de fleste gode \LaTeX -editorer kan man til en given fil angive hvilken fil der skal anvendes som masterfil, i.e. hvilken fil det er man vil oversætte når man ønsker at compilere sit dokument. (Man kan jo ikke compilere de enkelte kapitelfiler for sig idet de ikke har nogen preamble ej heller en `\begin/end{document}`, hvilket de heller ikke skal have.) Dette betyder at man bliver fri for at skulle skifte over til sin masterfil hver gang man ønsker at compilere.

Nogle editorer kalder en sådan konfiguration for et *projekt* eller lignende. Desværre er der også nogle editorer som ikke er videre gode til at håndtere større projekter.

³ I master filen til denne bog, har vi listet alle kapitlerne i en samlet `\includeonly`, og udkommenterer så dem vi ikke skal bruge.

11. Håndtering af større projekter

Sidebemærkning 11.8. Bare for god ordens skyld minder vi om at de kapitler som man har i eksterne filer altså *ikke* skal have nogen preamble ej heller noget `\begin{document}... \end{document}`-par.

En simpel skabelon til en masterfil kunne se sådan ud (visse af makroerne forklares senere):

```
\documentclass[options]{dokumentklasse}
<øvrige preamble>
%\includeonly{kapitel2}
\begin{document}
\include{forside}
\cleardoublepage
\frontmatter
\include{abstrakt}
\cleardoublepage
\tableofcontents
\include{indledning}
\mainmatter
\include{kapitel1}
\include{kapitel2}
...
\include{kapitelX}
\appendix
\include{appendiks1}
\bibliographystyle{style}
\bibliography{bibfil}
\printindex
\end{document}
```

11.2 Ekskludering af dele af teksten

Når man er i gang med større projekter vil skriveprocessen sjældent være lineær og ofte vil man få brug for at kunne fjerne/ignorere passager uden at slette dem helt. Den slags har man selvfølgelig også pakker til.

Sidebemærkning 11.9. Skulle man have brug for at ignorere slutningen af en fil som er inkluderet via `\input` eller `\include`, så kan man bare indsætte en `\endinput`, så slutter inkluderingen på det sted hvor man har placeret `\endinput`. Det er lige som med `\end{document}` alt efter `\endinput` ignoreres.

Sidebemærkning 11.10. Mange L^AT_EX orienterede editorer tilbyder muligheden for at man kan afmærke et stykke tekst og så få denne udkommenteret med % i starten af linierne. Som regel har man også muligheden for at markere et stykke udkommenteret tekst, og så få fjernet udkommenteringen.

Dette har en stor fordel fremfor metoden beskrevet herunder, man er slet ikke i tvivl om at noget er udkommenteret, tit vil editoren endda give den udkommenterede tekst en speciel farve.

Arbejdscommentarer

Har man ikke mulighed for at anvende betoden beskrevet i Bemærkning 11.10 på **modstående side**, eller ikke har lyst til at anvende den, så kan man anvende pakken `comment` (se Eijkhout (1999)). Den stiller umiddelbart environmentet *comment* til rådighed, om hvilket der gælder to regler

comment

- (i) `\begin/end{comment}` skal hver især stå på en linie for sig selv.
- (ii) der må *ikke* være noget mellemrum før `\begin/end{comment}` og der må heller ikke stå noget efter.

<pre>\usepackage{comment} ----- slut preamble ----- Noget tekst \begin{comment} noget som ikke kommer med \end{comment} og noget tekst efter kommentar.</pre>	Noget tekst og noget tekst efter kommentar.
---	---

Eksempel
11.3



En anden smart feature ved `comment` er dog at man selv kan definere sine egne specielle kommentar environments som så kan slås til og fra efter for godt befindende.

```
\includecomment{<env-navn>}
```

`\includecomment`

laver et *kommentar* environment som er ‘usynligt’, forstået på den måde at den *ikke* ignorerer noget (teksten man skriver indenfor dette environment bliver skrevet som hvis environmentet ikke havde været der). Dette er meget nyttigt til at lave kommentarer og lignende til sig selv som man senere nemt kan ekskludere.

```
\excludecomment{<env-navn>}
```

`\excludecomment`

som ovenfor men nu vil alt mellem `\begin{env-navn} . . . \end{env-navn}` blive ignoreret. Dvs. med

```
\includecomment{kommentar}
%\exclludecomment{kommentar}
```

Eksempel
11.4

ville kommentarens indhold blive skrevet, mens med

```
%\includecomment{kommentar}
\exclludecomment{kommentar}
```

Eksempel
11.5

ville den være ignoreret.

En rigtig skummel brug af dette er at man med

```
\excludecomment{proof}
```

Eksempel
11.6

kan fjerne alle beviser i sit dokument.

11.3 Arbejdscommentarer

Man får sikkert også brug for at kunne skrive kommentarer til sig selv (hvad mangler der her, hvad er problemet vi ikke kan løse. . .).

Der er umiddelbart to pakker som er relevante her: `fixme` og `todonotes`. Den første er meget tekstorienteret, hvorimod den anden er mere grafisk gennem brugen af `TiKz`.

fixme
todonotes

11. Håndtering af større projekter

Pakken *fixme*

Pakken *fixme* giver som navnet lidt antyder mulighed for at placere forskellige noter i teksten som man så senere kan vende tilbage til. Blandt andet har man mulighed for at indsætte en samlet liste af ting som man skal have kigget nærmere på.

Sidebemærkning 11.11. I version 3 af *fixme* kunne man nemt skrive i marginen via pakkens interface. I den nuværende version 4 af pakken er dette markant forringet. Herfra kan anbefales at man i stedet konfigurerer *fixme* til at skrive noterne som fodnoter, se nedenfor.

Pakken indlæses som sædvanligt via

```
\usepackage[options]{fixme}
```

hvor *options* bl.a. inkluderer *final*, *draft* og *danish*. Vi kommer senere tilbage til hvad de betyder. Nogle af makroerne man anvender i teksten er:

```
\fxnote[type]{tekst}
\fxwarning[type]{tekst}
\fxerror[type]{tekst}
\xfxfatal[type]{tekst}
\listoffixmes
\fxsetup{options}
```

Man anvender normalt pakke optionen »*draft*« hvorefter de fire første makroer vil skrive deres tekst som en del af dokumentet, f.eks. i marginen eller evt. som fodnoter, alt efter hvilket setup man anvender. Bruger man *ikke* *draft*, så er det det samme som »*final*« og så forsvinder resultatet af de tre første makroer, og hvis der er flere *\xfxfatal* så vil dokumentet ikke længere oversætte (deraf navnet).

Eksemplet herunder viser simpel brug af *fixme*.

Eksempel 11.7



```
\usepackage[draft,danish]{fixme}
\fxsetup{layout=margin} % footnote anbefales, men dårligt til eksempel
----- slut preamble -----

\small
noget tekst \xfxfatal{her er noget galt}
mere \fxnote[inline]{en note}
mere \fxwarning{en advarsel -- noget er helt forkert}
mere \fxerror[inline]{en fejl}
\listoffixmes
```

noget tekst mere **FiXme Note: en note** mere mere **FiXme Fejl: en fejl**

Rettelser

Dødelige: her er noget galt	1	FiXme Dødelige: her er noget galt
Note: en note	1	FiXme Note: en note
Advarsel: en advarsel – noget er helt forkert	1	FiXme Advarsel: en advarsel – noget er helt forkert
Fejl: en fejl	1	FiXme Fejl: en fejl

Observationer:

Pakken todonotes

- (i) `inline`-optionen (her givet til visse af makroerne) betyder at noten skrives direkte i teksten, mens uden skrives den direkte i marginen (alt efter hvad man konfigurerer den til).
- (ii) `\listoffixmes` laver en oversigt over de ting man mangler at få fikset.
- (iii) Margin noterne sættes *raggedleft*, i.e. højrejusteret uden orddeling.
- (iv) Holder man øje med `.log`-filen eller outputtet fra L^AT_EX i et terminalvindue vil man også se at den laver en oversigt over de manglede ting man har markeret med *fixmes*.

Pakken *fixme* har også mulighed for at lave større kommentarer via forskellige environments, se [Verna \(2009\)](#) for flere oplysninger. Desuden kan man lave makroer således at det kan knyttes en forfatter til hver *fixme* notits, se pakke manualen for mere information.

Pakken todonotes

Her vil vi bare komme med et eksempel og i stedet henviser til den glimrende manual, [Midtiby \(2009\)](#). Bemærk i eksemplet den specielle makro `\missingfigure` som kan være ret nyttig under skrivefasen.

```
\usepackage[danish]{todonotes}
```

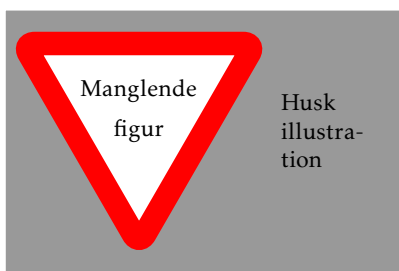
— slut preamble —

```
Dette er noget tekst\todo{Skal lave et eksempel med todonotes}, hmm
tror vi holder her.
\begin{figure}[ht]
  \centering
  \missingfigure[figwidth=6cm]{Husk illustration}
\end{figure}
```

Eksempel
11.8



Dette er noget tekst, hmm tror vi holder her.



Skal lave et eksempel med todonotes

11.4 At holde styr på labels og lignende

Vis indsatte og anvendte label- og citeringsnøgler

Pakken *showkeys* kan hjælpe med at holde styr på de mange labels man ender op med at have i et stort projekt.

```
\usepackage{showkeys}
```

— slut preamble —

```
i formel (\ref{a}) ses at \dots
\begin{equation}
\label{a}
a^2+b^2=1
\end{equation}
```

i formel ([1](#)) ses at ...

$$a^2 + b^2 = 1 \quad (1) \quad \boxed{a}$$

Eksempel
11.9



11. Håndtering af større projekter

Pakken vil så skrive alle labels ude i marginen (faktisk i begge marginer) indrammet i en kasse, alle steder hvor der er refereret eller citeret vil den label som er refereret til blive skrevet hen over referencen. Ofte ønsker man kun at få skrevet alle labels op i marginerne, hvilket kan gøres via

Eksempel `\usepackage[notref,notcite]{showkeys}`

11.10

Ønsker man noget som ikke er så dominerende kan man tilføje optionen `color`.

`showidx` Har man lavet et indeks til sit dokument kan pakken `showidx` også være meget rar,
`\index` den gør det samme for `\index` som `showkeys` gør for f.eks. `\label`.

Lidt smartere krydsreferencer

Når man nu f.eks. anvender `floats`⁴ til sine figurer kan man jo ikke længere sige sådan noget som: *På figuren herunder...*, thi man ved ikke hvor figuren havner. Man bliver derfor nødt til at henvise til figurnummeret og hvis man er meget venlig også et sidetal. Men hov der er jo et potentielt problem – hvad nu hvis man laver en sidehenvielse til en figur og figuren så faktisk havner på samme side som henvisningen. Det ser ikke ret pænt ud. Løsning: Anvend pakken `varioref`. Man vil oftest anvende den med dansk opsætning⁵:

```
\usepackage[danish]{varioref}
```

`\vref` Man anvender så `\vref` i stedet for `\ref`. Det kunne så give:

Eksempel
11.11

Dette er en henvisning til
eksempel `\textbf{\vref{fixme1}}`

Dette er en henvisning til
afsnit `\textbf{\vref{sec:mark-nomk-noidxsh}}`.

Til slut en henvisning til dette
eksempel: `\textbf{\vref{vari01}}`

Dette er en henvisning til eksempel **11.7**
på side 296

Dette er en henvisning til afsnit **11.4**
på forrige side.

Til slut en henvisning til dette eksem-
pel: **11.11**

Man skal bemærke forskellen i den fremhævede tekst.

`\vrefrange` Pakken har mange andre specielle muligheder, f.eks. `\vrefrange{<start label>}{<slut label>}` som kan anvendes til at angive en hel samling af referencer.

Eksempel
11.12

Kapitel `\vref{cha:satninger-i-latex}`
indeholder eksemplerne:
`\vrefrange{thm1}{optimering2}`.

Kapitel **5 på side 115** indeholder eksem-
plerne: **5.3 til 5.33 på side 116–133.**

`\vpageref` Der er desuden en `\vpageref` svarende til `\vref` bare kun omhandlende en henvisning til sidetallet. Den interesserede læser henvises til **Mittelbach (2004)** for mere information.

Det er faktisk også muligt at lave specielle henvisningsmakroer som automatisk kan indsætte f.eks. *formel* (3.39) etc. Det er ikke noget vi vil komme nærmere ind på her, men i stedet henvise enten til **Mittelbach (2004)** (`varioref`) eller måske mere relevant til pakken `fancyref` (se **Reichert (1999)**), alternativt kan pakken `cleveref` anvendes, se dog Bemærkning **11.14 på næste side.**

⁴ Se kapitel 6.

⁵ Hvilket netop er en god grund til at angive `danish` som option til dokumentklassen, så skal man ikke angive den til diverse andre pakker, det sker automatisk.

Referere til label fra et andet projekt

Sidebemærkning 11.12. Det skal lige bemærkes at brugen af `varioref` kommer med en pris. Nemlig at tingene kan gå temmeligt meget i ring hvis en `varioref` reference kommer omkring et sideskift. Dette har jeg for eksempel haft problemer med i denne bog. Man vil i så fald få en fejl om at der forekommer en `varioref`-reference hen over et side skift.

Her er der ikke ret meget andet at gøre end at forsøge at finde det relevante sted og så skrive teksten lidt om eller indsætte en `\newpage` på et passende sted.

Sidebemærkning 11.13 (Vedr. `fancyref`).

Pakken `fancyref` implementerer en `\fref` kommando, det gør *memoir* også, så her kan det være en ide at load `fancyref` via

```
\let\fref\relax
\usepackage{fancyref}
```

Eksempel
11.13

Sidebemærkning 11.14 (Vedr. `cleveref`).

Bemærk at `pt.` er pakken `cleveref` *ikke* kompatibel med *memoir*, samt en del andre pakker. `Cleveref` overtager helt `\label` og hvad den skriver til `.aux`-filen. Dette ødelægger *memoirs* `\titleref` feature.

6

11.5 Referere til label fra et andet projekt

Nogle gange får man brug for at kunne referere til en label i et helt andet projekt, altså i filer som ikke berøres mens man oversætter sit nuværende dokument. Dette kan f.eks. anvendes i `sldies` når man ønsker at referere til en bestemt sætning i sin afhandling.

Dette kan håndteres via pakken `xr`. Man specificerer så de eksterne filer via (evt. `xr` flere af disse)

```
\externaldocument{<filnavn>}
```

\externaldocument

`<filnavn>` skal svare til den pågældende `filnavn.aux`-fil, uden endelsen. Herefter kan man så referere til labels fra `<filnavn>`. Dog vil det være en god ide at kunne skelne mellem labels i det nuværende dokument og labels fra det eksterne dokument (f.eks. hvis der er sammenfald). Dette undgår man ved at angive et prefix til `\externaldocument` f.eks.

```
\externaldocument[A-]{filnavn}
```

Eksempel
11.14

Hvorefter man så skal referere via nøglen »A-`<nøgle>`« for netop dette dokument. Man kan mere eller mindre have alle de eksterne dokumenter man har lyst til. Se evt. [Carlisle \(2006\)](#).

En brug af dette er f.eks. hvis man er ved at lave slides til sit forsvar, så kan man anvende `xr` til direkte at referere til sætninger, formler og sidetal i afhandlingen.

Paul M. Bendixen har nævnt at der indenfor f.eks. ingeniørstudiet er krav om at en projektrapport og projektdokumentation faktisk er to separate dokumenter. Så her er `xr` igen nyttig således at man kan krydsreferere mellem de to dokumenter.

⁶ Fixme Note: Der bør skrives noget vedr. `cleveref`, `fancyref` og venner

11. Håndtering af større projekter

Sidebemærkning 11.15. Insisterer man på også at anvende hyperref, så er det en god ide at supplere xr med pakken xr-hyper, se evt. <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=extref>.

Dokumentopsætning med memoir-klassen

Det blev i indledningen nævnt at vi mener at man bør behandle især specialer som om det faktisk er en bog. Bøger har en speciel opbygning som gør at det faktisk er nogenlunde nemt at finde ud af hvor de enkelte ting skal være. Dette kapitel handler om opbygningen af bøger samt hvordan vi får dette implementeret med dokumentklassen *memoir*. Vi vil i denne henseende anvende ordet »*bog*« som dækkende over alle typer af større opgaver så som specialer, bachelorprojekter, ph.d.-afhandlinger samt store gymnasieopgaver.

Vi vil i første omgang behandle specialer (opsætningen til ph.d.-afhandlinger er den samme) da den ligner bøger mest. I slutningen af kapitlet vil vi komme ind på hvordan man ændrer skabelonen således at den i stedet passer til bachelor-, gymnasie- og andre projekter.

Det er en god ide at man sørger for at gøre sig bekendt med den filmæssige opbygning af større projekter, se afsnit [11](#) på side [291](#).

12.1 Pakker allerede inkluderet i memoir-klassen

Memoir-klassen har inkluderet funktionaliteten (men ikke altid syntaksen) fra følgende pakker (listen er *ikke* komplet), se punkterne (1)–(42) herunder. Funktionaliteten er i mange tilfælde enten udvidet i forhold til de nævnte pakker, eller anvendt med en anderledes syntaks. Er man i gang med at migrere fra f.eks. et layout baseret på *book*-klassen, og anvender man en af nedenstående pakker, så kan det nemt være således at man bliver nødt til lige at finde ud af hvordan man laver det tilsvarende med *memoir*. Det mest almindelige eksempel på noget som skal laves om er brug af fancyhdr-pakken (*memoir* er meget bedre til pagestyles).

(1) array	(12) shortverb	(23) chngcntr	(34) lastpage
(2) delarray	(13) tocvsec2	(24) tocbibind	(35) geometry
(3) tabularx	(14) epigraph	(25) tocloft	(36) endnote
(4) dcolumn	(15) nextpage	(26) verse	(37) setspace
(5) booktabs	(16) needspace	(27) ifmtarg	(38) parskip
(6) framed	(17) abstract	(28) fancyhdr	(39) newfile
(7) enumerate	(18) chngpage	(29) crop	(40) patchcmd
(8) remreset	(19) appendix	(30) sectsty	(41) titleref
(9) subfigure	(20) ccaption	(31) titlesec	(42) ifpdf
(10) verbatim	(21) hanging	(32) footmisc	
(11) moreverb	(22) titling	(33) index	

12.2 Opbygningen af en bog

I dette afsnit vil vi kort ridse op hvad et boglignende projekt (normalt) betstår af.

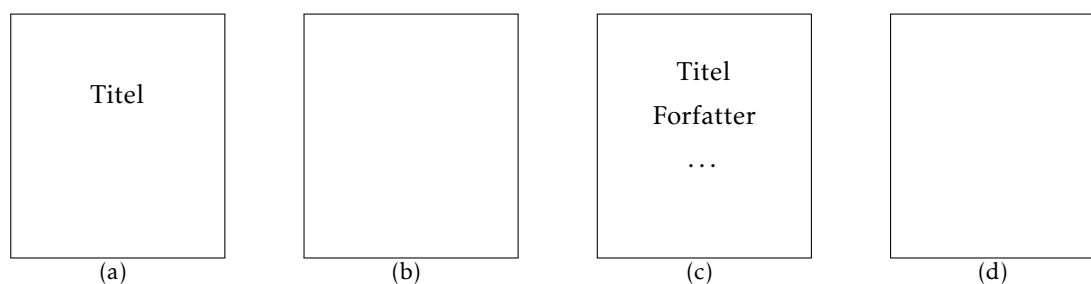
En bog består generelt set af fire dele.

- (a) Titelmateriale,
- (b) frontmatter,
- (c) mainmatter samt
- (d) backmatter.

Titelmateriale

Titelmaterialet dækker over alt hvad der hører til forsiden af bogen. Åbner man en almindelig bog (dog ikke de fleste forelæsningsnoter¹) ser man ofte, først en side hvor der bare står titlen på bogen, bagsiden af denne kan så rumme forskellige oplysninger (denne information kan også placeres bagpå selve forsiden). Herefter kommer så selve forsiden hvor man kan se titlen og forfatteren samt anden relevant information. Se figur 12.1.

forsiden



Figur 12.1: (a) er dækbladet, (b) bagsiden af dækbladet, (c) selve forsiden og (d) bagsiden af forsiden.

dækblad

Arket med titlen alene som kommer før selve forsiden kaldes et *dækblad* og har kun til opgave at beskytte forsiden. Dækblade er nok ikke særligt anvendt til specialer og bachelorprojekter, men skal bogen bindes ind (f.eks. med pap) er det almindeligt at anvende et *dækblad*.

Sidebemærkning 12.1. Når man har at gøre med noget som f.eks. skal indbindes med pap, så kan man enten gentage forsiden på dette omslag (ofte justeret lidt i forhold til marginer), eller man kan lave en specifik forside til omslaget. Denne form for design ser man ofte ved bøger, et eksempel ses f.eks. ved [Jensen \(2007\)](#).

Bagpå dækbladet eller forsiden kan man inkludere informationer omkring denne bog.² Man kan eksempelvis skrive noget om at dette er »Copyright 2011 *<navn>*« og noget om hvordan bogen er lavet. Når man nu alligevel har anvendt L^AT_EX til at sætte kronen på det værk man har været i gang med at skrive, så hvorfor ikke lade læseren vide at det var L^AT_EX som gjorde dette så pænt. Man kan jo også tilføje noget om hvordan eventuelle figurer er lavet. Denne (meta)-information kaldes ofte for en *kolofon*.

kolofon

¹ Endnu, er ved at indoktrinere mine lokale note-forfatterne.

² Hent inspiration fra andre bøger.

IMF krav til forsiden

Forsiden og det eventuelle dækblad nummereres normalt ikke og inkluderes normalt ikke i indholdsfortegnelsen.

3

IMF krav til forsiden

Sidebemærkning 12.2. I reglerne for Naturvidenskabelige Fakultet, AU, kan man læse at der er krav om at der skal indgå et resumé på engelsk samt at titlen foreligger på dansk og på engelsk. Der er ikke noget krav om at man har et dansk resumé, ej heller hvis man skriver på engelsk. Der er heller ikke noget krav om at dansk og engelsk titel begge skal fremstå på forsiden. Skriver man f.eks. på dansk, står det en frit for at nævne den engelske titel som en del af det påkrævede engelske resumé.

Ved Institut for Matematik, Aarhus Universitet har vi følgende krav til hvad der skal fremgå af forsiden af *specialer*. Hvordan resten så stilles op er helt op til den studerende. Senere i dette kapitel skal vi se to eksempler (Louises henholdsvis Strits forsider, navngivet efter de studerende de oprindeligt blev lavet kreeret for). På grund af omlægning af ph.d.-uddanenserne vides det pt. ikke hvordan kravene til ph.d.-afhandlinger bliver. For specialer projekter skal forsiden indeholde følgende:

- (i) Navn og årskortnummer på forfatteren.
- (ii) Titlen på værket.
- (iii) Navn på vejleder(e).
- (iv) Hvornår der er afleveret, kan f.eks. være måned og år.⁴
- (v) Hvilken type opgave der er tale om og i hvilket område. Dvs. om det er en specialafhandling i matematik, statistik eller matematik-økonomi. Forskningsgrupper nævnes ikke på forsiden. Denne regel er egentlig mest af hensyn til biblioteket, så de ved hvor materialet skal placeres, og reglen gælder kun for specialer.

Sidebemærkning 12.3. For bachelorprojekter er der så vidt vides ikke nogen specifikke regler, listen herover vil sikkert være en god guide.

Undersøg selv hvilke regler der gælder på dit institut/afdeling/gymnasie/skole.

Frontmatter

Frontmatter er et begreb⁵ som dækker over alle de indledende ting før man egentligt starter på selve indholdet af værket. Det er især her specialer og bachelorprojekter har specielle punkter som kan placeres her. Frontmatter indeholder normalt ting så som:

- (i) Abstrakt/resumé⁶ – hvor i rækkefølgen det skal placeres, er noget man skal drøfte med sin vejleder, samt konsultere instituttets regler (hvis de har sådanne).
Resumeet står normalt på en (ulige) side for sig selv (med blank bagside, hvis det altså fylder under en side). Man kan sagtens have et dansk og et engelsk abstrakt efter hinanden, se [Eksempel 12.22 på side 322](#).

³ Fixme Note: tjek lige hvor det er man anbefales at placere en dedikation.

⁴ Fixme Note: denne er jeg ikke helt sikker på er et krav

⁵ Kender ikke noget dansk ord for det

⁶ Bemærk at *book*-klassen faktisk ikke har noget *abstract*-environment, det har *memoir* til gengæld.

12. Dokumentopsætning med memoir-klassen

- (ii) Indholdsfortegnelse samt eventuelle oversigter over figurer og tabeller (`\listoffigures` og `\listoftables`)
- (iii) Man kan placere en symbolliste under frontmatter, den kan dog også placeres under backmatter i slutningen af bogen.
- (iv) Indledning og/eller opsummering. Da indledningen ofte ikke indeholder noget som anvendes senere i specialet bør dette placeres under frontmatter.
- (v) Taksigelsen, det er altid vigtigt at man takker alle de som har hjulpet med at lave denne bog, om det så bare er medstudenrede som har gjort at man ikke gav op.⁷ Det er i hvert fald god tone at takke sin(e) vejleder(e).
- (vi) Dedikation, eller tilegnelse. I [Chicago \(2003\)](#) anbefaler de at dedikationen havner på side »v«, men det er sikkert også noget man selv kan bestemme. Bemærk at det er ikke nødvendigt at skrive »Dedikeret til...«, »Til...« er nok. Men det betemmer man selv.

Der er tradition for at frontmatter nummereres med små romertal. Ofte undertrykkes sidetallet på siderne med abstrakt eller en eventuel dedikation. Samtidigt er der tradition for at man ikke nummererer afsnit eller kapitler indenfor frontmatter, disse bør dog i et vist omfang stadigvæk fremgå af indholdsfortegnelsen. Dette klarer man meget nemt med en systemkonfiguration. Se afsnit [12.3.7 på side 320](#).

Mainmatter

Mainmatter indeholder så selve teksten, normalt inddelt i kapitler (måske lidt anderledes vedrørende bachelorprojekter). Mainmatter nummereres med almindelige tal og første side starter fra nummer 1 (dvs. sidetælleren nulstilles ved indgangen til mainmatter). Side 1 behøver ikke være første side af et kapitel, det kan også være en `\part`-side.

Sidebemærkning 12.4. Der er mange som bliver forvirrede over dette med at indledningen er en del af frontmatter og ikke mainmatter.

Den nemmeste måde at forklare det er at indledningen ofte skal ses som en opsummering af selve værket. Man kan f.eks. forklare hvad der kommer til at ske i de enkelte kapitler, opsummere de vigtigste resultater, etc.

Hvis det man har skrevet som introducerende materiale inkluderer ting som anvendes senere (f.eks. definitioner som ikke specificeres andetsteds), så er det i og for sig ikke nogen indledning man har skrevet, så er det en introduktion og så skal den placeres indenfor mainmatter.

Backmatter

Backmatter indeholder de forskellige former for efterskrift man måtte inkludere. Dette kan omfatte:

- (a) Appendikser/bilag,
- (b) symbolliste (hvis man ikke har placeret den i frontmatter),
- (c) slutnoter (hvis man har valgt dette i stedet for/sammen med fodnoter),
- (d) litteraturliste,

⁷ Skulle undertegnede være nævnt i indledningen af de dokument man er ved at skrive, modtager jeg gerne en printet kopi til min trofæhylde. Så kommer den til at stå der for andre at søge inspiration fra.

- (e) indeks og til allersidst (hvis ikke man har placeret den bagpå forsiden)
- (f) kolofonen.

Appendikserne/bilag kommer normalt først, så symbolliste og slutnoter, litteraturliste og indeks. Nogle vælger at have både appendikser samt bilag.

Sidenummereringen fortsætter fra mainmatter. Bemærk at der er tradition for at kapitlerne i appendikset nummereres anderledes end i mainmatter – ofte med store bogstaver (A, B, C, ...) eller med store romertal. Hvis man anvender både bilag og appendices, så kan man anvende store bogstaver til den ene del og store romer tal til den anden, se afsnit 12.5 på side 328.

12.3 Specialelayout med memoir – casestudy

8

Dokumentklassen *memoir* er sammen med klasserne i KOMA⁹ bundtet nok de bedste og mest konfigurerbare dokumentklasser tilgængelige i dag. Man skal se *memoir* som en erstatning for dokumentklassen *book* idet den anvender `\chapter` som øverste primære afsnitsniveau. Men klassen kan om nødvendigt emulere *article*.

Memoir-manualen (Wilson, 2010) er stor og omfattende, dvs. den forklarer hvordan man konfigurerer alle aspekter af dokumentdesignet – vel at mærke stort set uden brug af (mange) eksterne pakker! Ergo er *memoir*-klassen sammen med manualen det eneste man har brug for at justere layoutet til sit dokument.

Den nemmeste måde at løse et memoir relateret problem/opgave er at slå op i indholdsfortegnelsen og kigge i teksten samt på eventuelle eksempler i det pågældende kapitel, eller sende et spørgsmål til nogen som ved mere om det (f.eks. undertegnede).

Da alt hvad man har brug for findes i *memoir*-manualen vil vi i det følgende ikke gå så meget i detaljer, men i stedet bygge det op som et casestudy og så i stedet se på hvordan man løser dette med *memoir*. For yderligere oplysninger henvises til *memoir*-manualen.



Formaning 12.5. Vi går i denne bog ud fra at man *altid* anvender den seneste version af *memoir*-klassen. Dette betyder i praksis at man anvender *memoir* v3.6j v1.61803 6. marts 2011, eller enhver nyere version.

12.3.1 Layoutformulering

Antag vi er blevet pålagt at anvende nedenstående layout-specifikation. Layoutet er *ikke* videre pænt og er *ikke* noget man skal gøre brug af til et virkeligt projekt. Meningen er i stedet at man skal læse diskussionen som følger med de enkelte dele og så ud fra dette samt eksemplerne justere sit dokument. Vi anvender den ikke helt pæne formulering fordi den af pædagogiske grunde er nem at formulere og nem at gå til.

⁸ Fixme Note: side henvisninger til memman skal tjekkes igennem

⁹ Findes på CTAN og er med i alle distributioner, udvikles i Tyskland og skulle være virkelig god. Personligt har jeg ingen erfaring med dem, idet jeg anvender *memoir* til det meste. Men klasserne skulle være lette at gå til og skulle være fint dokumenterede. KOMA-bundet omfatter klasserne *scrartcl*, *scrbook*, *scrlettr*, *scrlettr2* og *scrreprt* som erstatning for standardklasserne *article*, *book*, *report* og *letter*.



Formaning 12.6. Jeg vil gerne benytte lejligheden til at komme med en mindre formaning: Der er rigtigt mange skabeloner i omløb, folk arver/låner fra hinanden, hvilket jo er meget fint.

Disse skabeloner er sjældent videre godt dokumenteret, dvs. kommentarer indsat for at forklare hvad de enkelte dele i skabelonen gør. Man kan endda være så uheldig at støde på skabeloner som skriver noget som er direkte løgn! (Ofte misforståelser.)

Jeg vil derfor opfordre til, at modtager man en skabelon fra en eller anden så bør man sætte sig ned og kigge denne igennem, samt skrive ned hvad de enkelte dele gør. Fjerne det som du ikke får brug for (så er der mindre risiko for at der kommer konflikter). På denne måde får man et overblik over hvad der er i skabelonen, og ønsker man at ændre noget, så har man en ide om hvor det er henne.

Et par ting som man skal være opmærksom på:

- Pakken `graphicx` skal *ikke* specificeres med driveren »dvips« eller »pdftex«, det er totalt unødvendigt. Eneste tidspunkt hvor man skal angive en driver er når man skal bruge `dvipdfmx`, hvor man skal anvende driveren af samme navn.
- Margin justeringer hvor man ændrer længde makroer med `\setlength`, man bør anvende `geometry`, `typearea` eller klassens egne indbyggede features til at sætte marginer med. Dette er især vigtigt hvis man ønsker at ændre noget.

De specifikke layout ting vi vil vise hvordan man løser, er som følger:

Marginer: Top: 3cm, bund: top + 20%, venstremargin skal være 3/4 af højremargin.

Sidehoved og -fod: Sidetallet skal stå ved yderste margin i sidens fod, dette gælder også kapitelforsider og lignende. Sidehovedet skal indeholde kapiteloverskrift på venstresider og section- til og med subsection-overskrifter i det højre sidehoved. Kapitelsidehovedet skal yderligere indeholde ordet Kapitel (eller Appendix) samt kapitelnummeret. Sidehovedet på uligesider skal *ikke* indeholde afsnitsnumre. Begge sidehoveder centrerer.

Overskrifter: Alle overskrifter skrives med fed sans-serif. Alle afsnit (indenfor main-matter) nummereres undtagen `\subsection` og nedefter.

Indholdsfortegnelse: Indholdsfortegnelsen skal kun omfatte (`\part`), `\chapter` og `\section`. Bibliografi, figur/tabeloversigt og et eventuelt indeks skal også fremgå af ToCen. Desuden skal indholdsfortegnelsen have overskriften »Indholdsfortegnelse«.

Forside: Af forsiden skal det fremgå hvilket område værket hører under, navn og årskortnummer på forfatteren samt navnet på vejleder(e).

I det følgende forklarer vi kort hvordan man opnår dette med *memoir*, vi vil i de fleste tilfælde bare henvise til de enkelte sider i *memoir*-manualen for mere information om hvordan man anvender denne konfiguration.

12.3.2 Dokumentets dimensioner

Generelt er det meget svært at sige hvad der ville være gode marginer til et givet projekt. Det kommer an på mange ting. Nogle ønsker f.eks. at placere deres figurer i den ydre margin (noget som ikke er forklaret i denne bog), hvilket naturligvis betyder visse restriktioner for hvor bred selve teksten kan blive. En anden faktor er valget af skrifttype samt størrelsen af denne. En tekstbredde som er fin ved 12 pt, kan nemt være for bred ved 10 pt. I [Wilson \(2010\)](#), kapitel 2, finder man en generel diskussion af marginer anvendt i typografien ned gennem tiderne.

Dokumentets dimensioner

Noget man vil opdage er at man i et to-sidet layout *altid* har større margin ved den ydre kant i forhold til den indre. Dette forvirrer mange, som mener at det burde være omvendt. Men det er altså en meget gammel tradition, som man opfordres til at følge. Den større ydre margin kan jo så anvendes at placere tommelfingeren når man sidder og læser, samt til notater.

Man opfordres også til at passe på med at selve bredden af teksten ikke bliver for lang, da dette bevirker at man bliver forvirret og skal koncentrere sig mere for at forstå teksten. Længden af linierne bør være mellem 60 og 80 tegn per linie i gennemsnit (nogle sætter grænsen lidt lavere). Det kommer naturligvis også an på om man anvender et toøjlet layout.

Er man i tvivl kan man jo forsøge sige med det *gyldne snit*, dvs. således at forholdet mellem højden og bredden af teksten kommer til at svare til tallet $\varphi = (1 + \sqrt{5})/2 = 1.61803\dots$



Memoir-klassen har en funktionalitet svarende til `geometry`-pakken direkte indbygget, om end det er en noget anderledes syntaks. Vi vil anvende denne syntaks til at justere vore marginer.

Opsætningen af dokumentets indre dimensioner (papirstørrelsen specificeres normalt gennem klasseoption, se dog behandlingen af B5¹⁰) kan formuleres på to måder: (1) Man kan specificere hvor meget der skal være mellem kanten af papiret samt kanterne af området hvorpå teksten befinder sig (kaldet *tekstblokken*). Man specificerer så at sige marginerne. (2) Den anden metode vender dette på hovedet. Hvor (1) lader marginerne bestemme størrelsen af tekstblokken, kan man omvendt specificere størrelsen af tekstblokken og så efterfølgende *placere* denne på papiret.

Sidebemærkning 12.7. Begge typer har sine fordele og ulemper. Den første metode er nem at forholde sig til, men ønsker man at flytte lidt rundt på tekstblokken, skal man justere flere steder for at sikre sig at tekstblokken ikke ændrer sig.

Metode (2) er lidt sværere at forholde sig til som nybegynder, men har den fordel at når tekstblokkens højde og bredde ligger fast, så kan man nemt flytte rundt med tekstblokken, indtil man synes man har opnået den bedste placering. Og denne omflytning vil ikke påvirke ombrydningerne. Desuden kan man f.eks. på denne måde specificere at tekstbredden skal svare til et bestemt antal tegn.

Sidebemærkning 12.8. Selv om vi normalt anbefaler at man specificerer marginer aller først i preambelen, så vil vi anbefale at man justerer sine fontvalg lige inden. Dette skyldes at *memoir* kan beregne bredden af et bestemt antal tegn og denne størrelse afhænger naturligvis af skrifttypen.

Den sidste ting som er relevant for marginerne er at højden af tekstblokken *bør* svare til et helt antal linier (rent teknisk er det faktisk blokhøjde = $\backslash\text{topskip} + n \cdot \backslash\text{baselineskip}$ hvor $n \in \mathbb{N}$ og $n + 1$ er antallet af linier i tekstblokken). Men det klarer *memoir* for os, hvilket betyder at højden af tekstblokken ikke nødvendigvis svarer til det vi bad om (man kan selv justere på denne algoritme, se [Wilson \(2010\)](#), slutningen af afsnit 6.6).

¹⁰ FixMe Dødelige: B5 beskrivelse hvor?

12. Dokumentopsætning med memoir-klassen

Konstruktionen vi skal anvende for at klare vores specifikation er forklaret i [Wilson \(2010\)](#), kapitel 6, især afsnit 6.4–6. Så marginerne sættes op via (vi har bare valgt en værdi for højremargin, det var jo ikke specificeret hvad den skulle være):

Eksempel 12.1



```
\setlrmarginsandblock{*}{3.5cm}{0.75} % højre og venstre
\setulmarginsandblock{3cm}{*}{1.2}    % top og bund
\checkandfixthelayout[nearest]       % specifikt valg af højde algoritme
```

Den sidste makro er nødvendig idet *memoir* internt anvender sine egne længdenavne og `\checkandfixthelayout` sørger både for at udregne de ikke specificerede længder samt specificere disse værdier i de almindelige længdenavne som diverse andre pakker forventes at kunne reagere på.

I [Wilson \(2010\)](#), kapitel 6, kan man også se hvordan man kan justere hvor sidehoved og -fod skal placeres inden i top og bundmargin. Fælles for alle disse makroer er at de skal udføres *før* `\checkandfixthelayout`.



Ønsker man alternativt at specificere tekstblokken og dernæst placere den på papiret, så findes løsningen også i [Wilson \(2010\)](#), kapitel 6. Her skal man så anvende

```
\settypeblocksize{<højde>}{<bredde>}{<faktor>}
\setlrmargins{<ryg>}{<kant>}{<faktor>}
\setulmargins{<øvre>}{<nedre>}{<faktor>}
\checkandfixthelayout[nearest]
```

Hvor der så er andre regler for fortolkningen af argumenterne, f.eks. skal man kun angive *ét* af argumenterne til `\setlrmargins` og så lade de andre være »*«. Læs mere i *memoir*-manualen (tabel 6.2 og tabel 6.4).

Som et lille kuriosum kan nævnes at ønsker man at have en tekstbredde svarende til 65 tegn, og et forhold svarende til det gyldne snit, så kan dette gøres via

Eksempel 12.2



```
\setxlvchars[\normalfont] % beregner bredden 65 tegn i nuværende font
\settypeblocksize{*}{\lxvchars}{1.61803} % specifikation af tekstblok
...
\checkandfixthelayout[nearest]
```

(evt. kan man gøre den lidt bredere via f.eks. `1.03\lxvchars`).

12.3.3 Sidehoved og -fod

I *memoir* er der en tættere knytning mellem sidehoved/-fod end det er muligt at gøre i standardklasserne med pakken fancyhdr. Blandt andet har man givet kapitelforsider har deres egen pagestyle, hvis siden før et nyt kapitel er tom, så får den også en speciel pagestyle, alle disse kan nemt konfigureres. Her er nogle af de almindeligt brugbare pagestyles *memoir*-klassen stiller til rådighed:

empty tomt sidehoved og -fod,

plain tomt sidehoved, centreret sidetal i foden,

part anvendes på `\part` sider, normalt alias for **plain**,

chapter anvendes til kapitelforsider, alias for **plain**,

cleared anvendes på de tomme sider før kapitler (o.lign.), alias for **empty**

title anvendes på siden hvor man har anvendt `\maketitle`, alias for **plain**,

titlingpage anvendes indeni *memoir*'s konstruktion til forsideren gennem *titlingpage*-environmentet, se afsnit 12.3.6 på side 315. Alias for **empty**. Dog skal man være opmærksom på at dette kun gælder den første og den sidste side. Mellemliggende

Sidehoved og -fod

sider kan sagtens få sidehoved eller -fod. Dette må man så rette undervejs med `\thispagestyle`.

indextitlepagestyle denne anvendes til forsiden af index, alias for **chapter**. Grunden til at man ikke bare anvender **chapter** direkte er så man f.eks. kan lave short cuts i indekset som vi har i PDF-udgaven af denne bog.

headings dette er standard pagestylen i *memoir*. Den har en tom sidefod, sidetallet skrives i sidehovedet ved den yderste kant af papiret. Desuden indeholder den kapitel information i sidehovedet på venstre sider, og section information på højre sider.

Memoir har desuden nogle ekstra predefinerede pagestyles på linie med **headings**, disse er nærmere beskrevet i [Wilson \(2010\)](#).

For god ordens skyld forklarer vi lige hvad en pagestyle er. Pagestylen består af de makroer som indsætter sidehoved og -fod når siden bliver samlet. Disse makroer kan så få forskellige oplysninger stillet til rådighed, f.eks. sidetallet samt dynamisk titelmateriale (f.eks. kapitel- og sektionsoverskrifter). Klassen *memoir* har så lavet et specielt interface til disse makroer, som gør dem nemme at ændre. For flere detaljer om pagestyles, kan man tage et kig på [Madsen \(2008a\)](#).

De dynamiske overskrifter er gemmes og er tilgængelige for sidestilen via `\leftmark` og `\rightmark`.¹¹ Fortolkningen af dem er at `\leftmark` indeholder informationen fra den *sidste* specifikation til `\leftmark` før sideskiftet, mens `\rightmark` indeholder den *første* specifikation til `\rightmark`. Hvis ingen af dem er angivet på nuværende side tages de *seneste* specifikationer fra siden før. Vi vil senere forklare hvordan man får specificeret hvad `\leftmark` hhv. `\rightmark` kommer til at indeholde.

Man bør ikke selv ændre på `\leftmark` eller `\rightmark`. Det har vi specielle makroer til.



Memoir-klassen tillader os nemt at lave en ny pagestyle f.eks. på basis af en anden, eller at lade en pagestyle være et andet navn for en given pagestyle. Vi vil først kort se på hvordan man laver sin egen pagestyle i *memoir*, hvorefter vi løser problemet vi blev givet. De centrale makroer er

```
\makepagestyle{<navn>}
\copypagestyle{<nyt navn>}{<original>}
\aliaspagestyle{<alias navn>}{<original>}
\makeevenhead{<style>}{<venstre>}{<center>}{<højre>}
\makeoddhead{<style>}{<venstre>}{<center>}{<højre>}
\makeevenfoot{<style>}{<venstre>}{<center>}{<højre>}
\makeoddfoot{<style>}{<venstre>}{<center>}{<højre>}
\makepsmarks{<style>}{<kode definitioner>}
\makeheadrule{<style>}{<bredde>}{<tykkelse>}
\makefootrule{<style>}{<bredde>}{<tykkelse>}{<skip>}
```

Alle makroerne er forklaret i [Wilson \(2010\)](#), kapitel 14, specielt 14.1–3.

¹¹ Det er muligt at få adgang til endnu flere typer marks, men det ligger udenfor denne bog.

12. Dokumentopsætning med memoir-klassen

Sidebemærkning 12.9. Bemærk at stile lavet med `\aliaspagestyle`, ikke kan ændres via `\makeoddhead` og venner. Ønsker man at ændre noget ved en stil og bevare den originale så bør man starte med at lave en kopi med `\copypagestyle`. Nærmere forklaring findes i [Madsen \(2008a\)](#).

Sidebemærkning 12.10. Bemærk desuden at i *memoir* skal en dokumentstil aktiveres før ændringerne kan ses, dvs. med `\pagestyle`. Dette gælder ikke for stile som kun anvendes lokalt, f.eks. **chapter**-stilen.

De to første makroer laver/overskriver en helt ny pagestyle, forskellen er at `\copypagestyle` kopierer alle settings hørende til *original*, hvorimod `\makepagestyle` sætter dem alle til at være tomme. Aliaserne bevirker bare at `\pagestyle{alias}` er det samme som at udføre `\pagestyle{original}`, men det gør det nemt at flytte om på *alias* senere. Man anvender også `\aliaspagestyle` til at ændre et givet alias, kan endda anvende den til at lade en almindelig sidestil pege på en givet anden stil.

Brugen af de fire makroer `\makeoddhead`, `\makeevenfoot` giver sig selv. Man skal huske at *even* henviser til venstre-sider, mens *odd* er højre-sider. Husk også at på venstre sider er *højre* ind mod ryggen af dokumentet, mens *højre* på højre sider er ud om yderkanten af papiret.

Selve indholdet man anvender til *venstre*, *center* og *højre* består sædvanligvis af

- fontspecifikation incl. størrelse
- `\thepage`
- `\leftmark` eller `\rightmark`

En ofte anvendt opsætning er baseret på følgende eksempel

Eksempel 12.3

```
\makepagestyle{minstil}
\makeoddhead{minstil}{\small\itshape\rightmark}}{\small\thepage}
\makeevenhead{minstil}{\small\thepage}          {\small\itshape\leftmark}
% behøver ikke noget i sidefod
... % mangler marks som forklares senere
\pagestyle{minstil} % aktivering
```

I stedet at direkte at lave sin egen stil, kan man naturligvis også pille ved **headings**-stilen.

Sidebemærkning 12.11. Hvorvidt noget skal skrives med rent store bogstaver i f.eks. sidehovedet, skal *ikke* specificeres via eksempelvis `\makeoddhead`, af forskellige årsager skal dette specificeres før data gives til `\leftmark` eller `\rightmark`. Vi forklarer hvordan i næste sektion.

Man kan sætte streg under sidehovedet eller over sidefoden. Til dette anvendes de åbenlyse makroer `\makeheadrule` og `\makefootrule`, argumenterne er nogenlunde nemme, bredden vil oftest sættes til `\textwidth` og til tykkelsen kan man anvende `\normalrulethickness`, så tykkelsen passer med mange andre konstruktioner i L^AT_EX. *skip*-delen ved linen til footeren bør sættes til `\footruleskip`.

Opsætning af marks

Opsætning af marks

Justeringen af hvad f.eks. `\chapter` stiller tilrådighed til headerne har altid været et mareridt for nye og erfarne brugere, det krævede speciel viden om L^AT_EXs interne makroer. Det er heldigvis blevet bedre med den nye version af *memoir*.

`\leftmark` og `\rightmark` tildeles data via makroer som er indbygget i f.eks. `\chapter`, `\section`, `\tableofcontents` eller *thebibliography*-environmentet, navngivet hhv. `\chaptermark`, `\sectionmark`, `\tocmark` og `\bibmark`. Det er så op til brugeren at definere hvad alle disse makroer skal gøre. I *memoir* anvender vi `\makepsmarks{<style>}{<code>}` til at associere `\. . .mark` definitioner med en bestemt `<style>`. I den nye version af *memoir* har vi tilføjet tre ekstra makroer til at generere indholdet af `\. . .mark`-makroerne for os. De tre kommandoer er

```
\createplainmark{<type>}{<marks>}{<tekst>}
\createmark{<sec>}{<marks>}{<vis i mainm.>}{<prefix>}{<postfix>}
\addtopsmarks{<style>}{<ekstra kode til start>}{<ekstra kode til slut>}
```

Den første makro er til de såkaldte *plain* marks, dette dækker over indholdsfortegnelsen (og venner), indekset samt litteraturlisten. `<type>` skal være en fra listen `toc`, `lot`, `lof`, `index`, `bib` (samt andre som ikke anvendes så ofte). `<marks>` angiver om det kun skal skrives til `\leftmark` (»left«), til `\rightmark` (»right«) eller begge (»both«). Man anbefales *altid* at anvende »both«. `<tekst>` er så den tekst som skal gives til headeren. Den mest almindelige konfiguration af plain marks er:

```
\createplainmark{toc} {both}{\contentsname}
\createplainmark{lof} {both}{\listfigurename}
\createplainmark{lot} {both}{\listtablename}
\createplainmark{bib} {both}{\bibname}
\createplainmark{index}{both}{\indexname}
```

Eksempel
12.4

De generelle regler for hvad der skal med når f.eks. `\chapter` skriver noget til headeren er noget mere komplekse, hvorfor makroen `\createmark` har flere argumenter.

Sidebemærkning 12.12. Husk iøvrigt at `\chapter` i *memoir* kan håndtere op til to alternative argumenter, således at man har

```
\chapter{test} % test skrives til tekst, header og toc
\chapter[til toc og sidehoved]{til tekst}
\chapter[til toc][til sidehoved]{til tekst}
```

Eksempel
12.5

meget anvendelig hvis man har meget lange titler som ikke egner sig til headeren.

En af anbefalingerne er at når man er indenfor *frontmatter* så skal der slet ikke skrives nogen kapitel- og seksionsnumre, ej heller i headeren, mens man i *mainmatter* kan *vælge* at tage nummeret med i headeren. Nogle ønsker ofte at man indenfor *mainmatter* kan tilføje ordet »Kapitel« foran kapitelnummeret. Dette kan vi nu lave meget nemt via:

```
\createmark{<sec>}{<marks>}{<vis i mainm.>}{<prefix>}{<postfix>}
```

\createmark

`<sec>` henviser til »chapter«, »section«, etc.¹² `<marks>` er »left«, »right« eller »both« som tidligere beskrevet. `<vis i mainm.>` skal være enten »shownumber« eller »nonumber«

¹² Man kan også anvende »part« som jo ligger over kapitel niveau, men dette anvendes meget sjældent.

12. Dokumentopsætning med memoir-klassen

og angiver om man ønsker at tage et eventuelt nummer med udenfor frontmatter. Har man helt slået nummereringen fra for denne *sec* kommer der naturligvis heller ikke noget nummer med i headeren. I fald man ønsker nummerering så skrives *prefix* før nummeret og *postfix* efter. *prefix* kan f.eks. anvendes til at tilføje ordet »Kapitel« og *postfix* vil oftest være »._«.

Alle marks til standardstilen i *memoir* (**headings**) kan til et to-sidet setup nu skrives (jeg vil nu altid anbefale at man anvender »both« ved »chapter«):

Eksempel 12.6

```
\makeatletter % pga. \@chapapp
\makepsmarks{headings}{
  \createmark{chapter}    {left} {shownumber}{\@chapapp\ }{. \ }
  \createmark{section}    {right}{shownumber}{ }{. \ }
  \createplainmark{toc}   {both} {\contentsname}
  \createplainmark{lof}   {both} {\listfigurename}
  \createplainmark{lot}   {both} {\listtablename}
  \createplainmark{bib}   {both} {\bibname}
  \createplainmark{index}{both} {\indexname}
}
\makeatother
```

Hvor \@chapapp er den magiske makro

$$\@chapapp = \begin{cases} \appendixname & \text{hvis vi er indenfor \appendix} \\ \chaptername & \text{ellers} \end{cases}$$

Som man nok kan se laver *memoir* som standard ikke marks ved \subsection eller \subsubsection, for nemt at kunne tilføje dette skal vi anvende den sidste marko, \addtopsmarks, lad og tilføj marks til **headings**:

Eksempel 12.7

```
\addtopsmarks{headings}{}{%
  \createmark{subsection} {right}{shownumber}{. \ }
  \createmark{subsubsection}{right}{shownumber}{. \ }
}
```

Håndtering af kapitaler i marks

I standardklasserne samt som standard i *memoir* vil al data skrevet til headerne via \...mark kommandoerne blive skrevet med store bogstaver. I standardklasserne er dette ikke nemt at fikse, men i *memoir* er det nu blevet nemt at ændre. Som oftest ønsker man bare at fjerne kapitaleringen, hvilket gøres via

```
\nouppercaseheads      \nouppercaseheads
```

hvorefter pagestylen lige skal aktiveres igen.



Stilen vi skulle lave, er anderledes end **headings**, så vi laver vores egen stil kaldet »bog«:

Justering vedr. overskrifter

```
\makeatletter % pga \@chapapp
\makepagestyle{bog} % laver en ny tom pagestyle
\makeevenhead {bog}{\small\sffamily\leftmark} {}
\makeoddhead {bog}{\small\sffamily\rightmark} {}
\makeevenfoot {bog}{\small\sffamily\thepage} {} {}
\makeoddfoot {bog}{\small\sffamily\thepage} {} {}
\makepsmarks {bog}{
  \createmark{chapter} {both} {shownumber} {\@chapapp} {. \ }
  \createmark{section} {right}{nonumber} {} {. \ }
  \createmark{subsection} {right}{nonumber} {} {. \ }
  \createmark{subsubsection}{right}{nonumber} {} {. \ }
  \createplainmark{toc} {both}{\contentsname}
  \createplainmark{lof} {both}{\listfigurename}
  \createplainmark{lot} {both}{\listtablename}
  \createplainmark{bib} {both}{\bibname}
  \createplainmark{index} {both}{\indexname}
}
\nouppercaseheads % ingen rene kapitler
\makeatother
```

Eksempel

12.8



Dernæst skal denne nye sidestil aktiveres:

```
\pagestyle{bog}
```

Eksempel

12.9

For at fikse stilen anvendt på specielle sider overskriver vi **plain** med den pagestyle vi lige har lavet og sletter sidehovederne (husk at vi skal bruge `\coppagestyle`):

```
\coppagestyle{plain}{bog}
\makeevenhead{plain}{}{} % slet venstre sidehoved
\makeoddhead{plain}{}{} % slet højre sidehoved
```

Eksempel

12.10



Hvilket er alt hvad man behøver gøre for at konfigurere sidehoved og -fod. Da **chapter**, **part** og **indextitlepagestyle** alle på en eller anden måde peger tilbage på **plain** er det nok at ændre netop denne stil.

12.3.4 Justering vedr. overskrifter

Dette behandles i [Wilson \(2010\)](#), kapitel 8 som handler om dokumentopdelinger. Memoir-klassen har specielle konstruktioner som kan styre hvordan første side i et kapitel skal se ud, i *memoir* kaldes dette for *chapterstyle*, kan sammenlignes med pagestyle. Selve *memoir*-manualen ([Wilson, 2010](#)) er faktisk skrevet således at hvert kapitel anvender en speciel chapterstyle. Dette kan være en smule forvirrende, så undertegnede har samlet dem samt visse andre i showcase-filen man finder i [Madsen \(2008b\)](#), henter man kildekoden til dokumentet og oversætter den én gang på sin maskine så bliver koden som skal anvende til de enkelte chapterstyles skrevet til separate filer (så er man fri for at skulle skrive af).

Skulle man have et design som man ønsker at få lavet til en *memoir* chapterstyle, så kan man prøve at kontakte undertegnede så kigger vi på det.

Memoir har også metoder til totalt at styre hvordan `\part`-siden skal se ud. Desuden stilles simple makroer til rådighed som kan styre udseendet af `\section` og lignende. Disse kan dog ikke styres i helt samme grad som f.eks. kapitler, men man kan f.eks. styre mellemrummet før og efter. Standard fontstørrelserne for part, kapitel og sektion, kan ses i [Wilson \(2010\)](#), tabel 8.2. Standardværdierne for mellemrummet før og efter, må man finde i kildekoden. Bemærk at mellemrummene bør/skal angives med såkaldte stræk og krymp værdier, således at har en side brug for at blive strukket/krympet så er der mulighed for at tage lidt omkring overskrifterne.

12. Dokumentopsætning med memoir-klassen



I vores specifikation vil vi bare gerne have teksten i fed sans-serif.¹³ Vi vælger at pille ved den `chapterstyle` som er aktiv som standard (den hedder `default`), dvs. vi ændrer nogle af de makroer som den aktiverede `chapterstyle` ændrede.

Eksempel 12.11



```
% til kapitler
\renewcommand\chapnamefont{\huge\bfseries\sffamily}
\renewcommand\chapnumfont{\chapnamefont}
\renewcommand\chaptitlenamefont{\huge\bfseries\sffamily\raggedright}
% til afsnitsoverskrifter
\setsecheadstyle{\Large\bfseries\sffamily\raggedright}
\setsubsecheadstyle{\large\bfseries\sffamily\raggedright}
\setsubsubsecheadstyle{\normalsize\bfseries\sffamily\raggedright}
\raggedbottomsectiontrue
```

Makroen `\raggedbottomsectiontrue` sørger for at når en afsnitsoverskrift forekommer tæt på et sideskift, hvor L^AT_EX normalt placerer overskriften øverst på næste side, da vil siden før sideskiftet ikke blive strakt nævneværdigt.

12.3.5 Afsnitsnummerering og justering af indholdsfortegnelsen

I standardklasserne (*book* eller *article*) kan man justere på hvilke afsnitsoverskrifter skal nummereres og hvor langt ned i hierarkiet indholdsfortegnelsen skal gå, ved at justere tællerne `secnumdepth` og `tocdepth`. Men med denne metode skal man selv huske at f.eks. `\section` svarer til tallet »1«.

I *memoir* har vi et nemmere interface til at justere på disse tællere:

```
\setsecnumdepth{<afsnitsnavn>}
\maxsecnumdepth{<afsnitsnavn>}
\settocdepth{<afsnitsnavn>}
```

Hvor `<afsnitsnavn>` er navnet (uden »\«) på den makro man ønsker at lade dybden gå til og med. F.eks. vil

Eksempel 12.12



```
\setsecnumdepth{subsection} % til og med
\maxsecnumdepth{subsection} % den maksimale dybde
```

sørge for at f.eks. `\subsubsection` ikke nummereres (så man ikke behøver at anvende den stjernede udgave for at undgå nummeret). `\maxsecnumdepth` placeres normalt bare i preambelen, og anvendes i `\mainmatter` for at justere dybden efter `\mainmatter` (indenfor `\frontmatter` er der jo ingen nummerering).

Bemærk at `\settocdepth` ikke behøver være den samme dybde som `\setsecnumdepth`, man kan sagtens have ToC dybde som er dybere eller lavere end selve nummereringen af afsnit.

Anvender man `\setsecnumdepth` i selve teksten så vil den også sætte `\maxsecnumdepth`.



¹³ Vi går i denne gennemgang ud fra at man ikke anvender `\part`, ellers skulle man selvfølgelig også konfigurere den.

Forside med memoir

Justeringen vi blev bedt om at lave bliver så til

```
\setsecnumdepth{subsection} % til og med
\maxsecnumdepth{subsection} % den maksimale dybde
\settocdepth{section} % til og med
```

Eksempel
12.13



I *memoir* kommer bibliografi, indholdsfortegnelse, indeks etc. automatisk med i indholdsfortegnelsen. Skulle man ikke ønske at indholdsfortegnelsen selv optræder som punkt i ToCen skal man blot anvende »\tableofcontents*«, bemærk »*«.

\tableofcontents*

For at få skiftet overskriften på indholdsfortegnelsen justerer vi ved `babel`, se afsnit 9.4 på side 229.

```
\addto\captionsdanish{\renewcommand\contentsname{Indholdsfortegnelse}}
```

Eksempel
12.14



Se iøvrigt også afsnittet »*Få ordet Kapitel eller Appendix med i indholdsfortegnelsen*« på side 328 for et smart lille *memoir* trick.



I [Wilson \(2010\)](#) kan man også læse hvordan man kan justere forskellige ting i indholdsfortegnelsen, bl.a. fonte, prikker, luft etc. Det kan man selv læse om, vær meget opmærksom på at læse teksten således at man finder ud af hvad »X« betyder. Noget man nemt kan komme ud for med lange dokumenter er at afsnitsnummereringen eller sidetallet kommer til at fylde mere end der er afsat plads til i designet af indholdsfortegnelsen.

Til at justere bredden af området hvori sidetallet placeres anvendes

```
\setpnumwidth{<længde>}
\setrmarg{<længde>}
```

Den første sætter bredden af boksen hvori sidetallet sidder, den anden (`\setrmarg`) styrer hvor langt mod højre teksten i indholdsfortegnelsen må komme, det er så at sige den højre margin for titlerne i ToCen. Værdien angivet i `\setrmarg` skal være større end værdien i `\setpnumwidth`. I denne bog anvender vi noget lignende 2.55 em hhv. 3.55 em.

Har man problemer med afsnitsnumre, f.eks. »10.10« så er det længden

```
\cftXnumwidth
```

vi skal have gang i, hvor »X« svarer til typen af indgangen man retter, f.eks. »section«. Man kan så lægge lidt til denne via `\addtolength`.

12.3.6 Forside med memoir

Det at designe en forside er lidt en kunst i sig selv og det er ikke noget vi kan nå at dække til fulde her, vi vil i stedet præsentere nogle af de redskaber man kunne få brug for.

Sidebemærkning 12.13. Peter Wilson, som har skrevet *memoir*-klassen, har skrevet en oversigt over nogle forskellige forside-designs, se [Wilson \(2007b\)](#) for yderligere information. Dokumentet starter med at fremvise de forskellige designs, hvorefter koden til hvert design kan forefindes i slutningen af dokumentet.

12. Dokumentopsætning med memoir-klassen

Louises forside

Her på Matematik er der efterhånden en del studerende som har fået den samme forsideskabelon. Den blev første gang anvendt da jeg lavede den til Louise Pold Thomsens bachelorprojekt – derfor tituleret som Louises forside.¹⁴ Da denne forside har fået positiv respons tager vi den med her. [Eksempel 12.15](#) angiver forsiden som et selvstændigt dokument. Bemærk at denne forside viser eksempel på hvordan den engelske titel kunne vises frem. Cirklen skal naturligvis udskiftes med et logo eller et andet billede.

Eksempel 12.15

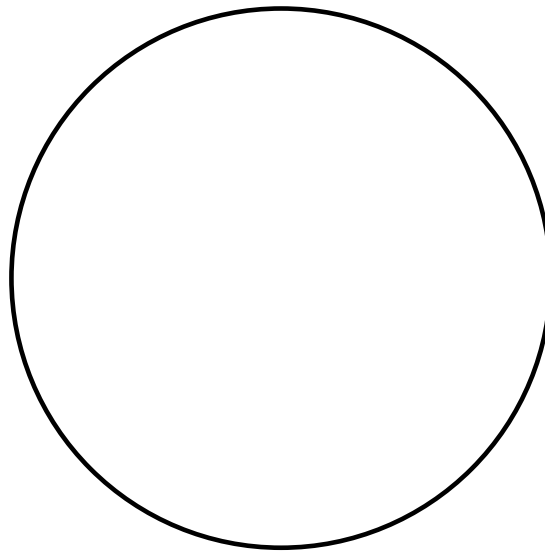


```
\documentclass[a4paper,12pt]{memoir}
\setlrmarginsandblock{3cm}{*}{1.25}
\checkandfixthelayout
\usepackage[latin1]{inputenc}
\usepackage[danish]{babel}
\usepackage[T1]{fontenc}
\usepackage{graphicx}
\usepackage{mathpazo}% palatino + matematik
\usepackage{soul} % lege lege
\define\an{}{0.2em}{.9em plus.6em}{1em plus.1em minus.1em}
\newcommand\stext[1]{\an{\scshape#1}}
\begin{document}
\begin{titlingpage}
  \thispagestyle{empty}
  \centering
  { \setlength{\baselineskip}{24pt}
    {\Huge \stext{Overlejringer} \par
      \textit{\&} \par
      \stext{Hadamards Sætning}
    } \par
    \stext{(Covering Spaces and Hadamards Theorem)}
    \par\vspace*{4\onelineskip}
    \par
    \includegraphics[width=8cm]{fig.2}
    \par\vspace*{4\onelineskip}
    \stext{Bachelorprojekt i matematik} \par
    \large\stext{Louise Pold Thomsen --- 12345678} \par
  }
  \vfill
  \vspace*{2\onelineskip}
  \stext{Vejleder: Ib Madsen} \hfill
  \stext{30. marts 2004}
  \par\vspace*{2\onelineskip}
  \small
  \stext{Institut for Matematiske Fag} \par
  \stext{Aarhus Universitet}
  \enlargethispage{2\onelineskip}
\end{titlingpage}
\end{document}
```

Makroen `\par` som er anvendt i koden er det samme som en blank linie, den afslutter et afsnit. Her anvendt for at få koden til at fylde mindre. Pakken `soul` giver den specielle effekt hvor teksten har større afstand mellem bogstaverne end normalt, se [Franz \(2003\)](#). `\onelineskip` er en speciel *memoir*-længde. Resultatet kan ses på [figur 12.2 på næste side](#). Bemærk at rammen omkring billedet *ikke* er repræsentativ for tekstens forhold til papirets størrelse.

¹⁴ Tak til Louise for tilladelse til at koden kunne inkluderes her.

OVERLEJRINGER
&
HADAMARDS SÆTNING
(COVERING SPACES AND HADAMARDS THEOREM)



BACHELORPROJEKT I MATEMATIK
LOUISE POLD THOMSEN — 12345678

VEJLEDER: IB MADSEN

30. MARTS 2004

INSTITUT FOR MATEMATISKE FAG
AARHUS UNIVERSITET

Figur 12.2: Forsiden til Louises bachelorprojekt.

12. Dokumentopsætning med memoir-klassen

Strits forside

Når tiden tillader det kan studerende ved Matematik komme forbi mit kontor og få hjælp til at få specialer, bachelorprojektor eller ph.d.-afhandlinger justeret. Så i samarbejde med Marie 'strit' Jensen (nu Janussen) har vi udarbejdet nedenstående forsidedesign til hendes speciale.¹⁵ Designet er inspireret af de designs man kan finde i [Wilson \(2007b\)](#). I stedet for cirklen er det naturligvis meningen at man skal anvende universitets segl, men det må vi bare ikke her.

De specielle ting som er anvendt her er Utopia fonten (gennem fourier-pakken) samt specialkonfigureret letterspacing via soul, se afsnit [9.6.6 på side 238](#). Vi har anvendt `\MakeUppercase` for ikke at skulle rette for meget i teksten mens vi eksperimenterede med designet. Desuden er størrelserne i titlen valgt således at de vigtige ting i specialet er fremhævet.

Eksempel 12.16



```
\documentclass[a4paper,11pt,oneside]{memoir}
\setlrmarginsandblock{3cm}{3cm}{*} \setulmarginsandblock{3cm}{4cm}{*}
\checkandfixthelayout
\usepackage[danish]{babel} \usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc} \usepackage[widexspace]{fourier}
\DeclareRobustCommand{\SetFourierSpace}{%
\fontdimen2\font=1.13\fontdimen2\font}
\usepackage{graphicx} \usepackage{fix-cm}
\usepackage{soul}
\sodef\ann{}{0.13em}{0em}{0em} \sodef\ann{}{0.13em}{0.5em}{0em}
\begin{document}
\begin{titlingpage} \vspace*{5mm} \centering
\huge \textsc{\ann{Marie Jensen}} \par
\small \ann{1234\,5678} \vspace{3mm} \par
\fontsize{1.2cm}{1cm}\selectfont \MakeUppercase{\ann{Coxeterelementers}}
\vspace{3mm} \par
\Huge \MakeUppercase{\ann{0rden}} \vspace{3mm} \par
\LARGE \MakeUppercase{\ann{og}} \vspace{5mm} \par
\fontsize{1.2cm}{1cm}\selectfont \MakeUppercase{\ann{Egenv\ae rdier}}
\vspace{5mm} \par
\large \sodef\ann{}{0.13em}{0.5em}{0em}
\textsc{\ann{Vejleder: Henning Haahr Andersen}}
\enlargethispage{15mm} \vspace{20mm} \par
\includegraphics[width=8cm]{fig.2} \par
\vspace{5mm} \vfill \par
\normalsize \scshape \sodef\ann{}{0.13em}{0.6em}{0em}
{\ann{Specialeafhandling i Matematik}} \par
\vspace{2mm}
\ann{Institut for Matematiske Fag} \par
\ann{Aarhus Universitet} \par
\ann{2007}
\end{titlingpage}
\end{document}
```

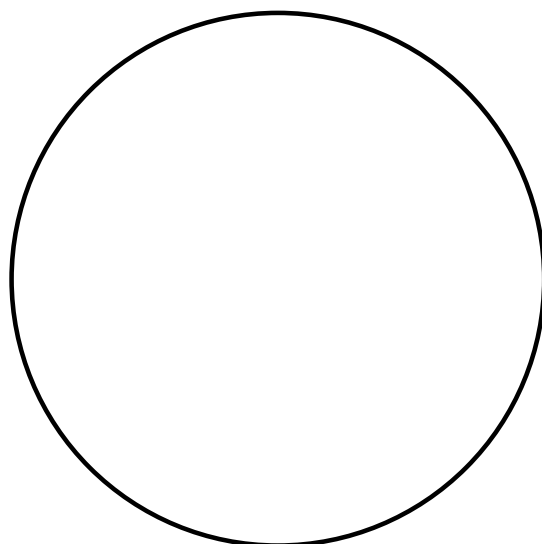
Resultatet kan ses på figur [12.3 på næste side](#). Bemærk at rammen omkring billedet *ikke* antyder papirets størrelse.

¹⁵ Tak til strit for tilladelse til at inkludere designet som et eksempel i bogen.

MARIE JENSEN
1234 5678

COXETERELEMENTERS
ORDEN
OG
EGENVÆRDIER

VEJLEDER: HENNING HAAHR ANDERSEN



SPECIALEAFHANDLING I MATEMATIK
INSTITUT FOR MATEMATISKE FAG
AARHUS UNIVERSITET
2007

Figur 12.3: Forsiden til Strits speciale.

12. Dokumentopsætning med memoir-klassen

Centrering af forside i forhold til papiret

Hvis forsiden skal anvendes som forside på et omslag kan det oftest være en god ide at sørge for at centrere teksten i forhold til papiret, i stedet for i forhold til tekstmarginerne. Dette kan man fikse meget nemt med *memoir*, se [Eksempel 12.17](#). Her anvender vi environmentet *adjustwidth* og makroen `\calccentering` til at få centreret forsiden i forhold til selve papiret. Hvis man ydermere ønsker at tekstbredden på forsiden skal være bredere end den gængse tekstbredde, så omkranser man bare forsiden af endnu en `\adjustwidth`.

Eksempel 12.17



```
\begin{titlingpage}
\calccentering{\unitlength} % forudsat \unitlength ikke bruges til andet
\begin{adjustwidth*}{\unitlength}{-\unitlength}
  \begin{adjustwidth}{-1cm}{-1cm}
    forside materiale
  \end{adjustwidth}
\end{adjustwidth*}
% efter forsiden
...
\end{titlingpage}
```

I [Eksempel 12.17](#) har vi lavet forsiden centreret i forhold til papiret og tekstbredden på selve forsiden er 2 cm bredere end i resten af dokumentet. Se [Wilson \(2010\)](#), afsnit 9.5 for yderligere information.

En kolofon bag på forsiden

Er simpelt:

Eksempel 12.18



```
\begin{titlingpage}
% forside kode
\newpage
\thispagestyle{empty} % fjerne evt. sidehoved og -fod
\small
%resten af teksten indenfor dette env skal være \small
\strut\vfill % pres alt ned i bunden af siden
\begin{flushleft}
  \copyright\ Fornavn Efternavn 2008 \par
  Dette værk er trykt med Computer Modern 11pt\,/,\,13.6pt \par
  Layout og typografi af forfatteren ved hjælp af \LaTeX \par
  Alle figurer er lavet med \MP
\end{flushleft}
\end{titlingpage}
```

Makroen `\MP` kommer fra pakken `mflogo` og giver »METAPOST«. Vi anvender `\par` for at spare plads, og *flushleft* for at skubbe alt mod venste, samt fjerne indentering og linieombrydning.

12.3.7 Frontmatter

`\frontmatter` For at skifte over til den normale frontmatter indstilling, skal man bare placere

Eksempel 12.19

```
\frontmatter
```

på det sted i sin tekst hvor man ønsker at frontmatter skal starte. Dette er oftest lige efter forsiden eller efter forside relateret materiale. `\frontmatter` gør så følgende (under *memoir*, se [Wilson \(2010\)](#), kapitel 8):

- (a) Skifter til en ny højre (ulige) side.

En dedikationsside

- (b) Starter sidenummereringen ved 1 og skriver sidetallet med romertal.
- (c) Slår alle afsnitsnummeringer fra. Dvs. at nu vil f.eks. `\section{<tekst>}` ikke længere have noget nummer, men `<tekst>` vil alligevel blive skrevet til indholdsfortegnelsen.
- (d) Figurer og tabeller nummereres (via `\caption`) fortløbende, som 1,2,3,...

En dedikationsside

16

Er ikke en del af forsiden, men meget nem med de teknikker som er beskrevet her eller i appendix.

```
% skal starte på en højre side
\cleardoublepage
\thispagestyle{empty} % ingen sidehoved eller -fod
\vspace*{\stretch{3}} % speciel faktoriseret gummilængde
\begin{center}
  Til min kat
\end{center}
\vspace*{\stretch{17}}
% ting efter siden skal starte på en højre side
\cleardoublepage
```

Eksempel
12.20



Med *memoir* kan man indeni *center* evt. anvende environmentet

```
\begin{adjustwidth}{<indryk venstre>}{<indryk højre>}
  <tekst>
\end{adjustwidth}
```

adjustwidth

hvis det er en længere tekst. Med dette environment gør man tekstbredden lokalt smallere, hvilket ser pænt ud på en dedikationsside.

Dedikationssiden i denne bog er lavet med noget som ligner

```
\vspace*{\stretch{3}}
\begin{adjustwidth}{9em}{9em}
  \centering \large \itshape

  til de danske \LaTeX-brugere med håbet om at
  dette vil hjælpe på vejen mod pænere dokumenter

\end{adjustwidth}
\vspace*{\stretch{11}}
```

Eksempel
12.21



Resumé på flere sprog

Vi har allerede nævnt at man kan indlæse `babel` med flere sprog. Så for at lave et resumé på engelsk og et på dansk på samme side, er det bare at anvende: *babel*

¹⁶ Fixme Note: skal nok placeres et andet sted

12. Dokumentopsætning med memoir-klassen

Eksempel 12.22

```
\begin{otherlanguage}{english}
\begin{abstract}
  blah blah blah (in english)
\end{abstract}
\end{otherlanguage}
\begin{abstract}
  bla bla (på dansk)
\end{abstract}
```

`\selectlanguage` `otherlanguage` Konstruktionen kan også laves med to `\selectlanguage{<prog>}`. Brugen af *otherlanguage* giver en fin overskuelig opdeling af kildekoden.

12.3.8 Mainmatter

`\mainmatter` Dette aktiveres på samme måde som ved `\frontmatter` via `\mainmatter` som så sørger for at:

- (a) Skifter til en ny højre side.
- (b) Genstarte sidenummereringen fra 1, nu med almindelige tal.
- (c) Sørger for at figur- og tabeltællerne nu igen er underlagt chapter-tælleren.

Se også [Eksempel 12.23](#) og [Eksempel 12.24](#) samt den tilhørende tekst.



Nogle ender deres opgave med et kapitel som tjener som konklusion. Til tider ønsker man ikke at dette kapitel bliver nummereret, men man vil naturligvis gerne have den med i indholdsfortegnelsen. Dette kan man gøre på to måder:

Eksempel 12.23

```
\setsecnumdepth{part} % en over chapter
\chapter{Konklusion}
```

eller

Eksempel 12.24

```
\chapter*{Konklusion} % ingen indgang i ToC
\addcontentsline{toc}{chapter}{Konklusion}
```

Så må man selve vælge. Anvender man [Eksempel 12.23](#), så slår man helt afsnitsnummeringen fra. Men så kan man jo bare slå den til igen senere, via

Eksempel 12.25

```
\setsecnumdepth{subsection}
```

12.3.9 Backmatter

Memoir har godt nok en `\backmatter`-makro, men vi anvender den ikke, idet, den slår afsnitsnummereringen fra, hvilket, for de fleste, ikke er det de er ude efter. Har man bilag eller appendices så kan man sige at backmatter starter ved `\appendix`. Bemærk iøvrigt at *memoir* inkluderer en `\appendixpage`-makro som vil lave en speciel side som illustrerer at nu starter appendices/bilag. Den vil desuden lave en indgang i indholdsfortegnelsen. Hvis man er utilfreds med de navne *memoir* anvender til henholdsvis den part-lignende side samt til indgangen i TOC, så skal man ændre makroerne `\appendixpage` og `\appendixtocname`. Man kan fint gøre det på samme måde som vi tidligere gjorde med `\contentsname`.

12.3.10 Eksempelfil for specialeopsætning med memoir

Store skabeloner hører ikke til på dette sted i manualen, dette skal i stedet tilføjes til kapitel F på side 419

12.4 Omjustering af layoutet i forbindelse med andre typer større opgaver

12.4.1 Et par ideer vedr. ph.d.-afhandlinger

En ph.d.-afhandling skal behandles som et speciale, måske endnu mere som en bog (afhængigt af længden af dokumentet). I hvert fald hos naturvidenskab, er ph.d.-afhandlingen enten udformet som et speciale på teorider eller den er opbygget af en opsamling samt en inklusion af en serie artikler man har skrevet (og forhåbentlig fået udgivet).

Vi vil i resten af dette afsnit kigge nærmere på hvordan man håndterer ph.d.-afhandlinger bestående af artikler. Vi har to umiddelbare muligheder: Enten inkluderer vi det hele i det samme dokument (sammenfletter preambles etc.), eller også forbereder man artiklerne hver for sig og inkluderer dem så på passende vis (som PDF filer).

Hver fremgangsmåde har sine fordele og ulemper. Har man det hele i samme dokument, så kan man f.eks. nemt få en fin indholdsfortegnelsen for det hele. Men det er bøvlet evt. at skulle skriver artiklerne om fordi de var skrevet med andre klasser. Desuden kan det være lidt besværligt at få litteraturlisten til at fungere.

I den anden lejr har man ikke brug for at tilpasse artiklerne ret meget, men det kan være besværligt at få konsistente sidetal, eller få overskrifterne automatisk med i indholdsfortegnelsen.

Afhandling: Artikler samlet i samme dokument

I teorien er dette den nemmeste løsning, man tager kildekoden til hver artikel og inkluderer dette som hver sit kapitel.¹⁷ Men hver artikel har jo sin egen litteraturliste, hvilket de naturligvis også skal have i afhandlingen.

Dette løses ret nemt via pakken `chapterbib` under forudsætning af at man anvender `BIBTEX`. Det er et krav at man inkluderer sine artikler via `\include{<filnavn>}` således at hver artikel får sin egen `.aux`-fil. Man skal så placere

chapterbib

```
\bibliographystyle{<bibstil>}
\bibliography{<bibfil>}
```

i hver af artikelfilerne. Der er nu bibliografiske data i artiklernes `.aux`-filer, så man skal huske at køre `bibtex` på hver af dem.

Det næste problem bliver at den genererede litteraturliste, normalt får samme status som et kapitel. Hvilket er skidt når artiklen selv er et kapitel. Pakken `chapterbib` kan selv fikse det via optionen `[sectionbib]`, men dette vil nummerere litteraturlisten, hvilket ikke falder i alles smag. Anvender man `memoir` kan man benyttes sig af følgende kode:

```
\newcommand{\MemSecBib}{%
  \section*{\bibname}%
  \bibmark
  \ifnobibintoc\else
    \phantomsection
    \addcontentsline{toc}{section}{\bibname}%
  \fi
  \prebibhook}
\renewcommand\bibsection{\MemSecBib}
```

Kodetip
12.1



¹⁷ De fleste artikler er jo skrevet med `article` klassen, så det skulle være lige til.

12. Dokumentopsætning med memoir-klassen

Sidebemærkning 12.14. Hvis man desuden anvender pakken natbib (det gør de fleste), så skal omdefinitionen af `\bibsection` placeres *efter* »`\begin{document}`«, ellers sker der ikke noget.

Afhandling: Artikler inkluderet som PDF

Her antager vi at man arbejder med et moderdokument, som kan oversættes med `pdflatex`¹⁸ samt at man har kildekoden til hver artikel tilgængelig. Ideen er nu at bruge pakken `pdfpages` til at inkludere de enkelte artikler. Vi kan så, i moderdokumentet, udstyre hver artikel med en form for forside, som nemt kan inkluderes i indholdsfortegnelsen.

Det vil være en fordel hvis man laver en smule om på artiklerne:

- (i) Fjern sidehoved og -fod. Dem sætter vi ind fra moderdokumentet og får dermed en bedre kontinuitet. Nemmeste måde at slippe af med sidehoved og -fod er

Eksempel 12.26

```
\pagestyle{empty}
```

- (ii) Den anden ting er at sørge for at artiklen anvender den samme fontstørrelse (og evt. font) som moderdokumentet, dette er nemt nok, ellers se andetsteds i denne bog.
- (iii) Sørg for at artiklen er sat op til at være i *twoside*-mode, således at marginerne skifter korrekt.
- (iv) Sørg for at artiklen anvender de samme marginer som moderdokumentet. Her kan det hjælpe en del at bruge pakken `geometry` i artiklen.

Tip 12.15. Når man har anvendt `\checkandfixthelayout` i *memoir*, så skriver den layoutet til `.log`-filen.

Det nemmeste er naturligvis at kunne inkludere selve koden til artiklen som et kapitel i afhandlingen. Til tider er dette enten ikke muligt, eller man ønsker bare at inkludere artiklen direkte uden at den bliver en direkte del af det nuværende dokument. Har man læst afsnit 6.1.3 på side 141 så kender man også den nemmeste løsning på problemet: `pdfpages` og dens makro `\includepdf`. Så det simpleste er at konvertere artiklerne til PDF og så inkludere disse i sit dokument. Det kan så være en god ide at man i sit hoveddokument lige sørger for at skrive en forside til hver artikel, disse kan så desuden blive anvendt til at indsætte artikelens titel i indholdsfortegnelsen, f.eks. via `\addcontentsline`.

19

Inklusion af PDF med en master som ikke tåler pdflatex

Skulle man være så uheldig at man ikke kan kompilere sit hoveddokument med `pdflatex` som jo er påkrævet af `pdfpages` så skal man være en smule mere kreativ. Men da det er sjældent i dag at finde dokumenter som ikke kan oversættes med `pdflatex`

¹⁸ For moderdokumenter som ikke kan, se næste afsnit.

¹⁹ FixMe Dødelige: tilføj mere her, bla. med forsider, samt `addtotoc`, samt den ikke nummererede opgave

Pagestyle til artikler i afhandling

så er det ikke noget vi ønsker at gå i detaljer med. Har man brug for hjælp til dette er man velkommen til at sende en mail.

Pagestyle til artikler i afhandling

Vi kunne godt tænke os at vi som sidehoved til hver artikel har forfatterens navne på venstresider og titlen på højre. Så kan læseren se hvilken artikel han/hun er i netop nu. Da vi er dovne gider vi ikke lave en ny pagestyle til hver artikel, eller lave en som vi skal lave om på hele tiden. Vi bruger i stedet følgende. Som illustration vælger vi at centrere navn, titel og sidetallet, tilpas det selv efter egne ønsker.

```
\makepagestyle{myarticle}
\makeevenhead{myarticle}{}{\leftmark}{}
\makeoddhead{myarticle}{}{\rightmark}{}
\makeevenfoot{myarticle}{}{\thepage}{}
\makeoddfoot{myarticle}{}{\thepage}{}
% dette er den smarte del, vi vil ikke
% have a sections etc. forstyrrer headeren
\makepsmarks{myarticle}{
  \clearmark{chapter}
  \clearmark{section}
  \clearmark{subsection}
  \clearmark{subsubsection}
  \clearplainmark{toc}
  \clearplainmark{lot}
  \clearplainmark{lof}
  %...
}
```

Kodetip

12.2



Når man så starter med sine artikler smider man først:

```
\pagestyle{myarticle}
\aliaspagestyle{chapter}{myarticle}
```

Eksempel

12.27

(alias delen behøver man ikke hvis man inkluderer PDF-filer). Efterfulgt af følgende før hver artikel:

```
\chapter{Blah blah}
\markboth{Navn}{Artikeltitel}
```

Eksempel

12.28

Helt konkret kan ændringen af pagestylen for kapitler egentlig være lige meget det kommer meget an på hvad man ønsker at bruge starten af kapitlet til. Det kunne jo være at man gerne ville skrive en eller anden introduktion inden man kom med selve indholdet af artiklen.

Tip 12.16. Hvis man gerne vil have artiklen til at starte på næste højreside, så kan man med fordel smide en `\cleartorecto` før artiklen.

12. Dokumentopsætning med memoir-klassen

12.4.2 Tanker omkring opsætning af bachelorprojekter

Specialer og ph.d.-afhandlinger forventes at fylde ca. 40+ sider hvormed man sagtens kan betragte dem som værende en bog. Ved bachelorprojekter for holder det sig lidt anderledes. De fleste projekter er sikkert omkring 20–30 sider, hvilket ikke ser ud af meget hvis man trykker det dobbeltsidet. Man vil derfor fint kunne komme med følgende anbefalinger vedrørende opsætningen af et bachelorprojekt med *memoir*.

- oneside* (i) Anvend *oneside* klasseoption og tryk projektet enkeltsidet. Så ser det ud af lidt mere.
- (ii) Husk at når man anvender enkeltsidet layout så vil kun headere og footere for *oddpge* blive anvendt. De tilsvarende for »*even*« sider anvendes *kun* ved *twoside*. Dette skal man altså overveje ved sin opsætningen af headere og footere. Bemærk dog at både `\leftmark` og `\rightmark` begge er til rådighed. Den generelle *pagestyle* opsætning vi tidligere har omtalt, hvor vi udvidede **headings** kan sagtens anvendes her, også selv om teksten faktisk siger at det er til et tosidet layout.
- headings* (iii) Undlad at skrive navn og titel i sidehovedet af rapporten. Med mindre man direkte bliver bedt om det er den slags redundant information.
- (iv) Til en bachelorrapport er der mange som synes det er lidt overkill at kalde kapitlerne for »Kapitel«. Man ønsker ofte noget mere afdæmpet til første side i hvert kapitel. Dette man man med *chapterstyle* »*section* «
- section*

Eksempel 12.29

```
\chapterstyle{section}
```

Så bliver overskriften ved `\chapter` bygget op via »`\langle kapitelnummer \rangle \langle titel \rangle`«.

Men husk at dette betyder stadigvæk at man opdeler sin tekst med `\chapter` som det øverste niveau.

- (v) Ønsker man fortløbende figur og tabel nummerering, så skal man bare anvende

Eksempel 12.30

```
\counterwithout{figure}{chapter}  
\counterwithout{table}{chapter}
```

i preamble.

Grunden til at vi ikke umiddelbart anbefaler at man bare skriver sit bachelorprojekt i *article*-klassen er naturligvis at *memoir*-klassen gør det meget nemmere efterfølgende at justere layoutet, sørger for at få de rigtige ting med i indholdsfortegnelsen, nemt at justere hvad der skal med i sidehoved og -fod. Det er simpelthen en god investering at lære om *memoir* så tidligt så muligt.

12.4.3 Større opgaver i gymnasiet

21

motivation: iflg. studieportalen er der faktisk en del brugere af L^AT_EX i gymnasiet. Umiddelbart ville en opsætning sammenlignelig med bachelor være brugbar plus dobbelt linieafstand. Man bør nok tjekke om der er nogle regler for dette.

Vi kan se at der er stadig flere i gymnasiet som anvender L^AT_EX, det er jo rart at se. Så vidt vides anvender man stadigvæk dobbelt linieafstand til opgaverne i gymnasiet. Så her skal man bare tage opsætningen til bachelorprojekter og så tilføje én af følgende i preamble:

²¹ Fixme Dødelige: skal skrives

Andre tricks man kan lave med memoir-klassen

```
\OnehalfSpacing  
\DoubleSpacing  
\SingleSpacing
```

Man anvender `\OnehalfSpacing` til at få det som dobbelt linieafstand burde svare til, det er det man normalt bruger. Har man en infantil lærer kan man anvende *ægte* dobbelt linieafstand (`\DoubleSpacing`), men det er grimt! Den sidste makro kan anvendes til at få den normale linieafstand tilbage igen.

Sidebemærkning 12.17. Det er som tidligere nævnt en meget dårlig ide at pille med `\linespread` for at opnå dobbelt linieafstand, denne piller ved noget som anvendes rigtigt mange steder hvorved man får alt for stor afstand på steder hvor man bestemt ikke ville have det. `\OnehalfSpacing` gør et noget bedre job.

Husk at L^AT_EXs gode evner til at behandle tekst, gør L^AT_EX ideel til brug ved dansk stil (hvis det da stadigvæk hedder det i dag). Her anvender man dog normalt ikke afsnitsnummerering, så den slår man bare fra via

```
\setsecnumdepth{part}  
\maxsecnumdepth{part}  
\settocdepth{subsection}
```

Eksempel
12.31

i preamble. Så har man simple overskrifter som fint kan komme med i indholdsfortegnelsen. Fordelen ved at anvende L^AT_EX til en dansk stil er naturligvis at man så bruger mere tid på at koncentrere sig om teksten i stedet for at sidde og pille ved layoutet.

12.5 Andre tricks man kan lave med memoir-klassen

Memoir-klassen har diverse små features som man kun kender hvis man har læst [Wilson \(2010\)](#) og [Wilson \(2007a\)](#). Her vil vi lige nævne nogle stykker som måske kunne gøre nytte for nogen.

Automatisk fjernelse af sidetal

Vi har allerede nævnt at har man et dokument som kun fylder en side og man har anvendt `\maketitle`, så skal man selv sørge for at anvende

```
\maketitle  
\thispagestyle{empty}
```

Eksempel
12.32

for at fjerne sidetallet.

Har man anvendt *memoir*-klassen, så kan følgende trick hjælpe. Man skal bare sørge for at placere følgende i sin preamble (kræver mindst to oversættelser):

```
\AtEndDocument{  
  \ifnum\value{lastsheet}=1  
  \thispagestyle{empty}  
  \fi  
}
```

Kodetip
12.3



Tælleren `lastsheet` anvendes i *memoir* til at kunne give det samlede antal ark i dokumentet.

12. Dokumentopsætning med memoir-klassen

Sidetæl af formen side / total

Skulle man have brug for at kunne henvise til den sidste side så kan man henvise til tælleren `lastpage`, f.eks. med `\value{lastpage}`. Bemærk at dette tal henviser dokumentets sidste værdi af sidetælleren. Dette er ikke altid det samme som antallet af ark i dokumentet, denne kan man få via `lastsheet` tælleren.

Så for at løse ovennævnte problem skal man bare tilføje følgende til sin pagestyle

Eksempel 12.33 `\theepage / \value{lastpage}`

Man skal dog være opmærksom på dette ikke altid er videre pænt, f.eks. er det grimt i et dokument hvor man også har frontmatter, men så kan man jo lave ændringen efter `\mainmatter`.

Få ordet Kapitel eller Appendix med i indholdsfortegnelsen

Det er ret nemt, se [Wilson \(2007a\)](#), afsnit 4.2.

Eksempel 12.34 `\renewcommand\cftchaptername{\chaptername~}`
`\renewcommand\cftappendixname{\appendixname~}`



Tilsvarende er der `\cfttablename` og `\cftfigurename` til brug i figur/tabelliste.

Fjerne afstand i figur/tabelliste

Hvis man ikke lige har opdaget det så vil man i f.eks. figurlisten kunne se når et nyt kapitel starter. Dette ses ved at der bliver indsat lidt luft i listen. Dette kan man naturligvis også rette ved. Denne luft indsættes via makroen `\insertchapterspace`, som i memoir er defineret ved

`\insertchapterspace`

Eksempel 12.35 `\newcommand{\insertchapterspace}{%`
`\addtocontents{lof}{\protect\addvspace{10pt}}%`
`\addtocontents{lot}{\protect\addvspace{10pt}}%`
`}`

Så kan man jo lave om på det som man vil. Vi har i denne bog omdefineret makroen til slet ikke at gøre noget, da det er meget forskelligt hvor mange figurer de enkelte kapitler indeholder.

Appendix OG bilag

Flere studerende har efterspurgt hvordan man kan få både appendix *samt* bilag i samme dokument. Argumentet er at materialet i appendix er noget man kan finde andre steder, mens bilag f.eks. kan være programkode, eller data.

Nuvel, dette kan man naturligvis også dressere *memoir* til at gøre. Vi vil antage at man desuden ønsker at få tilføjet *Kapitel*, *Appendix* og *Bilag* foran overskrifterne i indholdsfortegnelsen, samt at appendix nummereres med store bogstaver, mens bilag nummereres med store romertal. Konstruktionen går nogenlunde således:

Appendix OG bilag

```
% i preamble
\usepackage[danish]{babel}
\renewcommand\cftchaptername {\chaptername~}
\renewcommand\cftappendixname {\appendixname~}
\renewcommand\appendixpagenamename {Appendiks}
\renewcommand\appendixtocname {Appendiks}
\addto\captionsdanish{
  \renewcommand\appendixname {Appendiks}
}
% slut preamble

% start appendix
\appendix
\appendixpage

\include{appendix1}
...
% slut appendix

% start bilag
\renewcommand\appendixtocname {Bilag}
\renewcommand\appendixpagenamename{Bilag}
\addtocontents{toc}{\protect\renewcommand\protect\cftappendixname{Bilag~}}
\renewcommand\appendixname {Bilag}
\appendix
\renewcommand\thechapter{\Roman{chapter}} % bilag med store romertal
\appendixpage

\include{bilag1}
...
% slut bilag
```

Eksempel

12.36



Preamble-delen er ganske almindelig, desuden har vi pillet ved appendixforsiden således at den skriver »Appendiks« i stedet for »Appendices« (samme rettelse for teksten den skriver i indholdsfortegnelsen). I bilagsdelen laver vi samme smøre, dog bliver vi nødt til at lave ændringen af \cftappendixname direkte inden i indholdsfortegnelsen, idet værdien skal ændres midt i selve indholdsfortegnelsen. Da \appendix laver om på måden hvorpå kapiteltælleren ser ud, skal ændringen af \thechapter naturligvis ske *efter* \appendix.

Sidebemærkning 12.18. Konstruktionen i [Eksempel 12.36](#) fungerer ikke ordentligt sammen med hyperref i forbindelse med bookmarks og indholdsfortegnelsen. Den interne markør som hyperref anvender til appendixer skal være unik, og det bliver den ikke i den konstruktion vi præsenterede i [Eksempel 12.36](#). Løsningen er at lave nogle små tilføjelser:

```
\makeatletter
\def\Hy@appendixstring{bilag}
\makeatother
```

før bilagene (i.e. før første \appendix) og før den anden \appendix placeres

```
\makeatletter
\def\Hy@appendixstring{appendix}
\makeatother
```

så peger indholdsfortegnelsen og bookmarks igen de rigtige steder hen.

Eksempel

12.37

Eksempel

12.38

12. Dokumentopsætning med memoir-klassen

Referere til en overskrift

Hvis man som i denne bog vælger ikke at nummerere eksempel vis `\subsubsection`, så er det lidt besværligt at referere til netop disse afsnit. Dette kan man i *memoir* løse via

```
\titleref        \titleref{<label key>}
```

Dette vil så give overskriften af det afsnit som lablen hører til. Eksempel

Eksempel 12.39

```
Navnet på det nuværende afsnit er  
>>\titleref{sec:referere-til-en}<<.
```

```
Navnet på det nuværende afsnit er »Referere til en overskrift«.
```

Den kan anvendes til alle labels, dog er det nok kun de som henviser til en overskrift som faktisk vil give mening.

Markere noget nyt via en blank linie

Vi anbefaler *meget* kraftigt at man *ikke* markerer nyt afsnit via en blank linie i outputet.²² Men der kan nemt være situationer hvor det er på sin plads at man markerer starten på noget nyt via en blank linie. I *memoir* kan dette gøres f.eks. via

```
\fancybreak        \fancybreak{<fyld>}
```

`<fyld>` kan man lade være tom. I denne bog anvendes en variation af følgende makro:

Eksempel 12.40

```
\newcommand\starbreak{\fancybreak{%  
  \starredbullet\quad\starredbullet\quad\starredbullet}}
```

Hvor den her anvendte makro `\starredbullet` stammer fra *fourier*-pakken. Man kan også sagtens anvende f.eks. `$$$` eller forskellige tekst symboler.

Linien efter `\fancybreak` bliver ikke indenteret, helt i tråd med at første linie efter en `\section` heller ikke bliver det.



Mangler man andre sjove tricks i *memoir*, kan man bare spørge. *Memoir* kan mange ting, langt flere end vi kan nå og nævne her. Og hvis det ikke (endnu) kan så kan vi sikkert få den til det.

²² Dette er noget jeg øjeblikligt betyder at rette når studerende kommer og skal have hjælp. Så lad være med at bruge det og lad være med at spille min tid.

Generelle gode råd vedr. forskellige projekter

I dette kapitel vil vi komme med nogle generelle råd og forslag, både af generel karakter, emn også mere specielt til f.eks. forfattere af undervisningsmateriale.

13.1 Dødssynder

Følgende er en liste over nogle af de dødssynder man bør undgå i sine L^AT_EX-projekter. (Ikke prioriteret)

- (1) Fjernelse af afsnitsindenteringen. Der er en grund til at dette er standarden i T_EX, så lad den være. Det samme gælder forøget afstand mellem afsnit. Det gør ikke andet end kunstigt at gøre teksten længere. Anvend i stedet afsnitsindentering til at markere nyt afsnit. Se evt. afsnit E.1 på side 407.
- (2) Enhver brug af »\« i selve teksten. Anvend et nyt afsnit i stedet, eller skriv teksten om. »\« er OK at anvende inden i argumenter til andre makroer, eller indenfor almindelige environments.
- (3) Brugen af environmentet *eqnarray*, se [Madsen \(2006\)](#). Konstruktionen giver inkonsistent matematisk typografi.
- (4) Floats uden ordentlig angivelse af positionering. Enten sørg for at sætte standardlisten til »htbp« eller angiv selv »[htbp]« eller »[htb]«. Ellers risikerer man nemt at figuren flyder alt for langt væk. Standarden indeholder ikke »h«, så derfor skal man enten selv indsætte det, eller ændre på standarden.
Bemærk »[h]« eller »[!h]« gør ikke det du tror.
Hvis man synes det er sjovt hele tide at anvende »H«, så burde man nok overveje at blive i Word, lær i stedet de korrekte float placeringer.
- (5) Betitlede krydsreferencer uden tie. Sørg altid for at alle krydsreferencer med titel er limet sammen med ordet som kommer før.

Sætning~\ref{thm:1}

Andre steder hvor ties er en god ide er:

formlerne~\eqref{eq:1} og~\eqref{eq:4}.

Så slipper man for problemer med linieombrydning lige før en reference. Specielt relevant, hvis man får brug for at ændre på dokumentets dimensioner.

- (6) Enhver brug af $\$$. . . \$-$ -konstruktionen (eneste undtagelse er hvis man gør brug af nath-pakken). Konstruktionen er ikke kompatibel med *fleqn*-klasseoptionen samt `\qedhere` fra *amsthm*-pakken. Desuden gør brugen af environments kildeko-den meget nemmere at overskue, især hvis man altid sørger for at skrive `\begin` of `\end` delene på linier for sig selv. Man kan evt. kigge på *onlyamsmath*-pakken, som kan anvendes til at tvinge et dokument til at anvende $\mathcal{M}\mathcal{S}$ -konstruktioner.

13. Generelle gode råd vedr. forskellige projekter

- (7) Hjemmelavede overskrifter, altså noget lignende:

Eksempel 13.1

```
\noindent\textbf{bla bla}
```

Brug sektionens makroerne det er det de er der for, hvis man ikke ønsker noget nummer så anvend den stjernede-udgave, eller konfigurér dokumentet til ikke at anvende nummerering.

- (8) Blanke linier før fremhævede formler. Disse kan ofte ses i outputtet, ønsker man noget luft i kildekoden, så start de blanke linier med et %.

13.2 Værd at huske mens man skriver

- (a) Lav genveje (makroer) til de konstruktioner og mængder du ofte anvender og anvend konsekvent denne genvej. Dette gør det nemmere at ændre navnet på en mængde uden at skulle lave *søg og erstat*. Vælg navne som giver mening i sig selv, uden man har læst definitionen.
- (b) Hvis mens man skriver opdager tekstlinier som bliver for lange (går ud i marginen), kan også være inline matematik, så lad være med at rette dem her og nu. Chancen er at du alligevel senere kommer til at rette noget af teksten (eller marginerne), hvilket betyder at dine rettelser kan være nytteløse. Brug hellere tiden til at skrive din tekst. Vent med at rette disse til du er færdig med at skrive dit dokument og har fastlagt dine marginer/bredden og højden på tekstblokken. Fremhævet matematik kan man godt rette lidt på undervejs.
- (c) Husk brugen af tegnet »~« (tilde), se afsnittet »*Tvungen mellemrum uden linieombrydning*«, på side 28.
- (d) Hvis du skriver på *dansk* og skriver mange sammensatte ord, så væn dig til at skrive disse som »ord"-ord« i stedet for »ord-ord«, dette vil hjælpe på orddelingen.
- (e) Der må gerne være blanke linier før og efter `\chapter`, `\section`, etc., det sker der ikke noget ved, ved at indføre disse blanke linier omkring disse overskrifter bliver det nemmere at navigere rundt i kildekoden.
- (f) Der må *aldrig* være blanke linier i kildekoden *før fremhævede formler*, dette vil man kunne se i outputtet. Det er især en dum ide i forbindelse med *equation*-environmentet, da denne har specielle features som afhænger af længden på linien lige før formlen.
- (g) Med mindre indholdet af et environment er meget kort bør man *altid* skrive `\begin` og `\end` på linier for sig selv, således at de er det eneste som optræder på disse linier. Dette gør koden mere overskuelig og langt nemmere at editere for dig selv eller andre.
- (h) Når man i tabeller eller f.eks. i en *align* anvender `\\` til at skifte række, så er det en god ide at skrive `\\` på en linie for sig selv, igen gør dette koden nemmere at overskue, især ved meget komplicerede matematiske konstruktioner eller store tabeller.
- (i) Brug `\left... \right` med omtanke, i mange tilfælde er det bedre et skalere hegn manuelt. Brug *aldrig* i den almindelige tekst, `\left... \right` kommer nemt til at forstyrre linieafstanden. Skalerede hegn bør i den almindelige tekst ikke være større end de man får via `\bigl... \bigr`. Det kan anbefales at anvende

Værd at huske når man er ved at være færdig med at skrive

`\DeclarePairedDelimiter` fra `mathtools` pakken til at kan konstruktioner så som `\abs` eller `\norm`.

13.3 Værd at huske når man er ved at være færdig med at skrive

- (i) Bestem dig for hvilken teksthøjde og -bredde du ønsker eller hvilke marginer. Hver gang du ændrer disse får du nye side- og linieombrydninger, så det er vigtigt at bestemme sig for disse før man går videre med det afsluttende tjek, ellers skal man begynde forfra med de tjek som er beskrevet herunder.¹
- (ii) Start med at rette overflydende linier. Anvend evt. klasseoption »*draft*« (kan erstattes med `\setlength{\overfullrule}{10pt}` i preamble, thi *draft*-klasseoption, kan f.eks. få ting inkluderet via listings til at forsvinde midlertidigt), sæt desuden `\hfuzz=4pt`.
- (iii) Ved for lang matematisk tekst huskes
 - \LaTeX skal gives lov til at dele ved komma, skriv evt. `$1,\dots,n$` som `$1,\dots,$n$` eller som `$1,\allowbreak\dots,\allowbreak n$`.
 - I teksten er det tilladt at bryde linien *før* en relation, \LaTeX vil kun automatisk dele *efter*, så her kan `\allowbreak` igen anvendes.
 - Ofte er det bedst at skrive noget af teksten om, eller omformulere den lidt.
 - I desperate situationer kan man fremtvinge en 'pæn' linieombrydning via `\linebreak`.² Den vil strække linien så vi fortsat har lige højremargin, den giver dog ikke sikkerhed for pæne løsninger.
- (iv) Hvis der er sider som er forstruktet lidt rigeligt, pga. en stor udregning som ikke kan være på siden, så kan det være en ide at indsætte små kommentarer via `\intertext` (eller `\shortintertext`). Man kan vælge at dele en lang udregning over to sider, men det bør kun foregå hvis: (1) Man arbejder med et tosidet dokument, og (2) kun hvis sideskiftet kommer til at gå fra en venstre til en højre side. Det er besværligt for læseren hvis han/hun skal bladre i dokumentet for at finde resten af udregningen.
- (v) Start nu igen helt forfra i dokumentet, og hold nu øje med venstre- og højremargin
 - I venstre side holdes øje med udtryk hvor det ville have været smartere hvis man havde anvendt »~« til at binde f.eks. to ord sammen så de bliver på samme linie, eller i det mindste ikke deler linien mellem dem. Dette gælder desuden tal og korte matematiske udtryk. Vær desuden på vagt over for dette i slutningen af afsnit.
 - Højre side tjekkes efter dårlige orddelinger samt uhensigtsmæssige ombrydninger af matematik (e.g. $n = næste\ linie\ 1$).
 - Tjek desuden før og efter sideskift. Man skal være på udkig efter de såkaldte *horeunger* og *enker*. En horeunge er når man lige når at have første linie fra et afsnit med lige før en side ombrydning (huskereglen er at den ikke har nogen fortid). En enke er når den sidste linie af et afsnit er det første på en side.

¹ Har man valgt at angive tekstbredden og -højden direkte i stedet for via angivelse af marginerne, så vil placering af tekstblokken ikke ændre på linieombrydningerne.

² \TeX Note: det vil sikkert også være en ide at nævne `\sloppy`

13. Generelle gode råd vedr. forskellige projekter

Her kan man enten anvende `\newpage` (eller `\pagebreak`) til horeungerne, eller `\enlargethispage{<længde>}` før afsnittet, hvis det er en enke. Der er desuden to såkaldte *penalties* som styrer ‘muligheden’ for horeunger og enker. De er `\clubpenalty` og `\widowpenalty` og behandles ikke nærmere her.

`\enlargethispage`

13.4 Til noteskribenter

Skriver man forelæsningsnoter skal man naturligvis overholde alle de anbefalinger vi har nævnt tidligere. Et notesæt skal som oftest behandles (formateres) på samme niveau som en bog eller et speciale.

Det vigtigste en noteskribent skal huske er

»Dine læsere vil slavisk kopiere alt hvad du skriver, hver en formatering af sætninger, formler og lignende. Det er derfor vigtigt at man gør sig umage med at sørge for at tingerene er i orden således at læserne lærer nogle gode vaner. Især i en verden hvor god stil er på kraftigt tilbagetog, det er synd at 400+ års typografisk tradition skal forspildes pga. uvidenhed.«

På Matematik er man mere end velkommen til at anvende undertegnede som filter til at sørge for at noter og anden undervisningsmateriale har et konsistent udseende og layout.³

13.5 Til artikelskribenter

Med mindre man selv skal lave hele designet, bør man *ikke* anvende *memoir* til videnskabelige artikler. I stedet bør man⁴

- (1) Hvis du allerede ved hvilken journal du vil indsende til så find ud af om denne journal understøtter $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, og om de i så fald har en opsætning man skal anvende. Følg så denne opsætning til punkt og prikke.⁵

Hvis deres skabelon nævner environmentet *eqnarray*, så brug environments fra *amsmath* i stedet.

- (2) Har journalen ikke nogen anbefalinger, eller ved du ikke hvilken journal du gerne vil sende til, er anbefalingen:
 - 2a. Brug *article*-klassen.
 - 2b. Pil *ikke* ved marginerne, men det kan være en fordel at anvende *a4paper*-option, så den er til at printe på en dansk printer.
 - 2c. Lad den forblive i 10pt font størrelse.
 - 2d. Brug (mindst) pakkerne *amsmath*, *amssymb* og *amsthm*. De kan evt. krydres med pakkerne *bm* og *mathtools*
 - 2e. Brug ellers kun pakker som er tilgængelige på CTAN. Anvender du hjemmelavede pakker, skal du huske at vedlægge dem når du sender artiklen ind.

³ Det er blandt andet den slags jeg får min løn for.

⁴ Speciel tak til Barbara Beeton, AMS, for forslag til listen.

⁵ Er der noget du i den forbindelse er i tvivl om, så send mig en mail.

Til artikelskribenter

- 2f. Journaler er ikke altid lige glade for folk som redefinerer ting i L^AT_EX, så man bør undgå brugen af `\renewcommand` eller `\def`. Generelt bør man kun tage de ting med i artiklen som man faktisk anvender.
- 2g. Dokumenter dine egne makroer således at editorene også ved hvad de gør. Dette kan især være vigtigt i forbindelse med proceedings hvor mange artikler sammenflettes (inkl. preamble).
- 2h. Husk når du er færdig så bør artiklen passe inden for de marginer som *article*-klassen anvender som standard.

Tip 13.1. Ofte er man flere om at skrive en artikel, man får derfor brug for at kunne angive hvor de enkelte forfattere kommer fra. Der er nogle af de mere specialiserede klasser (f.eks. *amsart*) som har forskellige interfaces til dette, men det har *article* ikke. Her kan pakken `authblk` hjælpe.⁶ Pakken gør at `\author` kan anvendes flere gange og giver os mulighed for at angive tilhørsforhold.

```
\usepackage{authblk}
\renewcommand\Affilfont{\small}
% vil ikke have ', and ' men ' and '
\let\Authands\Authand

----- slut preamble -----

\title{My title}
\author[1]{An Author}
\author[2]{Another Author}
\author[2]{A Third Author}
\affil[1]{Department-of-Mathematics
Aarhus-University,~anauthor@imf.au.dk}
\affil[2]{CSGB, bb@csgb.dk, cc@csgb.dk}
\date{}
\maketitle
```

My title

An Author¹, Another Author² and A
Third Author²

¹Department of Mathematics
Aarhus University, anauthor@imf.au.dk
²CSGB, bb@csgb.dk, cc@csgb.dk

Eksempel 13.2



⁶ Speciel tak til Ute Hahn for det implicite tip om denne pakke.

Om justering af standardklasserne

Vi har i et tidligere kapitel set at *memoir*-klassen næsten kan konfigureres til hvad som helst. I dette kapitel vil vi se på nogle metoder som man så man anvende på f.eks. standardklasserne.

Vi vil se på nogle standard konstruktioner til at ændre på marginer, headere, indholdsfortegnelse, overskrifter samt fikse et par ting som standardklasserne ikke laver ordentligt.

Vi vil ikke gå så langt ned i detaljerne, men i stedet henvise til de respektive pakkers brugermanualer, som kan findes via litteraturlisten.

14.1 Opsætning af marginer

Sidebemærkning 14.1 (Standardmarginerne anvendt i L^AT_EX).

Nye L^AT_EX-brugere bliver tit forvirrede over de marginer/tekstbredde og -højde som klasserne anvender som standard. Der er især to ting man undres over: (a) Teksten er ret smal, og (b) i et to-sidet dokument er den indre margin mindre end den ydre. Burde det ikke have været omvendt?

Teksten er smal fordi dette er mere behageligt at læse. Man kan diskutere om den ikke er lidt for smal, og personligt har jeg da ændret bredden i denne bog. Men eksperimenter viser at en tekstbredde svarende til mellem 45 og 75 tegn pr. linie er det optimale. Se [Wilson \(2010\)](#), afsnit 2.3.

(b) er typografisk tradition og har været det gennem de seneste 500år! [Wilson \(2010\)](#) har en større gennemgang af de forskellige traditioner gennem tiden.

Har man mulighed for at besøge biblioteket på Institut for Matematiske Fag, Aarhus Universitet, så skulle man tage sig tid til at tjekke bachelorprojektet samt specialet for Emil Hedevang Lohse Sørensen. De benytter begge en forholdsvis smal tekstbredde og er begge af en meget høj typografisk (og faglig) kvalitet, se [Hedevang \(2004, 2006\)](#).

Desværre er der mange skabeloner i omløb som enten ikke har sat marginerne ret godt op eller direkte anvender hovedløse konstruktioner. Det, man kan komme ud for, er at skabeloner eller brugere sætter marginerne op ved manuelt at pille ved en masse længder. Dette kan nemt gå hen og give problemer, så her er nedenstående løsning bedre. Lad os derfor se på hvordan man sætter marginer op på en simpel måde.

Visse klasser har selv et system til at sætte marginer op med, f.eks. *memoir* mens KOMA-systemet anbefaler pakken `setspace`. Vi vil i stedet præsentere pakken `geometry` (marginopsætningne med *memoir* kommer vi til i afsnit 12.3.2 på side 306). Pakken kan anvendes på to måder:

```
\usepackage[options]{geometry}
```

eller via

```
\usepackage{geometry}
```

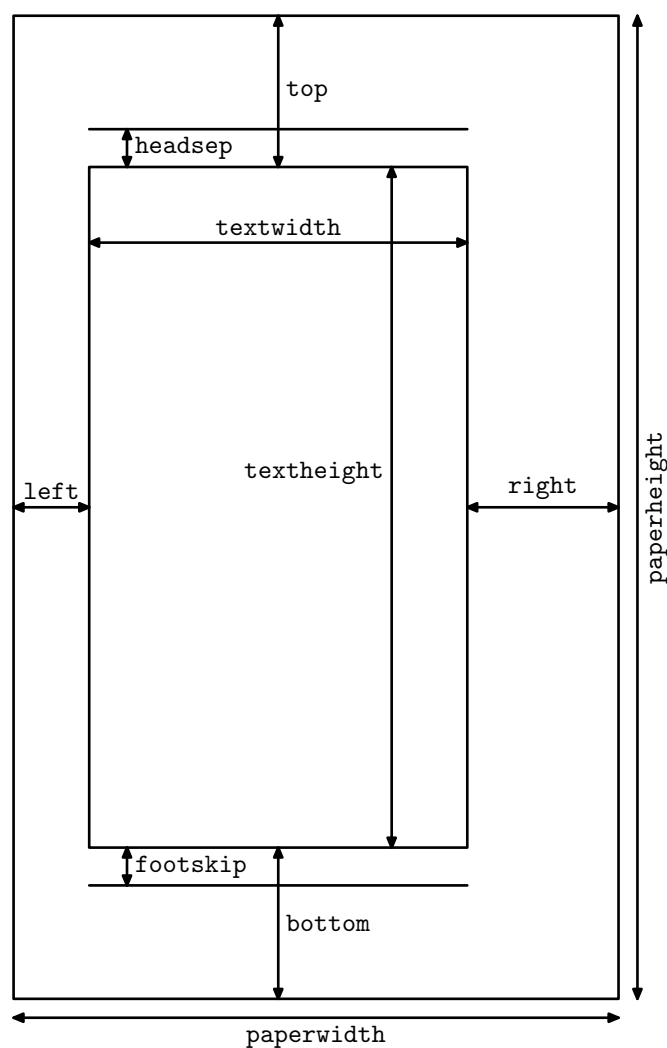
```
\geometry{options}
```

`setspace`
`geometry`

`\geometry`

14. Om justering af standardklasserne

Pakken `geometry` anvendes hovedsageligt ved at man angiver nogle længder som værdi til nogle af pakkens nøgleord, hvorefter pakken omformer disse informationer til ændringer af de rette L^AT_EX-længder. Pakken har en indbygget *auto-completion*¹ feature som gør at man kun behøver f.eks. at angive størrelsen på hhv. venstre- og højremargin hvor efter tekstbredden automatisk beregnes. Eller man kan angive venstremargin, topmargin og tekstbredden så bliver resten automatisk justeret derefter. De vigtigste $\langle key \rangle$ til $\langle key \rangle = \langle længde \rangle$ -par ses på figur 14.1.



Figur 14.1: Oversigt over fortolkningen af options til `geometry`-pakken.

Følgende liste indeholder er par forklaringer samt nogle ekstra options. $\langle options \rangle$ -delen af syntaksen på side 337 er en komma-separeret liste af $\langle keywords \rangle$ og/eller $\langle key \rangle = \langle længde \rangle$ -par.

`headheight`= $\langle længde \rangle$: højden af headerfeltet (ikke vist på tegningen).

`headsep`= $\langle længde \rangle$: afstanden mellem toppen af tekstblokken og bunden af headerfeltet.

¹ *Auto-completion* betyder her at har man angivet *visse* af værdierne hørende til figur 14.1, da sørger pakken selv for at fastlægge de resterende længder ud fra disse værdier.

B5 på A4

`footskip=<længde>`: afstanden mellem bunden af tekstblokken og *bunden* af området hvori footeren placeres, der findes ikke nogen længde som styrer højden af footerfeltet.

`margin=<længde>`: sætter automatisk `left = right = top = bottom = <længde>`

`paperwidth/paperheight=<længde>`: anvendes når man ønsker at anvende ikke standard papirstørrelser. Pakken fanger normalt selv papirstørrelsen (`a4paper`) direkte fra dokumentklassen.

`landscape`: kan selvfølgelig også angives som dokumentklasse option. Se desuden næste punkt.

Sidebemærkning 14.2. Når man sætter marginerne op med `geometry` er det altid en højre (ulige) side man sætter op.

Husk, anvendes dobbeltsidet setup (klasse option `twoside`) da vil alle lige (venstre) sider have venstre- og højremargin byttet om i forhold til en højreside.

twoside

Sidebemærkning 14.3. Der findes mange flere options til `geometry`, bl.a. papirstørrelsen, se [Umeki \(2002\)](#) for mere information.

Et meget simpelt eksempel på brugen af `geometry`, er følgende som jeg anvender til mit instituts preprintserier:

```
\usepackage[margin=3cm]{geometry}
```

Eksempel 14.1

Eller hvis man ønsker 3.5 cm i venstre side, 3 cm i toppen og i højre side samt 4 cm i bunden.

```
\usepackage{geometry}
\geometry{left=3.5cm,top=3cm,right=3cm,bottom=4cm}
```

Eksempel 14.2

Så anvendelsen af `geometry` er *meget* simpel, samt en hel del nemmere at holde styr på i sammenligning med at skulle fastsætte alle parametrene pr. håndkraft.

B5 på A4

Mange ph.d.-studerende vælger at aflevere deres afhandling i B5 format. I mange tilfælde er dette ikke noget problem for et trykkeri, man beder dem bare om at tilpasse materialet til B5 format. Hvis de gør det rigtigt vil der ikke være noget problem i at aflevere et dokument sat op som A4 med 12 pt fontstørrelse. De gange vi har gjort det hos SUN-TRYK ved AU, har resultatet været rigtig godt.



Har man ikke en trykker som kan håndtere omskaleringen til B5, eller man selv ønsker at styre alt, så må man arbejde en del for det. Sørg for at finde ud af hvordan trykkeren (eller andre) ønsker B5 siderne placeret på papiret.

Vi ser først på hvordan man fikser dette med `geometry` og dernæst med `memoir`. I det følgende ønsker vi at få placeret B5 området således at man ved beskæringen efter tryk, skal beskære ved yderste kant samt i bunden.

Det største problem er faktisk at når man printer tosidet så skal B5 papiret placeres til venstre på en ulig side, men til højre på en lige. Da `twoside`-klasseoptionen sørger for at skifte højre- og venstremargin, er dette dog ikke noget større problem. B5 papir har en højde på 250 mm og en bredde på 176 mm, til sammenligning er A4 297 × 210 mm.

14. Om justering af standardklasserne

Når man har besluttet sig for sine marginer for B5-dokumentet, lad os kalde dem l , r , t og b , så er formlen man skal anvende i geometry altså

Eksempel 14.3

```
left = l
top = t
right = <r plus 34 mm>
bottom = <b plus 47 mm>
```

– og så skal man naturligvis huske at benytte *twoside*-klasseoption.

Med *memoir* er det endnu mere simpelt. Vi antager at man beder *memoir* om et B5 design, herefter kan man bygge videre på dette eksempel

Eksempel 14.4

```
\documentclass[b5paper,showtrims]{memoir}
\stockaiv % antag A4 papir som basis at trykke paa
% trim ned til B5 og placer op til venstre (skifter automatisk til
% hoejre side paa lige sider)
\setpagetl{\paperheight}{\paperwidth}{*}
% indsaet marginerne i forhold til B5 her
\checkandfixthelayout % fastlaeg layout
\usepackage{lipsum} % fyld tekst
\begin{document}
\lipsum
\end{document}
```

Done!

14.2 Sidehoved og -fod

Alle sider i et L^AT_EX dokument har et sidehoved og -fod, til tider kaldet en løbende header/footer² – disse kan dog være tomme. I L^AT_EX er disse to områder styret af begrebet *pagestyle*. Sidehoved/fod kan enten indeholde statisk information (kan anvendes til små projekter) eller dynamisk genereret information, hvilket kunne være sidetallet eller informationer om nuværende kapitel- eller afsnitstitel.

Begrebet *pagestyle*

En given *pagestyle* kan aktiveres på to forskellige måder, enten globalt (gældende fra næste side og frem) eller lokalt hvilket kun gælder på denne side.

```
\pagestyle      \pagestyle{<navn>}
\thispagestyle  \thispagestyle{<navn>}
```

\pagestyle{<navn>} aktiverer *pagestylen* *<navn>*, men først fra *næstkommende* side! *\thispagestyle{<navn>}* aktiverer *pagestylen* *<navn>* på den nuværende side og *kun* denne side. Bemærk at det er den *sidste* *\thispagestyle* på en given side som er gældende. *\maketitle* og *\chapter* anvender begge denne feature. Efter en side, hvorpå man har anvendt *\thispagestyle*, vil den anvendte *pagestyle* så falde tilbage på den *pagestyle* som senest er udvalgt med *\pagestyle{<navn>}*. Har man ikke selv valgt en, så har klassen sikkert automatisk indlæst en.

De mulige *pagestyles* kommer an på klassen man anvender, men der er tre styles man normalt finder i de fleste klasser:

² Bemærk at footeren *ikke* har noget med fodnoter at gøre.

Justering af pagestyle med pakken fancyhdr

empty tomt sidehoved og fod.

plain tomt sidehoved og sidetallet centreret i footeren. Denne anvendes ofte (automatisk) til sider lavet med `\chapter` eller `\maketitle`.

headings tom sidefod, side og afsnitsinformation i headeren, al tekst er i (kursive/slantede) kapitaler (i.e. store bogstaver). Dette er normalt standard pagestyle i *book* og *memoir*.

Tip 14.4. Antag at man laver en aflevering som kun fylder en side og at man har skrevet sit navn via `\maketitle`, så får man et sidetal og denne kan ikke fjernes ved at placere `\pagestyle{\empty}` i preambelen. Da `\maketitle` internt anvender `\thispagestyle{empty}` skal vi stedet anvende

```
\maketitle
\thispagestyle{empty}
```

så forsvinder sidetallet.

**Eksempel
14.5**

Den opmærksomme læser har måske indset at standardklasserne *book*, *report* og *article* ikke er videre fleksible, eller giver nogen muligheder for konfigureringsmuligheder. Dette gør sig også gældende vedrørende behandling af sidehoved og -fod. Den traditionelle måde at løse problemerne med (i forbindelse med standardklasserne) er via pakken *fancyhdr*-pakken, hvilket vi vil behandle i næste afsnit.

fancyhdr

Vi vil også komme ind på en nogle problemstillinger vedrørende standardklasserne og deres behandling af sidehoved og -fod, helt specifikt vedrørende ikke-blanke sider samt auto-kapitaleret sidehoved.

Sidebemærkning 14.5. Anvender man *memoir*-klassen, så skal man *ikke* anvende *fancyhdr*-pakken, anvend i stedet de indbyggede features i *memoir*-klassen (der er masser) til håndtering af sidehoved og -fod. *Memoir* kan langt mere end hvad *fancyhdr* umiddelbart kan tilbyde og har rettet mange af problemerne som standardklasserne døjer med.

Justering af pagestyle med pakken fancyhdr

Vi vil kun kort beskrive hvordan man kan lave en pagestyle med *fancyhdr* og i stedet henvise til [van Oostrum \(2004\)](#) for flere detaljer.

Først skal man lige aktivere pagestylen **fancy**,

fancy

```
\usepackage{fancyhdr}
\pagestyle{fancy}
```

**Eksempel
14.6**

hvorefter vi kan konfigurere stilens interne dele via makroerne

```
\fancyhf[⟨placering⟩]{⟨kode⟩}
```

`\fancyhf`

```
\fancyhead[⟨placering⟩]{⟨kode⟩}
```

`\fancyhead`

```
\fancyfoot[⟨placering⟩]{⟨kode⟩}
```

`\fancyfoot`

```
\renewcommand\headrulewidth{⟨tykkelse⟩}
```

`\headrulewidth`

```
\renewcommand\footrulewidth{⟨tykkelse⟩}
```

`\footrulewidth`

14. Om justering af standardklasserne

Af forskellige årsager skal man huske at aktivere pagestylen **fancy** før man ændrer noget (ændringerne bliver nemlig ikke gemt i pagestylen).³

Det valgfri argument (ovenfor kaldet *placering*) styrer hvor *<kode>* skal placeres i sidehoved eller -fod. Det er en kommasepareret liste af tekststreng på op til tre tegn, taget blandt:

	Side	Intern placering	Hoved/fod
E	Lige side	L Venstre del	H Header
O	Ulige side	C Center del	F Footer
		R Højre del	

`\fancyhead` `\fancyhf` `\fancyhead` er så bare `\fancyhf`, hvor H automatisk er valgt. Hvis der ikke angives nogen *<placering>* så gælder *<kode>* for alle felter, se [Eksempel 14.7](#).

Eksempel 14.7



```
\usepackage{fancyhdr}
\fancyhf{} % tom header/footer
\fancyhead[OL,OR]{XXX}
\fancyhead[OC]{YYY}
\fancyfoot{Alle tre}
\renewcommand{\headrulewidth}{0.4pt}
\renewcommand{\footrulewidth}{0.4pt}
\pagestyle{fancy}
```

————— slut preamble —————
Noget tekst

XXX	YYY	XXX
Noget tekst		
Alle tre	Alle tre	Alle tre

4

Det nemmeste er at anvende `\fancyhead` og `\fancyfoot`. Som man kan se sættes placeringen sammen således at de angiver helt specifikke steder (OL er eksempel venstre del af sidehovedet på en ulige side) og disse separeres med komma. Man kan også se via [Eksempel 14.7](#) hvad der sker hvis man ikke angiver en placeringsstreng. Ønsker man ikke nogen linier skal man bare sætte `\headrulewidth` og `\footrulewidth` lig `0pt`.

Med [Eksempel 14.7](#) og *article*-klassen kan man nu meget nemt lave en skabelon til afleveringer hvor man (for at spare plads) har placeret ens navn samt anden information i header og footer.

Eksempel 14.8



```
\usepackage{geometry, fancyhdr}
\geometry{headheight=2cm}
\pagestyle{fancy}
\fancyhf{}
\fancyhead[L]{Navn\Calculus, Hold 3}
\fancyhead[R]{Aflevering 2\ 2007/11/06}
```

————— slut preamble —————
`\section*{Opgave 34}`
Trivielt.

Navn	Aflevering 2
Calculus, Hold 3	2007/11/06
Opgave 34	
Trivielt.	



headings

Det mest almindelige er at man anvender hjemmelavede pagestyles til større projekter, hvor man har overskriftsinformation stående i sidehovedet. For de flestes vedkommende er informationerne, man finder i headerne anvendt med stilen **headings** stor set det

³ I *memoir* er det lige omvendt, her skal man huske at aktivere pagestylen *after* man har ændret noget, dette skyldes at *memoir* gemmer ændringerne.

⁴ FixMe Note: det ville være smart her at have en funktionalitet hvor man kunne vise en højre og en venstre side samtidigt

Justering af pagestyle med pakken fancyhdr

de har brug for, men det skal bare ikke lige se sådan ud. Så lad os lige se på hvordan vi fikser sådan noget.

Her kommer (desværre) lidt teori. De information vi gerne vil anvende er sidetallet samt titlerne (samt evt. numrene) fra `\chapter`, `\section`, `\dots`, `\subsubsection`. Disse stilles til rådighed via makroerne:

```
\thepage  
\leftmark  
\rightmark
```

hvor `\leftmark` og `\rightmark` sættes internt i afsnitmakroerne via kommandoerne

```
\markright{\info til \rightmark}  
\markboth{\info til \leftmark}{\info til \rightmark}
```

bemærk at `\leftmark` kun kan sættes via `\markboth` (anvendes normalt kun internt i `\chapter`). Nu kommer teorien så:

- På en given side indeholder `\leftmark` den *sidst-kommende* tekst angivet via det første argument til `\markboth`, hvis der *ikke* er anvendt nogen `\markboth` på siden, da anvendes teksten fra den foregående side.
- På en given side indeholder `\rightmark` den *først-kommende* tekst angivet via `\markright` eller via argument 2 til `\markboth`. Hvis der ikke er nogen angivet på den nuværende side, anvendes den sidste tekst fra den foregående side.

Dette kan godt give en smule forvirring hvis man begynder at lave meget specielle headere. Men holder man sig til at angive kapiteltitler i sidehovedet på venstre sider, samt afsnitsoverskrifter i sidehovedet på højre sider, så fungerer det helt fint.⁵

Til brug ved f.eks. *book*-klassen (som har kapitler) kan man f.eks. anvende

```
\usepackage{fancyhdr}  
\fancyhf{} % slet alt  
\fancyhead[LE,R0]{\thepage} % sidetallet yderst  
\fancyhead[RE]{\leftmark} % lige side, kapitel titel  
\fancyhead[LO]{\rightmark} % ulige side, afsnitsinformation  
\renewcommand{\headrulewidth}{0.4pt}
```

Eksempel

14.9



Der man kan så selv bestemme størrelsen på teksten og om det skal være fed, etc. Det vil vi give et eksempel på herunder.

Lad os lige tage et kig på hvordan et sådant sidehoved med dynamisk information kommer til at se ud. Vi anvender *article*-klassen (fordi det gør vi ved alle vore eksempler), her vil fancyhdr sørge for at det er `\section` til skrive til `\leftmark`. Desuden er det et enkeltsidet layout vi leger med, så »E« vil ikke give mening. Bare for sjov skyld vil vi sætte sidetallet i kursiv og den dynamiske tekst i fed.

⁵ Dette er hvad de fleste gør.

14. Om justering af standardklasserne

Eksempel 14.10



```
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\fancyhead[RO]{\itshape\thepage}
\fancyhead[LO]{\bfseries\leftmark}
\renewcommand{\headrulewidth}{0.4pt}
----- slut preamble -----
\section{Test}
noget tekst
```

1 TEST	1
<hr/>	
1 Test	
noget tekst	

Her kan vi se det primære problem folk har med den en måde den dynamiske tekst skrives på. Det er rene kapitaler, hvilket ikke er videre pænt. Dette vil vi kigge på et det næste afsnit.

Standardklassernes problemer vedr. sidehoved og -fod

Dette er en liste over de mest almindelige problemer folk har med sidehoved og -fod.

- (i) Den dynamiske tekst som kan anvendes i f.eks. sidehovedet (f.eks. overskrifter) sættes automatisk i kapitaler og dette er hardwired⁶ ind i klassekoden. I standardklasserne er det faktisk så slemt at dette ligger dybt inde i `\chapter`, `\section`, ..., `\tableofcontents`, litteraturlisten og indekset. I *memoir*-klassen er disse lavet via specielle små makroer som derfor er nemmere at rette, se afsnit 12.3.3 på side 308.
- (ii) Anvender man f.eks. *book* og har alle kapitler til at starte på højre side (standarden), så vil de fleste komme ud for at kapitlet før sluttede på en højreside, og at der derfor kommer en blank side før starten af det næste kapitel.
Har man sidehoved/-fod i sit layout, så vil denne fortsætte på denne side, hvilket ikke er videre pænt.

Pakken *fancyhdr* har visse ideer til løsningen af begge problemer, men med omkostninger. I den nyeste udgave af *memoir* er dette fikset og page style håndtering er meget meget nemmere i *memoir*.

Problemet med kapitaler

`\nouppercase` Dette løser *fancyhdr* via makroen `\nouppercase`, som så anvendes som på [Eksempel 14.11](#).

Eksempel 14.11



```
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\fancyhead[RO]{\itshape\thepage}
\fancyhead[LO]{%
  {\bfseries\nouppercase{\leftmark}}
}
\renewcommand{\headrulewidth}{0.4pt}
----- slut preamble -----
\section{Test}
noget tekst
```

1 Test	1
<hr/>	
1 Test	
noget tekst	

⁶ Hardwired betyder at denne feature er skrevet så dybt inde i kildekoden at den eneste måde at fjerne den, er ved at kopiere meget store mængder kode.

For lang headertekst

Piet van Oostrum (som har skrevet fancyhdr) advarer om at dette kan gå galt, f.eks. i forbindelse med store romertal (skulle man f.eks. have valgt at nummerere appendikser med store romertal i stedet for med bogstaver).

Problemet med de non-blanke sider

I standardklasserne læser man dette via følgende trick

```
\let\oldcleardoublepage\cleardoublepage
\renewcommand\cleardoublepage{%
\clearpage{\pagestyle{empty}\oldcleardoublepage}}
```

Eksempel
14.12



I *memoir* har man været endnu smartere, her anvendes en speciel pagestyle til disse *blanke* sider, og denne stil kan så omdefinieres hvis man ønsker at den ikke skal være blank.

For lang headertekst

Til tider kan man have valgt afsnitstitler som er for lange til at kunne være i headeren (headere på mere end en linie er ikke pænt). Dette kan man afhjælpe på to måder.

- (i) Makroerne `\chapter` til `\subsubsection` kan tage et ekstra valgfrit argument⁷ som så vil blive anvendt i headeren og i indholdsfortegnelsen. Syntaksen er altså eksempelvis

```
\section[⟨kort titel⟩]{⟨Lang titel⟩}
```

- (ii) Den anden mulighed er at anvende pakken `truncate` til automatisk at begrænse længden af headerteksten. For information om brugen af denne pakke se [Arseneau \(2001\)](#). Eksempel:

truncate

```
\usepackage[hyphenate]{truncate}
```

— slut preamble —

```
\truncate{100pt}{Denne tekst
er blevet delt}
```

Denne tekst er ble- ...

Eksempel
14.13



Sidebemærkning 14.6. Det er også (i) man skal anvende hvis man absolut vil have en fodnote med i en afsnitstitel. Den skal naturligvis ikke med i sidehovedet eller i indholdsfortegnelsen.

Tilsvarende, hvis man gerne vil have en linieombrydning på et meget bestemt sted i en afsnitsoverskrift, så kan man indsætte `\` i den lange titel, mens man udelader den i den korte titel.

14.3 Indholdsfortegnelsen

Indholdsfortegnelse hedder *table of contents* på engelsk og aktiveres via `\tableofcontents` (som tidligere nævnt kan det tage et par kompileringer før den er helt opdateret).

`\tableofcontents`

Man kan sagtens styre hvor mange niveauer man ønsker at tage med i indholdsfortegnelsen, dette gøres via tælleren `tocdepth`:

`tocdepth`

⁷ I *memoir* kan de endda tage to ekstra argumenter, som så giver brugeren fuld kontrol.

14. Om justering af standardklasserne

```
\setcounter{tocdepth}{\langle tal \rangle}
```

hvor $\langle tal \rangle$ svarer til: -1 for ToC til og med `\part`, 0 til og med `\chapter`, 1 til og med `\section` ... Hvad denne tæller er sat til som standard afhænger af dokumentklassen.

Sidebemærkning 14.7. I *memoir* man kan anvende ovenstående, men da det er vanskeligt at huske disse numre, så kan man her anvende navnet på det niveau man ønsker at dybden skal gå til. I *memoir* er standarden at dybden kun går til og med `section`, så dette kan fikses via

**Eksempel
14.14**

```
\settocdepth{subsection}
```

eller hvor langt man nu ønsker.



Skulle man have brug for det kan man også tilføje en indgang til ToC pr. håndkraft via:

```
\addcontentsline{\langle type \rangle}{\langle niveau \rangle}{\langle tekst \rangle}
```

hvor $\langle type \rangle$ oftest vil være `toc`, $\langle niveau \rangle$ er navnet på den afsnitsmakro man ønsker at emulere (i.e. `part`, `chapter`, `section`, etc.). $\langle tekst \rangle$ vil normalt enten være

```
\numberline{\langle en eller anden titel \rangle}
eller
\langle en eller anden titel \rangle
```

Forskellen er nemmest at forstå via et eksempel:⁸

**Eksempel
14.15**



```
\tableofcontents
\section{Test}
\addcontentsline{toc}{section}%
    {Helt til venstre}
\addcontentsline{toc}{section}%
    {\numberline{}Indrykket}
```

Indhold

1 Test	1
Helt til venstre	1
Indrykket	1

1 Test

Sidebemærkning 14.8. Man skal være forsigtig med manuel brug af `\addcontentsline` idet man sagtens kan ende op med en ToC indgang som peger på den forkerte side. Er man i tvivl eller kommet ud i problemer, så spørg den nærmeste L^AT_EXniker.

Se desuden afsnittet »*At få bibliografi og indeks med i ToC*« på modstående side.

En anden makro man kan få brug for, men som *skal* behandles varsomt er

```
\addtocontents{\langle type \rangle}{\langle indhold \rangle}
```

`\addtocontents` anvendes til at skrive $\langle indhold \rangle$ direkte ned i f.eks. `.toc`-filen. $\langle type \rangle$ skal være `toc`, `lof` eller `lot`.⁹ Man vil normalt anvende konstruktionen til f.eks. at tilføje et tvunget sideskift til indholdfortegnelsen. Det kunne f.eks. være at man helst ville

⁸ Vi har snydt med den vertikale spacing for at spare lidt plads.

⁹ Eller en anden type hvis man selv har lavet en ToC lignende liste.

At få bibliografi og indeks med i ToC

have at ToCen blev delt før en bestemt sectionsoverskrift i stedet for efter. Hertil man så anvende

```
\addtocontents{toc}{\protect\newpage}
```

**Eksempel
14.16**

`\protect` skal med for at sikre at makroen ikke bliver ekspanderet (dvs. at den ikke erstattes af de interne definitioner i `\newpage`) inden den skrives til `.toc`-filen.

At få bibliografi og indeks med i ToC

Ligesom ved sidehoved og -fod er der visse ting som standardklasserne har *glemt*. F.eks. kommer bibliografien, indekset eller indholdsfortegnelsen *ikke* med som indgange i indholdsfortegnelsen. Det dur oftest ikke at anvende `\addcontentsline`, da den nemt kommer til at pege på den forkerte side. Anvend i stedet pakken `tocbibind`. Pakken har forskellige pakke-options hvor man så kan fjerne visse af punkterne igen, se [Wilson \(2004b\)](#). Den mest interessante er nok »`nottoc`« som sørger for at indholdsfortegnelsen *ikke* selv er en indgang i ToCen, så den mest almindelige brug af `tocbibind` er:

tocbibind

```
\usepackage[nottoc]{tocbibind}
```

**Eksempel
14.17**

Funktionaliteten af `tocbibind` er allerede indbygget i *memoir*.

Konfigurering af ToC

Man kan af visse årsager få brug for at kunne konfigurere, hvordan indholdsfortegnelsen ser ud, f.eks. lige skrue lidt ned for afstanden mellem punkterne (ses ofte hvis ToCen lige er lidt for lang til at kunne være på én side). Dette gøres nemmest med pakken `tocloft`, se [Wilson \(2003\)](#) for mere information.¹⁰

tocloft

Igen er funktionaliteten direkte indbygget i *memoir* og synktaksen er forklaret i [Wilson \(2010\)](#).

14.4 Overskrifter

For de flestes vedkommende vil standard udseendet af `\chapter... \subsubsection` til at leve med. Men skulle man have lyst til at pille ved det kan man anvende pakken `titlesec`. De mere specialiserede klasser inkluderer egne metoder til at ændre dette udseende (disse metoder bør anvendes i stedet for `titlesec` når man anvender en non-standardklasse).

titlesec

En hurtig ting de fleste gerne vil have er at afsnitsoverskrifter helst skal skrives uden brug af orddeling (det ser nu en gang bedst ud). Dette er ikke standard i f.eks. *article*-klassen, men er meget nemt at opnå med `titlesec`:

```
\usepackage[raggedright]{titlesec}
```

**Eksempel
14.18**

For flere ideer med `titlesec`, se [Bezos \(2002\)](#), man vil bl.a. finde eksempler på forskellige måder `\chapter` kan se ud.

Igen har *memoir*-klassen sit eget system til at ændre den måde hvorpå kapitelforsider ser ud på. Vi kigger lidt på det i næste kapitel. Man kan også tage et kig på [Madsen \(2008b\)](#) hvor undertegnede har samlet en masse ideer (med kode) til kapitelforside i *memoir*-klassen.

¹⁰ Bare det at anvende `tocloft` vil faktisk formindske denne afstand en smule set i forhold til standardopsætningen.

14. Om justering af standardklasserne

Afsnitstitler uden nummer men med i ToC

secnumdepth Hvorvidt et afsnit faktisk bliver nummereret (ved brug af de u-stjernede versioner af `\chapter...``\subsubsection`) afhænger af værdien af tælleren `secnumdepth`, de fortolkede værdier er de samme som for `tocdepth`, se side 345. Dette betyder at man uden problemer kan lave et helt dokument hvor afsnitsoverskrifterne ikke er nummererede, og hvor man samtidigt kan få en autogenerated ToC. Der er heller ingen problemer med at skifte værdien undervejs.¹¹

Eksempel 14.19



```
\tableofcontents
\noindent\rule{\textwidth}{1pt}\par
\section{Test 1}
% nummerer kun til chapter
\setcounter{secnumdepth}{0}
\section{Test 2}
% nummerer kun til section
\setcounter{secnumdepth}{1}
\section{Test 3}
```

Indhold

1	Test 1	1
	Test 2	1
2	Test 3	1
<hr/>		
1	Test 1	
	Test 2	
2	Test 3	

Sidebemærkning 14.9. Kom gerne med forslag til andre pakker man kunne skrive lidt om.

¹¹ Har igen snydt med mellemrummet i [Eksempel 14.19](#).

Yderligere materiale

Om generering af nye makroer

Når man skriver større projekter vil man hurtigt opdage vigtigheden af at kunne lave forkortelser til ofte anvendte konstruktioner eller lave sine helt egne specielle makroer.

Dette kapitel tjener som en mindre introduktion til det at lave makroer i L^AT_EX. Den er på ingen måde fyldestgørende, da man kunne skrive hele bøger om dette emne.

A.1 At lave sin egen makro

Til at lave en ny makro er den grundlæggende makro `\newcommand` med syntaksen¹

```
\newcommand{\langle makronavn \rangle}[\langle antal argumenter \rangle]{\langle kode \rangle}
```

Makronavnet må (normal) kun bestå af bogstaver. Her er store og små bogstaver forskellige, så `\Hest` og `\hest` er to forskellige makroer. Hvis makroen ikke skal have nogen argumenter anvender man enten `[0]` eller helt udelader `[\langle num arg \rangle]`. Koden som makroen skal udføre kan stort set være hvad som helst, med visse undtagelser, så som f.eks. `verbatim`² konstruktioner a la `\verb. \langle antal argumenter \rangle` skal være et tal mellem 0 og 9.

Sidebemærkning A.1. Man vil i koden til mange pakker og klasser se makroen `\def` blive anvendt. Hvis man ønsker at anvende `\def` skal man vide præcist hvad man gør, idet `\def` overskriver allerede eksisterende makroer uden advarsel. Syntaksen vedrørende `\def` og venner er ikke noget vi kommer nærmere ind på, den interesserede læser kan tage et kig på [Knuth \(1986\)](#) eller [Eijkhout \(2001\)](#). Vi har dog et eksempel i afsnit [E.4 på side 411](#).

Navne på makroer

De fleste af de kommandoer man selv kommer til at lave vil tjene som et fast navn for et eller andet, f.eks. et symbol. I den forbindelse er det en god ide at komme på et sigende navn til makroen, et navn som kan lede en på sporet af hvad denne makro har med at gøre.

Vi har allerede set eksempler på dette med f.eks. `\setR` fra afsnit [3.2.5 på side 61](#). Det anbefales at man undlader at lave makronavne som er meget korte og uforståelige for andre end en selv. Det kan godt være man synes det er træls at skulle huske på længere navne eller træls at skulle skrive dem, i så fald er rådet at man finder sig en bedre editor, eller lærer at snyde sin editor. Mange editorer i dag tilbyder valgmuligheder når man bare kan huske den første del af et makronavn.

Fordelen ved en konsistent gennemskuelig makronavngivning kommer naturligt nok bedst til udtryk i samarbejdet med andre. Her skal man jo sammenflette flere

¹ Der findes også en `*` version af `\newcommand`, men den skal vi ikke komme nærmere ind på her, der er ikke den store forskel.

² Dvs. ordret fremvisning uden L^AT_EX-fortolkning.

A. Om generering af nye makroer

personers unikke måder at skrive på. Det er langt nemmere at enes som brugen af konsistente letforståelige makronavne end det er uvejsomme korte navne (hvor man måske ikke er enige om hvad en specifik makro skal gøre). Jeg har flere gange tilrettet preprints hvor den ene forfatter tydeligvis vidste hvad han/hun gjorde og sørgede for at skrive på en pæn og let tilgængelig måde, hvorimod den anden forfatter skrev mængdenavne ud hver gang, lavede aldrig forkortelser, og ofte de endda ikke altid var enige om den anvendte syntaks i artiklen.

Et godt eksempel på dette kan vi lave med lidt statistik:

Eksempel A.1

```
\newcommand{\DistEq}{\stackrel{\smash{\scriptstyle\mathcal{D}}}{=}}
\l \DistEq \l
```

$$\mathcal{D} =$$

som er lighed i distribution (» \mathcal{D} «-et er naturligvis anderledes alt efter den font man gør brug af).³

Andre eksempler vi allerede har set er `\ArrowBetweenLines`, `\MoveEqLeft`, `\DeclarePairedDelimiter`, *empheq* (fremhæv formel).

Efter sigende skulle pakken `cool` kunne hjælpe med at kunne lave konsistente nemtforståelige makronavne. Men med lidt disciplin kan man sagtens selv lave dem. Det er også en gode ide at skrive en lille kommentar over hver definition, så andre senere får en ide om hvad man har gang i.⁴

Makroer uden argumenter

Langt de fleste af de makroer man selv kommer til at lave bliver nogle som ikke tager i mod argumenter. I [Eksempel A.1](#) ser vi et eksempel på dette. Man skriver altså bare sin erstatningstekst i det andet obligatoriske argument, og denne vil så kunne tilgås via det navn man har givet makroen.

Hvis vi leger videre med statistik eksemplerne, så anvender man tit \sim til at indikere hvordan en stokastisk variabel er fordelt, f.eks. X er standardnormalfordelt:

Eksempel A.2

```
 $X \sim N(0,1)$ 
```

$$X \sim N(0,1)$$

Hvis man ikke kender udseendet af `\sim`, så er koden svær at forstå. Men det hjælper hvis vi laver vores egen makro:

Eksempel A.3

```
\newcommand\DistAs{\sim}
 $\DistAs N(0,1)$ 
```

$$X \sim N(0,1)$$

Dvs. med små midler kan man gøre sin kode meget mere læsbar for sig selv og andre.

Ud over at `\newcommand` til at lave makroer (som ikke tager argumenter), så har vi jo også set eksempler på en konstruktørkommandoer til anvendes til at lave specielle makroer for os. Eksempelvis

`\DeclareMathOperator` som giver »operator« makroer så som `\log`.

Man kan sagtens selv lave konstruktørmakroer, vi så det tilbage i afsnit [5.4.1 på side 128](#), hvor vi lavede en makro til at deklarere farvedesætninger med. Når man roder med den slags makroer har man brug for at kende en del detaljer vedr. L^AT_EX-makroprogrammering, som vi ikke kommer nærmere ind på her. Et eksempel er tilfældet når man laver makroer indeni andre makroer og hvordan dens argumenter så skal håndteres (kendt som #-duplering).

³ `\smash` anvendes for at `\DistEq` ikke forstyrrer linieafstanden når den anvendes i den almindelige tekst.

⁴ Jeg har til tider set folk anvende skabeloner hvor kommentarene næste var direkte løgn!

Makroer med argumenter

Eksemplerne vi har lavet her har mest med matematik at gøre, men man kan naturligvis også lave makroer som kan anvendes i teksten. Her skal vi dog huske dette med at anvender man makroer i teksten som ikke tager noget argument, så har de det med at spise mellemrum. Dette kan man dog også komme udenom, se afsnittet »*Makroer specielt til teksten*« på side 355.

Makroer med argumenter

Med `\newcommand` kan man anvende op til 9 argumenter⁵ som i erstatningskoden kan tilgås via #1 til #9. Et meget hurtigt eksempel kunne være

```
\newcommand\NiceRed[1]{\textcolor{nicered}{#1}}
\NiceRed{Noget i farver}
```

Noget i farver

Eksempel
A.4

hvor vi kan se at »#1« anvendes til at henvise til den tekst man giver til makroen.

Her er et lidt mere kompliceret eksempel med to obligatoriske argumenter. Bemærk at idet erstatningskoden kun kan anvendes indenfor matematik-mode, så kan `\diff` også kun anvendes i matematik.

```
\newcommand{\diff}[2]{\frac{\partial #1}{\partial #2}}
```

Eksempel
A.5

hvor efter `\diff{f}{x}` giver $\frac{\partial f}{\partial x}$. Så kan man jo også lave en version med d i stedet for ∂ eller vi kan være lidt smart. Vi kender jo syntaksen for kubikrødder $\sqrt[3]{2}$ ($\sqrt[3]{2}$) – dvs. hvor man med et ekstra valgfrit argument kan angive graden. Hvordan laver man nu sådan noget?

Et sådant ekstra valgfrit argument kan man også lave via `\newcommand`, dog kun med ét argument og det er altid #1. Syntaksen er

```
\newcommand{\makronavn}[<antal argumenter>][<std værdi for #1>]{<kode>}
```

Hvilket betyder at hvis man ikke anvender noget valgfrit argument vælges *<std værdi for #1>* automatisk og når man giver det valgfri argument anvendes det i stedet. Så en generalisering af `\diff` har nu tre argumenter med koden

```
\newcommand{\diff}[3][\partial]{\frac{#1 #2}{#1 #3}}
\l \diff{f}{x}\diff[d]{x}{t} \quad \quad
\diff{x} \int_0^x f(t)dt = f(x) \l
```

$$\frac{\partial f}{\partial x} \frac{dx}{dt} \quad \frac{\partial}{\partial x} \int_0^x f(t)dt = f(x)$$

Eksempel
A.6

Et tredje eksempel der også er ret anvendeligt (bemærk det specielle kolon):⁶

```
\newcommand\morf[4][\to]{#2\colon#3\#1#4}
$\morf{f}{A}{B}$ eller
$\morf[\hookrightarrow]{f}{A}{B}$
```

$f: A \rightarrow B$ eller $f: A \hookrightarrow B$

Eksempel
A.7

Hvordan man laver makroer som har en stjernet version (så som f.eks. `\section`) eller som har mere end et valgfrit argument (f.eks. `\section` indenfor *memoir*), er ikke noget vi kommer nærmere ind på i denne bog.⁷

⁵ Hvis man har brug for flere, bør man overveje at benytte en *<nøgleord>* = *<værdi>* syntaks i stedet, dette implementeres via pakken *keyval* eller lignende pakker.

⁶ Makroen skyldes Emil Hedevang Lohse.

⁷ For en moderne metode, se *xparse*-pakken.

A. Om generering af nye makroer

Lave om på en allerede eksisterende makro

Man vil nok have opdaget at `\newcommand` tjekker hvorvidt det makronavn man havde tænkt sig, allerede er i brug og den vil selvfølgelig brokke sig hvis den er. Dette er naturligvis for at brugeren ikke uforvarende skal ændre en vigtig intern komponent. Hvis man alligevel gerne vil ændre en allerede eksisterende makro skal man anvende `\renewcommand`. Syntaksen er den samme som for `\newcommand`, men man gør her, via det lille »re«, opmærksom på at man godt er klar over at denne kommando findes i forvejen og ja, man vil gerne ændre den.

Man benytter `\renewcommand` når man vil ændre ting, et eksempel er at der i visse fonte er to udgaver af det græske bogstav epsilon `\epsilon` og `\varepsilon` (ϵ). Hvis man bedst kan lide den sidste, kan man jo bare lave

Eksempel A.8

```
\renewcommand{\epsilon}{\varepsilon}
```



Advarsel A.2. Man skal passe meget meget på når man vælger at overskrive en eksisterende makro. Det er specielt vigtigt at man i forvejen ved præcist hvad originalmakroen gør og at man bestemt ikke får brug for dette.

Det helt grelle eksempel på dette er makroen »`\span`«. Det ville være oplagt at anvende `\span` som navn for span-operatoren fra lineær algebra. Men den er defineret i forvejen og ingen normale L^AT_EX-bøger forklarer hvad `\span` anvendes til.

Dette betyder *ikke* at man så bare skal overskrive den! Makroen `\span` er faktisk en meget vigtig intern komponent i T_EX, som anvendes i tabeller og mange andre interne konstruktioner, så den skal man bestemt ikke overskrive. Anvend i stedet `\Span` som makronavn for span-operatoren (som så bare skal skrive teksten »`\span`«).

Konklusionen er, at ønsker man at anvende `\renewcommand` skal man have en god grund til det og have styr på hvilke konsekvenser det medfører.

Lave en makro hvis den ikke findes i forvejen

Betragt følgende situation: (a) Man er ved at lave sin egen makropakke for nemt at kunne genanvende sine makroer. (b) Man vil ofte (men ikke altid) anvende pakken sammen med en bestemt dokumentklasse. (c) Man anvender tit visse makroer fra denne specifikke klasse og disse makroer ville man gerne også have til rådighed når man anvender sin personlige pakke med andre klasser (eller hvis man havde anvendt denne specielle makro i nogle egendefinerede makroer). Et sådan problem kan løses via en funktionalitet som definerer en makro hvis den ikke findes i forvejen, og hvis den findes, gør den ingenting. Denne makro hedder `\providecommand` og har ellers nøjagtigt den samme syntaks som `\newcommand`. Der findes en tilsvarende makro til at lave environments med. Et eksempel på brugen kunne være de to *memoir*-makroer `\firmlist` og `\tightlist`, disse kunne så inkluderes i ens egen pakke via

Eksempel A.9

```
\providecommand{\firmlist}{%
  \setlength{\itemsep}{0.5\itemsep}\setlength{\parskip}{0.5\parskip}}
\providecommand{\tightlist}{%
  \setlength{\itemsep}{0pt}\setlength{\parskip}{0pt}}
```

Et andet eksempel så vi tidligere (afsnit 10.11 på side 285), hvor vi sørgede for at vi kunne anvende makroen `\texorpdfstring` selv uden hyperref.

Lave om på en makro mens man genbruger den gamle definition

Lave om på en makro mens man genbruger den gamle definition

Når man er ved at lave om på makroer, er det til tider en fordel at kunne gemme den nuværende definition af makroen, lave om på originalen og så f.eks. indsætte den gamle definition igen. I L^AT_EX opnås dette via makroen `\let`.

```
\let\nytnavn=\gammeltnavn
```

`\let`

Et simpelt eksempel på anvendelsen er hvis man ønsker at indbygge `\FloatBarrier` direkte ind i `\section`:

```
\let\oldsection=\section
\renewcommand{\section}{\FloatBarrier\oldsection}
```

Eksempel A.10

Faktisk kan man slippe endnu nemmere om ved dette idet pakken `placeins` (som giver os `\FloatBarrier`) faktisk har en option (`section`) som vil lave ovenstående omdefinering, se desuden [Arseneau \(1999\)](#).

Sidebemærkning A.3. Begynder man på at anvende tricks af denne slags, skal man vide hvad man gør og hvordan tingene fungerer. Dette gælder især hvis der er makroargumenter involveret. Ovenstående omdefinition af `\section` ville eksempelvis gå helt galt hvis man byttede om på rækkefølgen af `\FloatBarrier` og `\oldsection`, thi så ville `\oldsection` spise `\FloatBarrier` som et argument, hvilket tydeligtvis ikke er meningen.



Farlig (læs teksten) A.4. En anden ting man skal være særlig opmærksom på, er at man ikke kommer til at lave en ikke-tiltænkt rekursion. F.eks.

```
\renewcommand\nabla{\overline{\nabla}}
```

Eksempel A.11

Dette vil gå ind i en uendelig rekursion og L^AT_EX vil fejle når den er løbet tør for hukommelse.

Makroer specielt til teksten

Man har sikkert opdaget at undertegnede er ret glad for *memoir*-klassen. For at skrive »*memoir*« kan jeg i min opsætning skrive

```
\markup[nomk,cls]{memoir}
```

Men det bliver man jo træt af, så man laver sig en genvej:

```
\newcommand\memoir{\markup[nomk,cls]{memoir}}
```

Eksempel A.12

Men så er problemet jo at man bliver nødt til at skrive »`\memoir{}`_`_`« hele tiden for ellers spiser den mellemrummet efter. Ved at anvende makroen `\xspace` fra pakken `xspace`, kan man komme ud over dette problem, dvs.

`\xspace`
`xspace`

```
\usepackage{xspace}
\newcommand\memoir{memoir\xspace}
```

```
bla \memoir bla
```

bla memoir bla

Eksempel A.13



Via `\xspace` vil makroen man har lavet ikke spise mellemrummet efter.

A.2 Hvor gælder ændringer af makroer, længder etc.?

De fleste programmeringssprog har et begreb kaldet *scope* som beskriver hvor en ændring af en variabel etc. er gældende, ofte global ændring versus lokal ændring. Man kunne oversætte scope med *spillerum*.

I \TeX er ændringer normalt globale i den forstand at de virker alle steder fra deres definition og frem. Ændringerne er dog lokale hvis omstændighederne er i orden. (Visse ændringer er dog *altid* globale.)

Definition A.5. Ændringer eller definitioner er lokale hvis

- de er omkranset af et `{...}`-par. Eksempel

Eksempel
A.14

```
Almindelig tekst
{\tiny tekst}.
Almindelig tekst igen.
```

```
Almindelig tekst tekst. Almindelig tekst
igen.
```

Dette inkluderer dog *ikke* det `{}`-par som omkranser argumentet til en makro. Her bliver man nødt til at tilføje et ekstra par. Eksempel:

Eksempel
A.15

```
\newcommand\lilletekst[1]{\small #1}
```

- omkranset af parret `\begingroup... \endgroup`, som fungerer på nogenlunde samme måde som et `{}`-par.
- ændringen sker indenfor et environment (environment definitionen inkluderer normalt et `\begingroup–\endgroup`-par, dette indsættes automatisk af `\newenvironment`)

```
\begin{<navn>}
ændringer af fonte, fontstørrelser, omdefinering af makroer, ændring af længder
etc. som sker herinde har ingen effekt efter \end{<navn>}, de vil efter slutningen
tilbage til den værdi de havde før environmentet.
\end{<navn>}
```

Personligt anvender jeg tit `\begingroup... \endgroup` konstruktionen til lokalt at ændre på ting. Dette kan f.eks. være lokalt at pille ved `\descriptionlabel` som står for formateringen af labels i *description*-environmentet. Ville man lokalt gøre disse labels kursive kunne man anvende

Eksempel
A.16

```
\begingroup
\renewcommand\descriptionlabel[1]{%
\hspace{\labelsep}\textit{#1}}
\begin{description}\tightlist
\item[foo] bar
\item[hest] nisse
\end{description}
\endgroup
```

```
foo bar
hest nisse
```

Efter `\endgroup` er definitionen af `\descriptionlabel` tilbage til hvad den var ved `\begingroup`.

Se iøvrigt [Eksempel A.19 på næste side](#).

A.3 Lave sit eget environment

Med `\newenvironment` laver man åbenlyst nye environments. Syntaksen og muligheden for argumenter er nøjagtigt den samme som for `\newcommand`, dog skal man her udfylde to kode segmenter.

```
\newenvironment{<env navn>}[<antal argumenter>][<std #1>]%
  {<start kode>}{<slut kode>}
```

`\newenvironment`

Her er dog en vigtig lille twist, eventuelle argumenter kan kun anvendes i *start koden*, ikke i *slut koden*.⁸ Eksempel:

```
\newenvironment{myquote}[1][\itshape]{%
  \begin{quotation}#1{\end{quotation}}
```

**Eksempel
A.17**

som nu er en variation af *quotation* som automatisk sætter al teksten i *italic* (eller hvad man nu ønsker).

Skal man ændre et eksisterende environment benyttes selvfølgelig `\renewenvironment` med de sædvanlige forholdsregler. Her bør man dog være opmærksom på at der er visse environments som man ikke kan ændre eller anvende på denne måde, hvilke skal vi dog ikke komme ind på her.

Her er et eksempel på hvordan man kan kombinere `babel` med *quote*-environmentet for at lave et environment til engelske citater.

```
\newenvironment{engquote}%
  {\begin{otherlanguage}{english}%
   \begin{quote}\itshape\small}%
  {\end{quote}\end{otherlanguage}}%
\chaptername % dansk
\begin{engquote}
  \chaptername % engelsk
\end{engquote}
\chaptername % dansk igen
```

Kapitel <i>Chapter</i> Kapitel
--

**Eksempel
A.18**



Bemærk at det ikke kun er inden i brugen af et environment at man har lokale ændringer, det har man også når man definerer sit eget environment. Man kunne vælge at pille ved ting internt i environmentet, eller definere makroer som kun er defineret indenfor environmentet, men ikke udenfor.⁹ (Her har vi faktisk et eksempel på den førnævnte #-duplering.)

```
\newenvironment{foobar}{%
  \renewcommand\emph[1]{\textcolor{nicered}{##1}}
  \newcommand\MyLittlePony{Nej, vel!}
}{%
\begin{foobar}
  \emph{\MyLittlePony}
\end{foobar}}
```

Nej, vel!

**Eksempel
A.19**

⁸ Dette er også noget man kan komme ud over. I den moderne løsning med pakken `xparse`, så vil argumenterne til environmentet være tilgængelig i både start- og slutdel af environmentkoden.

⁹ Dette kan være ret nyttigt i visse sammenhænge.

A.4 Makronavne med @

Et af de områder hvor ændring af eksisterende makroer volder de største problemer er brugen af »@« i makronavne.

Makronavne med »@« er normalt *interne* makroer, altså noget som pakke- og klasseforfattere anvender for at lave de makroer som slutbrugeren har til rådighed. Eksempelvis er \@tempdima ofte være anvendelig som en midlertidig længdevariabel.

Under normale omstændigheder må brugeren kun anvende bogstaver i makronavne. Forsøger man i preamblen (eller andet sted i ens dokument) at ændre ved en makro med »@« i navnet uden at tage sine forholdsregler vil man opleve en fejlmeddelelse lignende:

```
! LaTeX Error: Missing \begin{document}.
```

```
See the LaTeX manual or LaTeX Companion for explanation.
```

```
Type H <return> for immediate help.
```

```
...
```

```
l.3 \newcommand\foo@b
```

```
ar{}
```

Ønsker man at ændre på en makro med »@« i navnet skal man derfor, overfor L^AT_EX, gøre opmærksom på at man altså godt ved at det man gør kan være potentielt farligt. Dette gør man ved at pakke udtrykket ind på følgende vis

\makeatletter
\makeatother

**Eksempel
A.20**

```
\makeatletter
\newcommand\foo@bar{...}
\makeatother
```

\makeatletter fortæller så L^AT_EX at herfra skal den opfatte »@« (engl. »at«) som et bogstav, og så må den gerne anvendes i makronavne (sammen med a–Z og A–Z). Tilsvarende vil \makeatother lave »@« om til noget *andet* igen. Se også [FAQ \(2007\)](#).

Sidebemærkning A.6. Hvis man i stedet har flyttet ændringen af @-makroer til ens egen pakke, så behøver man ikke \makeatletter–\makeatother-parret.

A.5 Om at lave sin egen pakke

Har ens dokument en meget lang preamble eller er der nogle konstruktioner man ofte benytter, kan man med fordel lave sin egen pakke. Dette er faktisk meget nemt. Her vil vi kort beskrive fremgangsmåden:

- (a) Find et navn (e.g. minpakke), og åbn filen minpakke.sty (endelsen er vigtig) i en teksteditor.
- (b) Start filen med

```
\ProvidesPackage{<pakkenavn>}[YYYY/MM/DD tekst]
```

Eksempel:

**Eksempel
A.21**

```
\ProvidesPackage{minpakke}[2005/07/18 preamble til mit speciale]
```

Den dato man anvender kan faktisk anvendes når man loader pakken, hvis man gerne vil sikre sig at man får en version som er nyere end en given dato.

Makroer med avancerede argumenter

- (c) Så flytter man bare alle ens definitioner og egne makroer til denne pakke som så i dokumentet kaldes via `\usepackage{minpakke}`. Pakken placeres et sted hvor \LaTeX kan findes den, oftest i det samme bibliotek som dokumentet, eller i ens lokale pakke-træ. Se dokumentationen for din egen \LaTeX installation for mere information om hvordan man installerer pakker.
- (d) Husk at man i ens *egen* pakke gerne må benytte »@« i makronavne og man kan uden videre ændre ved makroer hvis navn indeholder et »@«, så nu har man ikke længere brug for `\makeatletter` eller `\makeatother`.

\LaTeX 3 (1999) kan være til hjælp når man vil starte med at lave sin egen pakke, her kan man f.eks. finde metoder til at load e f.eks. en konfigurationsfil (hvis den findes), eller hvordan man definerer pakkeoptions.



Ender man op med en pakke som andre kunne få glæde af kunne man overveje at uploade den til CTAN. Her skal man lige huske at give pakken en licens, normalt lppl-licensen (<http://www.latex-project.org/lppl.txt>), som bl.a. siger at andre gerne må rette ved pakken, men det må ikke frigive den under det samme navn. Dette gør man for at sikre at filer ser ens ud selv om de bliver oversat på forskellige systemer.

A.6 Makroer med avancerede argumenter

Her er der to typer: $\langle key \rangle = \langle value \rangle$ -systemer samt makroer med mange valgfri argumenter eller stjerne-makroer.

Til den første type kan man tage et kig på pakken `xkeyval`. Vi vil her tage et kig på en løsning til den anden type.

Pakken vi vil anvende er faktisk en del af det som en gang måske bliver til \LaTeX 3: `xparse`. Vi har allerede set den i brug i afsnit 5.4.1 på side 128 til at lave de farvede sætninger.

I den lidt mere skumle ende anvendes makroen fra [Eksempel A.22 på næste side](#) til at give en line i listen over græske bogstaver (tabel 3.2). For at gøre tingene endnu nemmere kombinere vi kode med lidt kode fra `etoolbox`. Ideen er at hvis bogstavet er et symbol, så vil vi gerne indeksere det, i.e. vi vil indtaste det uden `\`, mens hvis det bare er et latinsk bogstav, så skal den bare kunne skrives. Vi vil også gerne kunne tilføje en note til outputtet (i form af en tabelnote label). Eksempler på brugen af makroen kan ses i slutningen af [Eksempel A.22 på den følgende side](#). Ret smart, og meget pladsbesparende i koden.

A. Om generering af nye makroer

Eksempel A.22

```
\DeclareDocumentCommand\graeskebogstaver{ mso mso m}{%
  \ifblank{#1}{%
    &
  }{\IfBooleanTF#2{%
    \texttt{#1} & $#1$%
    \IfNoValueTF{#3}{\tnotex{#3}}%
  }{%
    \markup[msym,nomk]{#1} & $\csuse{#1}$%
    \IfNoValueTF{#3}{\tnotex{#3}}%
  }%
}%
&&
\ifblank{#4}{%
  &%
}{%
  \IfBooleanTF#5{%
    \texttt{#4} & $#4$%
    \IfNoValueTF{#6}{\tnotex{#6}}%
  }{%
    \markup[msym,nomk]{#4} & $\csuse{#4}$%
    \IfNoValueTF{#6}{\tnotex{#6}}%
  }%
}%
&& #7%
\\
}
%
brug:
%\graeskebogstaver{navn}*[note-label]{navn}*[note-label]{udtale}
\graeskebogstaver{alpha}{A}*{alfa}
\graeskebogstaver{sigma}{Sigma}[tn:4]{sigma}
```

A.7 Andre ting værd at kende vedr. makroprogrammering

En pakke jeg personligt har haft megen glæde af på det seneste er etoolbox som bl.a. biblatex er bygget over. Pakken giver en masse spændende redskaber for programmører med kendskab til \TeX . Dog er der også relevante ting som almindelige $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -brugere kunne få brug for. Antag f.eks. vi har lavet en makro som tager et valgfrit argument i #1. Hvordan tjekker vi om dette argument er tomt?

Eksempel A.23

```
\usepackage{etoolbox}
\newcommand\test[2][]{%
  \ifblank{#1}{\#1 er tom}{\#1 er ikke tom}
}
```

Tag et kig i [Lehman \(2009\)](#) for mere information, eller print den ud, de ting den bidrager med er *meget* nyttige.

Her er et andet eksempel baseret på et problem en studerende kom med. Vi vil gerne kunne skrive f.eks. DNS-sekvenser ind i teksten, men på en form hvor vil kan styre udseendet af dem i preamblen. Vi vil gøre dette ved at lave en data struktur som minder lidt om den måde hvorpå referencer virker. Vi laver to makroer, en til at definere en sekvens og tildele den en nøgle, samt en til at bruge en sekvens. Koden er som følger.

Andre ting værd at kende vedr. makroprogrammering

```
\usepackage{etoolbox}
\newcommand\DefineSequence[2]{%
  \ifcsdef{DNA-#1}{%
    \GenericError{DNA Seq.}{Sequence key '#1' already defined}{}%
  }{%
    \csdef{DNA-#1}{#2}%
  }}
\newcommand\DNA[1]{%
  \ifcsdef{DNA-#1}{\csuse{DNA-#1}}{%
    \GenericError{DNA Seq.}{Sequence key '#1' unknown}{}%
  }}
}
```

Eksempel

A.24



Det vi gør er at gemme data i makronavne som indeholder nøglenavnet. Vi tester så efter om det findes i forvejen, og definerer kun en ny nøgle hvis den ikke allerede findes. I det andet argument til `\DefineSequence` kan man så anvende den formatering man nu har lyst til. Skal de alle skrives som sans-serif, så kan man jo bygge dette ind i selve makroen. En brug ville så være (meget bizar DNA-sekvens)

```
\DefineSequence{hest}{\textsf{HeST}}
\DNA{hest}
```

Eksempel

A.25

Tip A.7. Hvis der af en eller anden grund skal indgå »_« som en del af det formaterede navn, så man kan f.eks. anvende

```
{\urlstyle{rm}\path{#2}}
```

Eksempel

A.26

Så er man fri for at skulle skrive »_« som »_«.

Det overlades til læseren at lave en `\reDefineSequence`-makro til at ændre på en eksisterende makro. (Tip: `\csdef` tjekker ikke selv om makroen den definerer findes eller ej.)

Der findes også makroer til at tjekke om bestemte pakker er indlæst, samt test for om de er kaldt med bestemte options.

Men generelt, hvis man ønsker at forstå \LaTeX -makroprogrammering, så er der ingen vej uden om at kende til noget \TeX -programmering. Så er der en god ide at kigge i [Eijkhout \(2001\)](#) eller [Knuth \(1986\)](#) (hvis man har den). Desuden er \LaTeX -kildekoden et godt sted at kigge efter hvordan ting er lavet, eller tjekke syntaksen for noget man mener at kunne huske. I moderne \LaTeX -installationer kan man få fremvist \LaTeX -kildekoden via

```
texdoc source2e
```


Nyttige metoder og konstruktioner

I indledningen skrev vi at vi ikke ville komme nærmere ind på hvordan man laver avanceret makroprogrammering i L^AT_EX. Vi vil dog forklare lidt om nogle redskaber som kan være nyttige.

B.1 Om længder

I L^AT_EX er en *længdekommando* en speciel type af makro, hvor værdien skal være et tal med en længdeenhed, understøttede enheder kan ses i Tabel B.1. Som man kan se understøttes enheden *meter* ikke.

Tabel B.1: Længdeenheder i L^AT_EX

pt	punkt
pc	pica, (12 pt)
in	amerikansk tomme (1 in er lig 72.27 pt)
bp	big points (72 bp er lig 1 in) ^a
mm	millimeter
cm	centimeter (2.54 cm er lig 1 in)
dd	didot (1157 dd er lig 1238) ^b
cc	cicero (12 dd) ^b
em	dynamisk længde svarende til den nuværende skriftstørrelse ^c
ex	ændres også dynamisk med skriftstørrelsen, svarer ca. til højden af »x«.

^a trim-optionen til `\includegraphics` (se afsnit 6.1.1 på side 137) anvender denne enhed hvis man undlader at angive en enhed.

^b minder om pt samt pc og anvendes især i Europa.

^c dvs. skriver man noget med `\small`, så ændres størrelsen af 1 em indenfor dette område. Som fingerregel siger man ofte at en *em* svarer cirka til bredden af »M«.

Til at oprette nye længdemakroer eller ændre deres værdi, anvender man en af følgende makoer:

<code>\newlength{\langlenavn\rangle}</code>	<code>\newlength</code>
<code>\setlength{\langlenavn\rangle}{\langleværdi\rangle}</code>	<code>\setlength</code>
<code>\addtolength{\langlenavn\rangle}{\langleværdi\rangle}</code>	<code>\addtolength</code>
<code>\settowidth{\langlenavn\rangle}{\langlemateriale\rangle}</code>	<code>\settowidth</code>
<code>\settoheight{\langlenavn\rangle}{\langlemateriale\rangle}</code>	<code>\settoheight</code>
<code>\settodepth{\langlenavn\rangle}{\langlemateriale\rangle}</code>	<code>\settodepth</code>

De tre første makroer er til at oprette en ny længde makro, til at sætte en længde til en bestemt værdi og den tredje til at addere til en given længde (\pm). Bemærk at der *skal* enhed med, men også at den valgte enhed er ligegyldig. De tre sidste makroer kan anvendes til at måle på forskellige størrelser vedr. det *⟨materiale⟩* man har angivet. Bemærk at højden er højden over basislinien (den linie teksten står på) og at dybden henviser til dybden under (derfor er der to makroer).

B. Nyttige metoder og konstruktioner

`calc` Det er en god ide altid at anvende pakken `calc` som udvider funktionaliteten af bl.a. `\setlength` og `\addtolength` således at man nu kan lave beregninger indeni $\langle værdi \rangle$. Det vil sige at har man indlæst `calc` så er følgende lovligt:

Eksempel B.1

```
\newlength{\mylength}
\setlength{\mylength}{400pt+5cm}
\setlength{\mylength}{\textwidth-15mm}
\addtolength{\mylength}{400pt*6}
```

Bemærk at man gerne må blande enheder (dog er $4\text{cm} + 5$ forbudt), samt at skal man gange et »tal med enhed« med et helt tal, så skal $*6$ komme *efter* tallet med enhed. Bemærk også at man altid gerne må skrive f.eks.

Eksempel B.2

```
\addtolength{\mylength}{0.6\textwidth}
```

når der er tale om en given længdemakro. Ønsker man at gange en længde med et reelt tal (decimaltal) så skal man anvende syntaksen » $400\text{px} * \text{real}\{1.7\}$ «, se [Thorup og Jensen \(1998\)](#) for yderligere information.

`Calc` pakken giver os også nogle ekstra måle makroer:

```
\widthof      \widthof{\langle materiale \rangle}
\heightof    \heightof{\langle materiale \rangle}
\depthof     \depthof{\langle materiale \rangle}
```

Svarer til makroerne beskrevet tidligere, men her kan de med det samme anvendes som en længde angivelse, uden at man skal bruge en eventuel midlertidig længdemakro til at gemme værdien i.

```
\totalheightof  \totalheightof{\langle materiale \rangle}
\settotalheightof \settotalheightof{\langle navn \rangle}{\langle materiale \rangle}
```

Hvilket svarer til i et hug at fange højden og dybden af noget $\langle materiale \rangle$.

En anden fordel ved brugen af `calc` er at man så kan anvende længde-addition alle de steder som internt anvender `\setlength`, dette tæller f.eks. `geometry`-pakken, `memoirs` margin justeringer, `\parbox` og `minipage`-environmentet (som vi beskriver senere i dette kapitel) samt f.eks. søjlespecifikationen »p« til tabeller.

Tip B.1 (Skrive værdien af en længde ud i teksten).

Sæt »`\the`« foran, f.eks. er liniebredden lige nu

Eksempel B.3

```
\the\linewidth
```

197.525pt

Hvis man ikke lige har begreb om størrelsen af pt i hovedet, så kan man anvende pakken `printlen` til at få længden vist frem i en mere forståelig enhed.

Eksempel B.4



```
\usepackage{printlen}
-----slut preamble-----
\uselengthunit{mm}
\printlength{\textwidth}
```

69.42128 mm

Nyttige længder

B.1.1 Nyttige længder

Her lister vi nogle af de længder som anvendes i L^AT_EX, der kan være nyttige at kende, f.eks. er det meget anvendeligt at kunne skalere et billede i termer af liniebredden i stedet for en fast værdi i centimeter, for så skalerer det fint hvis man senere ændrer tekst-/liniebredden, se Tabel B.2.

Tabel B.2: Nyttige længder i L^AT_EX

<code>\textwidth</code>	tekstbredden i dokumentet
<code>\textheight</code>	teksthøjden i dokumentet
<code>\paperwidth</code>	bredden af den valgte papirtype
<code>\paperheight</code>	højden af den valgte papirtype
<code>\linewidth</code>	den nuværende liniebredde, ændrer sig dynamisk. ^a
<code>\unitlength</code>	dette er en længde som normalt anvendes som enhed i <i>picture</i> -environmentet (som ellers ikke omtales i denne bog, se dog appendix i seneste version af Wilson (2010)). ^b
<code>\columnsep</code>	halvdelen af afstanden mellem søjlerne i en tabel.

^a `\linewidth` er f.eks. kortere end tekstbredden hvis man er i et punkt i en liste eller indeni `\parbox{5cm}{...}`

^b Det skulle ikke være noget problem at anvende den til egne formål. De fem foregående længder bør man *ikke* pille ved i dokumentet.

Man har desuden tre temporære længde makroer man kan bruge: `\@tempdima`, `\@tempdimb` og `\@tempdimc`. Disse bør alle kun anvendes lokalt og hurtigt (da der er andre makroer som også internt anvender disse makroer og derfor ændrer på værdien).

B.2 Om tællere

En tæller (engl. *counter*) er en anden type makroer i L^AT_EX som kan holde hele tal (og kun hele tal). Der er så markoer til at manipulere disse.

Egentlig kender vi allerede til tællere, der er f.eks. en tæller (`page`) som holder det nuværende sidetal, en anden (`chapter`) som holder styr på det nuværende kapitelnummer.

En speciel ting ved tællere er at de faktisk består af to makroer: En som er selve tælleren, samt en som er en grafisk repræsentation af tælleren. Den sidste kommando er den makro man så anvender for at skrive værdien af en tæller i selve dokumentet. Det kunne jo være at tælleren skulle skrives med romertal, så definerer man bare den grafiske version af tælleren til at konvertere det hele tal gemt i tælleren til romertal.

Makroerne vi anvender er:

```
\newcounter{<navn>}[<navn på dominereden tæller>] \newcounter
\setcounter{<navn>}{<værdi>} \setcounter
\addtocounter{<navn>}{<værdi>} \addtocounter
```

hvor $\langle værdi \rangle \in \mathbb{Z}$. Har man indlæst `calc`-pakken så må man lave udregninger i $\langle værdi \rangle$. Har man angivet en $\langle dominerende tæller \rangle$, så vil den tæller vi opretter blive nulstillet hver gang den dominerende tæller ændrer værdi.

Til at tælle værdien op med +1 har vi to *step* makroer:

```
\stepcounter{<tæller>} \stepcounter
\refstepcounter{<tæller>} \refstepcounter
```

B. Nyttige metoder og konstruktioner

Her vil `\refstepcounter` desuden gøre at man kan henvide til den nye værdi via `\label`.

Skal man bruge den nuværende værdi af en tæller (f.eks. i en sammenligning, se senere) så kan man anvende

```
\value      \value{<tæller>}
```

Dette kan så bruges til f.eks. at sætte værdien af `<tæller2>` til at være værdien af `<tæller1>`:

Eksempel B.5

```
\setcounter{counter2}{\value{counter1}}
```

B.2.1 Den grafiske repræsentation af en tæller

Selve tælleren vil *altid* holde et helt tal. Den grafiske repræsentation af tælleren klares af makroen `\the<navn>`. Denne kan man så omdefinere som man lyster. Til at hive værdien ud i forskellige formater har vi makroerne i Tabel B.3.

Tabel B.3: Hjælpe makroer til at skrive værdien af en tæller.

<code>\arabic{<navn>}</code>	Almindelige tal
<code>\roman{<navn>}</code>	Små romertal ^a
<code>\Roman{<navn>}</code>	Store romertal ^a
<code>\alph{<navn>}</code>	Små bogstaver ^b
<code>\Alph{<navn>}</code>	Store bogstaver ^b
<code>\fnsymbol{<navn>}</code>	Symboler

^a Understøtter kun værdier som er større end nul.

^b Kun bogstaverne »a« til »z« inkl. »w«.

Så hvis man har en tæller ved navn »hest« og man ønsker at denne skal skrives med store romertal så laver man bare

Eksempel B.6

```
\renewcommand\thehest{\Roman{hest}}
```

Hvis tælleren er domineret af en anden tæller og man gerne vil vise dette (på samme måde som f.eks. sektionnumrene i denne bog) så skal man tage dette med når man laver sin `\the<navn>`-makro. F.eks. er `\thesection` ofte defineret som

Eksempel B.7

```
\renewcommand\thesection{\thechapter.\arabic{section}}
```

Har man en tæller som allerede er defineret, og man gerne vil have den domineret, er der et par konstruktioner man kan anvende, se Tabel B.4.

Tabel B.4: S/M for tællere.

<code>\numberwithin{<tæller>}{<dominatrix>}^a</code>
<code>\counterwithin{<tæller>}{<dominatrix>}^b</code>
<code>\counterwithout{<tæller>}{<dominatrix>}^{b,c}</code>

^a Kræver *amsmath*-pakken.

^b Kræver *memoir*-klassen eller *chngcntr*-pakken.

^c Denne vil åbenlyst fjerne domineringen igen.

B.2.2 Om at ramme tællerværdier i errata eller præsentationer

En bruger havde fornyligt brug for at kunne gengive en korrekt version af en given tabel i et errata dokument til en bog. Her er det svar jeg gav ham.

```
\newcounter{extra}
\renewcommand\thetable{\arabic{extra}.\arabic{table}}
... % og lige før hans tabel
\setcounter{extra}{12}
\setcounter{table}{1}
```

Eksempel

B.8



For så ville hans næste tabel få nummeret »12.2«, hvilket var det han ville.

B.3 Betingedekonstruktioner (if-then)

T_EX har sine egne lav-tekniske *hvis*-konstruktioner, som ligger noget uden for denne bog, den ivrige læser kan tage et kig i [Knuth \(1986\)](#) eller [Eijkhout \(2001\)](#). Til specielle (store) konstruktioner anbefales det at man gør sig bekendt med `\ifdim` og `\ifnum`.



Vi vil i stedet anbefale en mere brugervenlig konstruktion. Der er forskellige løsninger, f.eks. `xifthen`-pakken. Vi vil dog i stedet præsentere en anden, nyere løsning – `etoolbox`-pakken. Den kan det samme bare på en lidt anden måde. Det skal dog nævnes at nogle af de ting som `etoolbox` kan lægger meget op til at man også ved noget om koncepterne bag T_EX-programmering, men de fleste vil nu sagtens kunne anvende de betingede konstruktioner alligevel. Vi vil ikke gå så meget i detaljer, men vil i stedet henvise til [Lehman \(2009\)](#).

B.3.1 Sammenligninger

```
\ifnumequal{<tal udtryk>}{<tal udtryk>}{<kode ved sandt>}{<kode ved falsk>}
```

```
\ifnumgreater{<tal udtryk>}{<tal udtryk>}{<kode ved sandt>}{<kode ved falsk>}
```

```
\ifnumless{<tal udtryk>}{<tal udtryk>}{<kode ved sandt>}{<kode ved falsk>}
```

Disse tre svarer til at der er =, > eller < mellem tal udtrykkene. Et tal udtryk kan i denne forbindelse være f.eks. »3*(15-8)« eller hvis man vil sammenligne kapitelnummeret med noget: »\value{chapter}«

```
\ifnumodd{<tal udtryk>}{<kode ved sandt>}{<kode ved falsk>}
```

Det er vist ret åbenlyst hvad denne gør.

```
\ifdimequal{<længde udtryk>}{<længde udtryk>}{<kode ved sandt>}{<kode ved falsk>}
```

```
\ifdimgreater{<længde udtryk>}{<længde udtryk>}{<kode ved sandt>}{<kode ved falsk>}
```

```
\ifdimless{<længde udtryk>}{<længde udtryk>}{<kode ved sandt>}{<kode ved falsk>}
```

De tre tilsvarende konstruktioner for længder. Man må gerne anvende notationen svarende til »(\textwidth - \mylength)/2«.

B. Nyttige metoder og konstruktioner

B.3.2 Boolske udtryk

Her henvises til sammenligninger under *ét*, det kan være sammenligninger vedr. variable som kan være sandt eller falsk, det kan være on/off variable (som ligner meget sandt/falsk bare implementeret på en anden måde), og det kan være blandinger med andre »if« konstruktioner.

Vi vil bare forklare nogle få, samt fortælle hvordan man laver en løkke.

```
\newtoggle{\langlenavn\rangle}
```

```
\toggletrue{\langlenavn\rangle}
```

```
\togglefalse{\langlenavn\rangle}
```

```
\iftoggle{\langlenavn\rangle}{\langlekode for sandt\rangle}{\langlekode for falsk\rangle}
```

Disse laver en on/off variabel og den andre makroer sætter værdien eller tjekker den. Standardværdien efter den er oprettet er »false«. Der findes også en sætter makro hvor man selv skal give den værdien true|false. Skal man have noget til at være sandt globalt set kan man skrive \global foran \toggletrue.

Der er et lignende system til 'rigtige' »if« konstruktioner som de L^AT_EX selv kan lave. I *memoir*-klassen har vi f.eks. en »if« konstruktion som kan fortælle os om vi er i mainmatter eller ej. Dette kunne man også have lavet med en toggle.

```
\whileboolexpr{\langleudtryk\rangle}{\langlekode\rangle}
```

Dette er faktisk løkke konstruktionen. Man skal passe meget på med at sørge for at løkken faktisk stopper igen¹ for eller stopper L^AT_EX, aldrig.

Der kan stå mange forskellige ting i *⟨udtryk⟩*, løkken stopper når den får et udtryk som er falsk. Dette kunne f.eks. være en toggle, i så fald ville man skrive

Eksempel B.9

```
\whileboolexpr{ toggl{navn} }{ kode hvor navn bliver falsk på et tidspunkt }
```

Her er et andet simpelt eksempel, som vil gennemløbe tallene fra 1 til 10

Eksempel B.10



```
\newcounter{mincctr} % default er nul
\whileboolexpr{ test{
\ifnumless{\value{mincctr}}{10} } }{%
% vigtig, ellers stopper vi aldrig
\stepcounter{mincctr}
\themincctr % udskriv
}
```

1 2 3 4 5 6 7 8 9 10

```
\ifboolexpr{\langleudtryk\rangle}{\langlekode hvis sandt\rangle}{\langlekode hvis falsk\rangle}
```

Dette er så den helt generelle blanding if konstruktion. I *⟨udtryk⟩* kan man blande længetests, toggles, negeringer, *and* og *or* osv. Læs venligst selv detaljerne i [Lehman \(2009\)](#).

¹ En fejl jeg ofte har lavet.

B.3.3 Andet fra etoolbox

Ud over en masse meget nyttige redskaber til at teste for ting (f.eks. `\ifblank`) eller til at oprette dynamisk definerede makroer (til f.eks. at holde data), så har jeg selv haft glæde af et liste koncept. Man kan se dette i brug i indekset.

Indgangene i indekset er lavet meget specielt, hver især udfører faktisk op til flere makroer. Dette anvendes til at alle indgangene på en side skriver til en liste hvad deres bogstavsgruppe er. På hver side tilføjes så den lodrette liste, og for hvert bogstav spørger vi om bogstavet findes i sidens liste. Hvis den gør markeres den, og hvis ikke skrives den normalt. Herefter tømmes listen, så den er klar til en ny liste.

I en anden sammenhæng har jeg anvendt listerne således at hver indgang i listen f.eks. havde formatet »`{...}{...}{...}`«. I etoolbox har man en gennemløbningsmekanisme, som man skal give en makro, listen gennemløbes så og makroen udføres på hver indgang i listen en for en. Dette kan så i dette tilfælde anvendes til at få de tre værdier ud af selve indgangen. Vi anvendte det til at lave en løsning af at indsamle (og senere formatere) forfatternavne og deres affiliations på en mere konfigurerbar måde end `\authblk` giver os adgang til.

B.4 Luftkasteller

Til specialdokumenter samt til forsider har man brug for at kunne indsætte ekstra luft vertikalt og horisontalt. Dette vil vi kigge lidt nærmere på i dette afsnit.

B.4.1 Lodret

Der er tre makroer man bør kende til

<code>\vspace{⟨længde⟩}</code>	<code>\vspace</code>
<code>\vspace*{⟨længde⟩}</code>	<code>\vspace*</code>
<code>\vfill</code>	<code>\vfill</code>

Disse skal helt placeres *mellem afsnit*, ellers får man unintuitive resultater. Argumentet kan være en længdemakro, eller et tal med en enhed, se evt. Tabel B.1 på side 363.

Forskellen mellem `\vspace` og `\vspace*` ligger i at hvis `\vspace` kommer som det første på en side, så ignoreres den, mens `\vspace*` ikke ignoreres. Hvorfor denne forskel? Fordi, hvis man f.eks. tager `\section` så laver den jo noget luft over for at adskille den fra teksten som kommer lige før. Men hvis der er et sideskift lige før overskriften er der jo ikke nogen grund til at anvende luft til at lave denne adskillelse.

`\vspace*{⟨længde⟩}` indsættes uanset hvad. Vi anvender ofte `\vspace*` til forsider, for enten lige at presse den øverste del lidt ned, eller hvis `⟨længde⟩` er negativ, for at hive indholdet lidt op ind i området over tekstblokken (anbefales *kun* til forsider!).

Makroen `\vfill` er lidt anderledes, den vil rykke *alt* ned mod bunden af siden, dog vil den ignoreres hvis den placeres først på siden, hvilket kan afhjælpes med f.eks. `\strut\vfill`.

Man kan desuden anvende specielle elastiklængder og -angivelser til at skabe andre nyttige effekter, se afsnit B.4.3 på næste side. Hvis f.eks. man ønsker at centrere noget vertikalt på på en side, måske til et skilt eller en forside, så er det faktisk pænere at dette står lidt over den faktiske midten af siden.

B.4.2 Vandret

Til de vandrette har vi tilsvarende

B. Nyttige metoder og konstruktioner

```
\hspace      \hspace{<længde>}
\hspace*    \hspace*{<længde>}
\hfill      \hfill
```

Her er det dog i starten af en linie at `\hspace` eller `\hfill` kan forsvinde. Disse tre makroer anvendes mest til forsider samt i floats hvori flere figurer står ved siden af hinanden. Eksempel

Eksempel B.11

```
tekst\hspace{1cm}tekst\hfill tekst
```

tekst	tekst	tekst
-------	-------	-------

B.4.3 Elastiske konstruktioner

Man har måske lagt mærke til at mellemrummet omkring `\section` kan variere fra side til side alt efter indholdet af siden. Man siger at dette mellemrum indeholder en vis mængde stræk og krymp (engl. *stretch* og *shrink*). Man kan selv specificere sådan noget via `\setlength`, syntaksen er så

```
\setlength{<navn>}{<længde plus stræklængde minus krympelængde>}
```

Man skal bemærke de to nøgleord »plus« og »minus«, og husk at man behøver ikke have begge med. Denne specifikation gør så at hvis længden `\<navn>` bliver anvendt i en lodretsammenhæng, og hvis denne side har brug noget strækning eller sammentrækning, så vil de angivne længder tages med i betragtning. Bemærk at konstruktionen kan også anvendes i argumentet til f.eks. `\vspace`.

Mine personlige ynglingseksempler til dette er indholdsfortegnelsen samt indekset. I indholdsfortegnelsen indættes der luft før hvert nyt kapitel, hvis man ændrer den længde som indsættes således at den nu indeholder stræk og krymp, så opnår man ofte en pænere indholdsfortegnelse med bedre sideombrydninger, eller en ToC som er blevet trukket lidt mere sammen således at den nu kan være på én side. I indekset er det afstanden mellem bogstavgrupperne man kan pille ved på denne måde.



Der er også en anden elastikkonstruktion som kan være meget nyttig. Makroen

```
\stretch{<helt tal ≥ 0>}
```

kan anvendes som erstatning til `<længde>`

Præcist hvordan den skal fortolkes er lidt svært at forklare, så det undlader vi. Vi laver i stedet et eksempel. Man skal forestille sig at vi ønsker at lave en forside hvor der bare står en titel og at denne skal stå lidt over midten på siden.

Eksempel B.12



```
\fbox{\begin{minipage}%
      [c][0.8\textheight][c]{\linewidth}
\strut\vspace{\stretch{3}}

\centering
En titel

\vspace{\stretch{5}}\strut
\end{minipage}}
```



For at kunne vise resultatet som et af vore små eksempler bliver vi lige nødt til at smide det hele ind i min *minipage*-konstruktion² og smide endnu en ramme omkring.

² Som forklares i afsnit B.5 på modstående side.

Kassetækning

Det man skal observere er at `\vspace{\stretch{3}}` indsætter et elastisk mellemrum på samme måde som `\vfill`, men her der er en form for samarbejde mellem de to `\stretch`-makroer således at de deler luften mellem sig efter den nøgle som er givet.

B.5 Kassetækning

Vi har allerede mødt L^AT_EXnikske kasser i f.eks. afsnit 6.3 på side 147, her vil vi forklare syntaksen lidt nærmere.

B.5.1 Kasser med linieombrydning

Vi har to primære konstruktioner (som egentlig er det samme),

```
\parbox[⟨pos⟩]{⟨bredde⟩}{⟨indhold⟩}
\begin{minipage}[⟨pos⟩]{⟨bredde⟩}
⟨indhold⟩
\end{minipage}
```

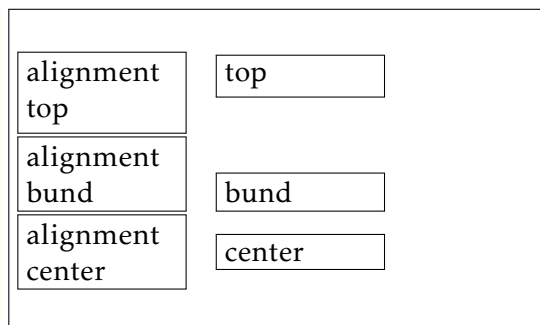
`\parbox`
`minipage`

Eneste forskel mellem de er at den ene anvendes som et environment mens den anden bare er en makro. `⟨pos⟩` er en af »t«, »b« eller »c« (standard), anvendes i og for sig kun hvis en `\parbox`/*minipage* står ved siden af en `\parbox`/*minipage* eller f.eks. en *tabular*. Herefter er tolkningen at hvis de to har angivet [t] så vil den første basislinie i hver kasse komme til at stå overfor hinanden. Tilsvarende i bunden hvis man har angivet [b]. Ved [c] centrerer indholdes vertikalt. Lad os illustrere med et eksempel, se [Eksempel B.13](#).

```
\fbox{\parbox[t]{2cm}{%
alignment top}}\quad
\fbox{\parbox[t]{2cm}{top}}

\fbox{\parbox[b]{2cm}{%
alignment bund}}\quad
\fbox{\parbox[b]{2cm}{bund}}

\fbox{\parbox[c]{2cm}{%
alignment center}}\quad
\fbox{\parbox[c]{2cm}{center}}
```



Eksempel
B.13

Man skal være opmærksom på at indholdet inden i en `\parbox`/*minipage* brydes *ikke* hen over en side, det er bare en kasse med en anden bredde end den normale tekst. Vi anvender dem primært til forsider (tekster på hver sin side af forsiden justeret i toppen eller bunden, her kunne man også anvende en smart tabel) eller til opsætning af flere figurer i samme float, se afsnit 6.3 på side 147.

B. Nyttige metoder og konstruktioner

Advanceret B.2. Makroen `\parbox` og environmentet *minipage* kan faktisk tage endnu flere argumenter. Man kan nemlig også sætte højden af en af disse kasser. Den komplette liste af argumenter er

```
\parbox[⟨pos⟩][⟨højde⟩][⟨indre position⟩]{⟨bredde⟩}{⟨indhold⟩}
\begin{minipage}[⟨pos⟩][⟨højde⟩][⟨indre position⟩]{⟨bredde⟩}
⟨indhold⟩
\end{minipage}
```

Hvor *⟨indre position⟩* forklares i [Mittelbach og Goossens \(2004\)](#).

Dette med at kunne sætte højden af f.eks. en *minipage* kan være ret nyttigt i egne meget specielle makroer. På Inst. for Matematiske Fag har vi en webapplikation som kan lave lister som de studerende skal registrere sig på til de teoretiske øvelser. Her styrer vi feltstørrelsen via `\parbox`.



Der findes et par pakker som kan være nyttige.

varwidth Giver environmentet

varwidth

```
\begin{varwidth}{⟨max bredde⟩}
⟨indhold⟩
\end{varwidth}
```

som minder om *minipage*, men som faktisk justerer sin størrelse således at den passer med den maksimale bredde af *⟨indhold⟩*.

boxedminipage Giver environmentet

boxedminipage

```
\begin{boxedminipage}{⟨max bredde⟩}
⟨indhold⟩
\end{boxedminipage}
```

Som igen svarer til en *minipage* uden valgfri options. Her sættes så desuden en ramme omkring indholdet.

B.5.2 Inline kasser (uden linie ombrydning)

Som vi kan se skal `\parbox`/*minipage* have en bredde og at disse har linieombrydning. Vi har også kassekonstruktioner som *ikke* har ombrydning, men som stadigvæk kan være interessante.

```
\mbox      \mbox{⟨indhold⟩}
\fbbox     \fbbox{⟨indhold⟩}
\makebox   \makebox[⟨bredde⟩][⟨pos⟩]{⟨indhold⟩}
\framebox  \framebox[⟨bredde⟩][⟨pos⟩]{⟨indhold⟩}
```

Makroerne `\fbbox` hhv. `\framebox` sætter en ramme omkring deres indhold (tykkelse: `\fbboxrule`; polstring: `\fbboxsep`).

Tip B.3. Har man brug for at sætte en ramme omkring noget som skal være flere linier, så pak indholdet ind i en `\parbox`/*minipage*-konstruktion.

Et par makroer fra `graphicx`-pakken

De to makroer som kan tage $\langle bredde \rangle$ har et sjovt trick i ærmet. Inden i $[\langle bredde \rangle]$ kan man anvende makroerne `\width`, `\height`, `\depth`, og `\totalheight` til at tilgå størrelserne af $\langle indhold \rangle$.

$[\langle pos \rangle]$ er »l«, »r« eller »c« (standard) og anvendes oftest til at specificere alignment inden i kassen. Man kan lave sjove effekter ved at anvende bredden »0pt«:

```
tekst\hspace{2cm}
\makebox[0pt][r]{Hopla!\ }tekst

tekst\hspace{2cm} uden makebox
```

tekst	Hopla!	tekst
tekst		uden makebox

**Eksempel
B.14**

Vi laver en kasse uden bredde og justerer tingene således at højre kant af $\langle indhold \rangle$ havner det sted hvor `\makebox` er placeret. Observer hvordan teksten skrives indeni området vi laver med `\hspace`.

Makroen `\mbox` anvendes primært til at smide omkring noget man ikke ønsker skal deles over to linier.



Vi kan vertikalt flytte placeringen af en via

```
\raisebox{\langle løft \rangle}[\langle højde \rangle][\langle dybe \rangle]{\langle indhold \rangle}
```

`\raisebox`

hvor $[\langle højde \rangle]$ $[\langle dybe \rangle]$ påvirker den opfattede størrelse af det man har løftet. Tit sætter man disse til »0pt«, hvis de ikke er sat, anvendes de naturlige størrelser ud fra $\langle indhold \rangle$. Et eksempel på brugen af `\raisebox` kan ses i [Eksempel 6.23 på side 149](#). Bemærk at vi også her kan anvende `\width`, `\height`, `\depth`, og `\totalheight` til at henvise til den naturlige størrelse af indholdet.

B.6 Et par makroer fra `graphicx`-pakken

Vi har tidligere set at `\includegraphics`-makroen kan skalere og rotere sit indhold. Disse features fåes også som enkeltstående makroer.

```
\scalebox{\langle vandret skalering \rangle}[\langle lodret skalering \rangle]{\langle indhold \rangle}
\resizebox{\langle vandret bredde \rangle}{\langle lodret højde \rangle}{\langle indhold \rangle}
\reflectbox{\langle indhold \rangle}
\rotatebox[\langle options \rangle]{\langle vinkel \rangle}{\langle indhold \rangle}
```

`\scalebox`

`\resizebox`

`\reflectbox`

`\rotatebox`

Den første makro anvender decimaltal (faktor) til at skalere indholdet med, hvis $\langle lodret skalering \rangle$ ikke er angivet, anvendes samme faktor her. Hvis faktoren er -1 vil $\langle indhold \rangle$ blive spejlvendt). Omvendt kan man med `\resizebox` skalere til en bestemt bredde eller højde. Skulle man ønske at skalere til en hvis bredde og gerne vil have at f.eks. højden skaleres med samme resulterende faktisk s kan man anvende et »!« som magiske længde. Et sted hvor jeg har fundet god nytte af `\resizebox` er i et dokumenter hvor vi skulle lave en navneliste og hver der også skulle være email-adresser og disse måtte ikke linie ombrydes. Dette kan man fikse via

B. Nyttige metoder og konstruktioner

Eksempel B.15



```
\usepackage{etoolbox,url,graphicx,calc}
```

— slut preamble —

```
\newcommand\printemail[1]{%
  \begin{group}%
  \settowidth\unitlength{\url{#1}}%
  \ifdimgreater\unitlength{2cm}%
  {%
    \resizebox{2cm}{!}{\url{#1}}%
  }%
  \url{#1}%
}%
\end{group}%
}
\printemail{daleif@imf.au.dk}

\printemail{dr@dr.dk}
```

```
daleif@imf.au.dk
dr@dr.dk
```

Dvs. hvis den er for stor, skaleres den automatisk ned.

[*options*] til `\rotatebox` skal vi ikke komme nærmere ind på her, de har noget at gøre med hvilket punkt man roterer om samt hvilke enheder vinklen skal angives i (normalt er det i grader). Se i stedet [Goossens et al. \(2007\)](#) eller [Reckdahl \(2005\)](#).

B.7 Gemme/undersøge indhold

L^AT_EX kan gemme noget *indhold* i en intern kasse, som så kan hentes frem, måles på og sættes ind igen. De relevante makroer er

```
\newsavebox    \newsavebox{\<boksnavn>}
\savebox       \savebox{\<boksnavn>}[<bredde>][<pos>]{<indhold>}
\usebox        \usebox{\<boksnavn>}
\lrbox         \begin{lrbox}{\<boksnavn>}
               <indhold>
               \end{lrbox}
```

Man skal huske på at når (hvis overhovedet) man igen ønsker at anvende indholdet igen, så *skal* man anvende `\usebox` for at få fat på den.

Som eksempel, lad os se på hvordan man selv kan implementere noget der ligner *boxedminipage*.

Gemme/undersøge indhold

```
\usepackage{calc}
----- slut preamble -----
\newsavebox{\mybox}
\newenvironment{myboxedminipage}[1][\linewidth]{%
  \begin{lrbox}{\mybox}%
    \begin{minipage}{#1-2\fbboxsep-2\fbboxrule}
      \raggedright
    }{%
      \end{minipage}%
    \end{lrbox}%
    \fbbox{\usebox{\mybox}}}
\begin{myboxedminipage}[3cm]
  Noget tekst inden i en kasse af bredden 3cm minus polstringen af
  rammen omkring.
\end{myboxedminipage}
```

Eksempel

B.16



Noget tekst inden
i en kasse af
bredden 3cm
minus
polstringen af
rammen
omkring.

Det er faktisk noget lignende dette vi anvender for at kunne fremvise eksemplerne i denne bog. Bare på en lidt mere ‘bøvlet’ måde.

Tip B.4. Hvis man leger med gemte kasser og skal bruge højden, dybden (under basislinien) eller bredden af den, så kan man anvende syntaksen

```
\ht\boksnavn
\dp\boksnavn
\wd\boksnavn
```

F.eks. er `\ht\mybox+\dp\mybox` lig den totale højde af `\mybox`.

Sidebemærkning B.5 (Gemme kasse globalt).

Man kan gemme en kasse globalt med f.eks. `\global\sbox`. Dette kan *ikke* gøres med `lrbox`-environmentet. Man kan dog snyde via

```
\global\sbox{\mybox}{\usebox{\mybox}}
```

Det trick anvender vi f.eks. når vi laver eksempler som står ved siden af hinanden, således at vi kan vide hvor store de er og således at kassen og den farvede baggrund kan få den samme størrelse.

Eksempel

B.17

L^AT_EX-distributioner samt andre redskaber

Denne bog koncentrerer sig om hvordan man *bruger* L^AT_EX, ikke så meget om hvor man får L^AT_EX fra eller hvilke redskaber man bør bruge (eller i det mindste kende til).

I dette kapitel vil vi skrive noget om dette emne, vi vil ikke gå i dybe detaljer med alt, men i stedet give et par hints for at styre nye brugere i den rigtige retning.

Der skal ikke herske nogen tvivl om at vi herfra anbefaler brugen af T_EX Live på alle platforme. Så har vi alle de samme redskaber og de virker på samme måde (via T_EX Live til Linux og Windows, samt MacTeX til Mac OS X).

C.1 L^AT_EX-distributioner

Der er for så vidt kun to reelle L^AT_EX-distributioner: T_EX Live og MikT_EX, hvor af den sidste kun er tilgængelig til Windows.

C.1.1 T_EX Live

Dette er den primære distribution til Linux og Mac, men den kører naturligvis lige så godt på Windows. Man skal være opmærksom på at T_EX Live er langt mere restriktiv vedr. licenser på pakker og fonte end MikT_EX er. T_EX Live kræver frie licenser uden specielle krav, derfor kan man finde visse pakker på CTAN som ikke er til at finde i T_EX Live. Man vil også til tider se at opdateringen af T_EX Live vil afinstallere pakker, dette skyldes primært inkompatible licenser.

I modsætning til MikT_EX, opdateres T_EX Live dagligt (hvis der er noget at opdatere). Det kan dog tage et par dage før opdateringerne når ud til de såkaldte *mirrors*, dvs. decentrale websites rundt om i verden hvorfra man så henter sine opdateringer.

T_EX Live på Mac

Dette kommer normalt i form af MacTeX samlingen, se <http://tug.org/mactex>, som er et T_EX Live snapshot plus et par ekstra programmer. Efter at have hentet MacTeX.mpkg.zip, og pakket den ud, skulle den være ganske nemt at installere for en almindelig Mac bruger.¹

¹ Det har det i hvert fald været det gange jeg har gjort det, og jeg er bestemt *ikke* en regelmæssig Mac-bruger.

Tip C.1 (T_EX Live-opdatering på Mac).

Da der er tale om et snapshot (i skrivenden stund fra juli 2011), er det ret vigtigt at man sørger for at opdatere L^AT_EX-distributionen efter man har installeret MacTeX. Der er to måder at gøre dette:

- (a) Anvend »T_EX Live Utility«, burde ligge i **Applikationer** → **TeX** eller noget i den retning. Dette er et alternativt interface til T_EX Live's egen pakkemanager, således at det mere føles som et Mac program. Eller man kan
- (b) anvende T_EX Live's egen pakkemanager. Har man Perl/Tk installeret kan det startes grafisk fra kommandolinien via »sudo tlmgr -gui« (»sudo« fordi MacTeX installeres som rodbrugeren). Vælg herefter »Update all installed«, dette vil opdatere det man har, og installere de nye ting som er kommet siden sidst.

Har man *ikke* Perl/Tk, kan man køre den kommandolinie via

```
sudo tlmgr update --all --self
```

Bemærk at den nogle gange vil bede om selv at blive opdateret, her fortæller den så selv hvilke options man skal kalde den med.

Ud over T_EX Live og et par ekstra programmer (Se TeX-folderen), installerer MacTeX desuden editorerne TeXShop og TeXworks.²

T_EX Live på Windows

Den nemmeste måde at få T_EX Live er via installation over nettet. På en god internetforbindelse tager det ca. en times tid (ofte hurtigere). Selve netinstallationen foregår ved at installationsprogrammet henter ca. 2300+ filer med pakker og programmer, også installerer disse. Har man en meget emsig anti-virus, kan installationen derfor blive ret langsom.

T_EX Live installeres over nettet ved at man går til

<http://tug.org/texlive/acquire-netinstall.html>

hent `install-tl.zip` og pak den ud,³ dette laver et bibliotek med et navnet »install-tl-⟨dato⟩«, gå ned i det og kørs »install-tl.bat«. ⁴ Herefter svarer man »Next« indtil den giver en knap som siger »Install«, start installationen og lad den køre til den er færdig.

Ud over T_EX Live og venner installeres desuden editoren TeXworks.

² Den sidste er en forholdsvis ny editor, bygget over TeXShop, men nu skrevet således at den er tilgængelige på alle de store platforme.

³ Til Windows-brugere kan det anbefales at installere programmet 7-zip, se <http://www.7-zip.org/>, det er gratis, og langt mere stabilt end Windows' eget program til udpakning af ZIP-filer. Desuden håndterer 7-zip langt mere end bare ZIP-filer.

⁴ Som standard viser Windows ikke filendelser i Windows Explorer (WE), men det anbefales at omjustere WE til netop at vise endelser, så forveksler man ikke tingene.

Sidebemærkning C.2 (TeXworks 0.2).

Indtil slutningen af 28. marts 2011, kom TeX Live med TeXworks 0.2, som var konfigureret en smule uhensigtsmæssigt. Nu er TeXworks 0.4 i stedet være kommet med, og den ser ud til at være konfigureret en del bedre. Derfor er det en god ide at få tingene opdateret.

Et af det nye ting som er kommet med er muligheden for at oversætte med programmet latexmk i stedet for pdflatex. Programmet anvender fortsat pdflatex til at oversætte med, men sørger automatisk for at oversætte alle de gange det er nødvendigt for at holde krydsreferencer og citeringer opdateret (inkl. kørsler af bibtex og makeindex eller lignende).⁵

Tip C.3 (texdoc).

Har man installeret TeX Live 2011 så kan man få hurtig hjælp til en pakke ved gennemkommandoen »Kør« at udføre f.eks. »texdoc memoir«, så får man manualen til *memoir* (Wilson, 2010) frem i Adobe Reader. På Vista/7 kan man bare skrive kommandoen i søgefeltet i start-menuen.

Texdoc-kommandoen findes også på MikTeX, men på TeX Live har man gjort det endnu bedre. På MikTeX vil man ikke få *memoir*-manualen via »texdoc memoir«, her skal man vide, at den manual faktisk hedder »memman.pdf« i virkeligheden. I TeX Live vedligeholdes et større arkiv af disse aliaser.

Tip C.4 (Få TeX Live til at kigge i ekstra folder).

Nogle gange kan det være en fordel at fortælle L^AT_EX at den også lige skal tjekke for pakker i *denne* specielle folder.

Lad os antage at man ligger tingene med basis i E:/minefiler/.⁶

Man skal lave en såkaldt TDS-struktur i en folder, dvs. man skal placere sine filer i E:/minefiler/tex/latex/minpakke/. Stien »/tex/latex/« er en del af TDS-strukturen, hvilket er en speciel organisering af filer så L^AT_EX nemmere kan holde styr på hvad der er hvad.

Vi skal så fortælle TeX Live at den også skal kigge i E:/minefiler/:

```
tlmgr conf texmf TEXMFHOME E:/minefiler//;
```

Det er *meget* vigtigt med »//;« til slut. Udtrykket ovenfor skal udføres i en kommandoprompt.



Opdatering er nem: **Start** → **TeX Live 2011** → **TeX Live manager**. Bed om at køre »Update all installed«, hvilket vil opdatere pakkerne, og installere nye hvis der er nogen.

Bemærk at manageren en gang i mellem skal opdateres, dette vil den selv fortælle, og forklare hvad man skal gøre.

Sidebemærkning C.5 (Tid til opdatering).

Opdater ca. en gang hver eller hver anden måned. Der er en del, også populære, pakker som opdateres ofte (f.eks. hyperref eller siunitx).

⁵ Ud over min editor er latexmk det program jeg anvender mest i forbindelse med L^AT_EX.

⁶ På andre styresystemer, skrives folder placeringer anderledes, men ellers er metoden den samme.

Sidebemærkning C.6 (T_EX Live 2011 på Windows).

T_EX Live 2011 udkom i august 2011. Desværre er der på Windows ikke mulighed for at opgradere fra version 2010 til 2011. Man bliver nødt til at geninstallere.

T_EX Live på Linux

Gennem de fleste Linux-distributioner kan man få en eller anden version af T_EX Live. Desværre er man af nogle af disse Linuxer meget langt bagud i deres opdateringer, især Ubuntu kan være slet slem.⁷ Da Ubuntu er den mest udbredte Linux blandt studerende, vil vi kun behandle den. Bruger du en anden Linux-distribution, kan det være at man har mere held med sig vedr. T_EX Live-versionen. Dog skal man kigge langt efter T_EX Live-opdateringsværktøjet »t_lmgr«, er der kommet opdaterede pakker, må man pænt vente på at Linux-vedligeholderne opdaterer pakkerne i selve Linux-distributionen.

Sidebemærkning C.7 (Ubuntu problem 1).

Ubuntu opdaterer ikke deres T_EX Live installation når TUG udsender opdateringer. Så man kommer hurtigt til at mangle pakker med nye features. I Ubuntu 10.10 har vi f.eks. T_EX Live 2009, og ingen mulighed til at opdatere til T_EX Live 2011.

Der er to endnu værre ting (efter min mening): (a) Man har fjernet t_lmgr, i.e. man kan ikke selv opdatere pakker, og (b) Ubuntu splitter T_EX Live op i en hulens masse Ubuntu pakker, så man selv kan finde ud af hvad man har brug for. Jo, tak de fleste ender med at installere alt (den hedder vist nok »texlive-all«) fordi de ikke kan finde rundt i det rod.

»Ok, fint, så installerer vi da bare TUGs T_EX Live via net-installering som vi gjorde på Windows.«

Det er faktisk det vi anbefaler at gøre, men her kommer så problem nummer to.

Sidebemærkning C.8 (Ubuntu problem 2).

Når man installerer software på Ubuntu, så har man et koncept kaldet »recommended«. Dvs. ud over det software man gerne ville have, så vil Ubuntu også lige installere dette som du 'sikkert' får brug for.

Dette er meget fint, hvis ikke det lige var fordi Ubuntu så vil installere Ubuntu's T_EX Live hvis man f.eks. beder om at få installeret en L^AT_EX-editor som TeXworks eller TeXmaker.

»Hvad gør vi så?«

Har man pladsen til det, er det nemmeste faktisk at have »to« T_EX Live installationer. En gennem Ubuntu selv, således at *dependencies/recommended* for editorer er opfyldt, samt TUGs T_EX Live installation inkl. T_EX Live-manageren således at vi *selv* kan bestemme hvornår der skal opdateres pakker!⁸

Har man *ikke* pladsen, så skal man enten slå *recommended software* fra mens man installerer en editor, eller man skal bygge en *dummy*-pakke til Ubuntu som siger til Ubuntu at vi har installeret noget som er ækvivalent med en liste af Ubuntu-pakker.

⁷ Vist noget med at gammel kode er stabil kode. Noget jeg ikke er helt enig i.

⁸ Linux er ligeglad med at et program findes i flere varianter på maskinen, den har en liste af steder hvor den leder, og den første som findes, vælges.

Sidebemærkning C.9 (Installering af editor udenom recommended).

På Ubuntu kan man fra kommandolinien installere f.eks. editoren TeXMaker *uden* at installere *recommended* via

```
sudo apt-get install --no-install-recommends texmaker
```

Sidebemærkning C.10 (Ækvivalent i Ubuntu).

En anden løsning ligger i noget Ubuntu kalder »*equivs*«, altså at man installerer noget som får Ubuntu til at tro at noget ækvivalent til et bestemt produkt er blevet installeret. Nedenstående er hentet fra <http://www.tug.org/texlive/debian.html>, næremre bestemt afsnittet »*Vanilla TeX Live on Debian*«.

Kort fortalt er det følgende der skal laves

- (a)

```
%> sudo apt-get install equivs
```

, dette vil også installere en del ekstra Ubuntu ting.
- (b)

```
%> equivs-control texlive-local
```

, den fil som dette genererer skal vi have overskrevet med en vi henter på nettet
- (c)

```
%> wget http://www.tug.org/texlive/debian-control-ex.txt
%> cp debian-control-ex.txt texlive-local
```
- (d) Overvej at rette i »*texlive-local*«, så »2009« erstattes med »2011«.
- (e) Nu skal vi bygge en Ubuntu pakke med

```
%> equivs-build texlive-local
```
- (f) Pakken skal så installeres

```
%> sudo dpkg -i texlive-local_2011-1~1_all.deb
```



Lad os nu installere TUGs T_EX Live på Ubuntu og sørg for at denne vil blive valgt selv om Ubuntu selv skulle finde på at installere deres version af T_EX Live. Strategien er at installere T_EX Live et alternativt sted, og så justere den såkaldte søgesti således at vores version altid kommer først.⁹

- (1) Først vil vi lave et sted vi kan installere T_EX Live. Her går vi ud fra at maskinen er en personlig maskine og at der kun er én bruger. Har man flere brugere, kan det være en ide at lave en dedikeret bruger til at installere T_EX Live under. Vi vælger at installere i `/opt/texlive`, så den skal vi lige have oprettet. Desuden vil vi gerne installere som os selv, ikke som rod-brugeren.¹⁰
- (2) Udfør nu nedenstående

```
%> cd /opt
%> sudo mkdir texlive
%> sudo chown luser:luser texlive
%> ls -ld /opt/texlive
```

hvor »*luser*« naturligtvis skal være ens eget brugernavn. Den sidste kommando, skulle gerne få et output lignende

⁹ Ting som Ubuntu installerer bliver placeret meget bestemte steder, som allerede er i søgestien, der er derfor ikke den store fare for at søgestien overskrives af en opdatering.

¹⁰ Det er bare en personlig præference.

C. L^AT_EX-distributioner samt andre redskaber

```
drwxr-xr-x 8 user user 4096 2011-10-26 16:45 /opt/texlive
```

som viser at brugeren og gruppen »user« ejer /opt/texlive.

- (3) Start med at downloade netinstalleren, fra

<http://tug.org/texlive/acquire-netinstall.html>

vi skal bruge `install-tl-unx.tar.gz`.

- (4) Læg den et sted hvor du kan finde den, og pak den ud. Er man, som jeg, kommando-linie-rytter, kan man anvende

```
%> tar zxvf install-tl-unx.tar.gz
```

- (5) Gå ned i det bibliotek som udpakningen lavede.
(6) Nu er det så en fordel at man sørger for at der er et par ting som er installeret i ens Ubuntu installation. Kør følgende i en kommandoprompt (eller oversæt selv til Synaptics)

```
%> sudo apt-get install perl-tk perl-doc
```

Dette gør at vi kan køre det grafiske installationsprogram, som er lidt mindre skæmmende end at bruge kommando-linie-installationen, og for at »man tlmgr« formateres korrekt.

- (7) I en kommandoprompt startes det grafiske installationsprogram

```
%> ./install-tl -gui
```

Her vil man se at den brokker sig over at »TEXDIR« ikke er skrivebart. Ændr det til »/opt/texlive/2011«, og start installationen. Lad den køre færdig.

- (8) Slutteligt vil vi pille ved søgestien. Men først vil vi gøre det lidt lettere for os selv til når T_EX Live 2011, kommer.

```
%> cd /opt/texlive
%> ln -s 2011 current
```

Så skal vi kun flytte »current« (det er en pegepind) for at skifte L^AT_EX-installation. Vi skal lige kende din Linux arkitektur

```
%> ls -l /opt/texlive/current/bin
```

På min maskine får jeg

```
i386-linux
```

dvs. jeg skal tilføje »/opt/texlive/current/bin/i386-linux« til min søgesti.

```
%> sudo cp /etc/environment /etc/environment.backup
%> sudo gedit /etc/environment
```

I denne fil finder man noget lignende »PATH="/usr/local/sbin:/usr/local/bin:... "«, ret dette til

```
PATH="/opt/texlive/current/bin/i386-linux:/usr/local/sbin:... "
```

ergo jeg smider »/opt/texlive/current/bin/i386-linux:« ind i starten af den værdi som »PATH« sættes til. Bemærk kolonen! Det er *meget* vigtigt.

- (9) Herefter skal man lige logge ud og ind igen for at »/etc/environment« bliver eksekveret, og så skulle man have installeret TUGs T_EX Live.

Man kan teste via

Andet vedr. T_EX Live

```
%> which latex
```

Dette skulle gerne give noget lignende »/opt/texlive/current/bin/i386-linux/latex«



Opdatering af T_EX Live på linux sker faktisk med det samme program som på Mac og Windows: tlmgr. Man kan gøre det grafisk via

```
%> tlmgr -gui
```

og så klikke sig frem (»Update all installed«), eller man kan lave det hele fra kommandolinien (det gør jeg)

```
%> tlmgr update --all --self
```

Sidebemærkning C.11 (Opgradering fra T_EX Live 2010 til T_EX Live 2011).

Det er på Linux muligt at *opgradere* fra T_EX Live 2010 til T_EX Live 2011. Det er kræver en smule manuelt arbejde, men det fungerer fint. Metoden er beskrevet på <http://www.tug.org/texlive/upgrade.html>. Så behøver man ikke installere alt igen fra bunden, man får bare de ændringer der er kommet til siden man sidst opdaterede.

Andet vedr. T_EX Live

Det er visse fonte som ikke kan inkluderes i T_EX Live pga. den licens de er udgivet under. Især den såkaldte »no-sell« licens.¹¹ Men ofte må man *gerne* anvende fontene som almindelig bruger (også til at lave materiale som senere skal sælges).

Da fonte kan være ret svære at installere i L^AT_EX¹² er der nogen som har lavet et hjælpende redskab til os. Gå til

<http://www.tug.org/fonts/getnonfreefonts/>

og hent `install-getnonfreefonts`. Dette er en såkaldt Lua-fil, og den skal oversættes med

```
texlua install-getnonfreefonts
```

Dette vil så installere to nye programmer `getnonfreefonts-sys` og `getnonfreefonts`.¹³ De ekstra fonte som dette program giver adgang til kan så installeres via f.eks.¹⁴

```
getnonfreefonts-sys -a
```

Giver man i stedet argumentet »-l« så vil den vise de fonte som den kan hente, og argumentet »-h« viser alle de argumenter som programmet understøtter.

Blandt de understøttede fonte er en Garamond variant, samt LuxiMono.



¹¹ Der bliver lavet DVD'er med T_EX Live og venner som bl.a. sælges af boghandlen Lehmanns.

¹² Fonthåndtering i X_eL^AT_EX skulle være noget nemmere, men det er en anden historie.

¹³ Man kan diskutere hvorfor udviklerne ikke har inkluderet dette program i T_EX Live? Men det skyldes nok at man heller ikke har inkluderet supportfilerne for andre fontpakker, hvor selve fonten ikke kan inkluderes i T_EX Live.

¹⁴ Har man mulighed for det, så anvend `-sys`-versionen, MAC og evt. Linux brugere skal huske `sudo` foran.

Tip C.12 (Få skrevet filnavn foran linie med fejl).

Når man arbejder på store projekter i flere filer, kan det være svært, udfra .log-filen, at se hvilken fil en given fejl ligger i. I T_EX Live kan vi tilføje en konfiguration som gør at det aktuelle filnavn skrives foran en given fejlmeddelelse.

Man skal finde filen `texmf.cnf` (der er flere af versioner af denne fil, så pas på), den findes i den samme folder hvor man finder folderen `texmf-dist/`. I denne fil tilføjer man linierne

Kodetip
C.1


```
% add file names to errors  
file_line_error_style = t
```

C.1.2 MikT_EX

Man kan umiddelbart få MikT_EX på to forskellige måder, enten installere det via nettet, eller få det som et samlet snapshot sammen med nogle andre redskaber. Lad os tage den sidste først.

ProTeXt

ProTeXt indeholder en kopi af MikT_EX (taget på en eller anden given dato), sammen med editoren TeXnicCenter og værktøjerne GhostScript og GSView. Det hele er nemt at installere via en speciel PDF-fil (kræver sikkert Adobe Reader).

Selve proTeXt-filen kan downloades fra <http://tug.org/protext>. Bemærk venligst de øvrige kommentarer vedr. MikT_EX som kommer herunder.

Ulempe. I og med proTeXt er et 'snapshot' af MikT_EX, betyder det at der sikkert er kommet opdateringer til MikT_EX siden dette snapshot er lavet. Alt efter hvor lang tid dette er siden, kan mængden af opdateringer nemt betyde at det ville være mere fordelagtigt at installere MikT_EX direkte fra nettet, og efterfølgende selv installere f.eks. TeXnicCenter.

ProTeXt er desuden inkluderet på T_EX Collection DVDen som udgives af TUG, og som kan fås gennem den lokale T_EX-brugergruppe.¹⁵

Ren MikT_EX

Man får MikT_EX 2.9 fra <http://miktex.org/2.9/setup>, det anbefales (på det kraftigste) at man installerer den fulde MikT_EX (via *MikT_EX Net installer*), ikke bare *basic*. Man henter via såkaldte *mirrors*, dvs. steder i verden som har stillet båndbredde til rådighed. Et godt dansk mirror er via <http://dotsrc.org>.¹⁶

¹⁵ I alle tilfælde er det en god ide at støtte sin lokale T_EX-brugergruppe.

¹⁶ Endnu bedre hvis man kan hente fra et dansk universitet, da <http://dotsrc.org> er koblet på forskningsnettet – dejlig hurtigt.

Opdatering af MikTeX

Sidebemærkning C.13 (MikTeX automatisk pakke installation).

MikTeX indeholder (som den eneste) et system hvor hvis ens preamble kalder en pakke man ikke har installeret, så vil MikTeX gå på nettet og hente den. Nyttig feature, men den virker bare ikke altid optimalt. F.eks. fortæller den ikke brugeren hvor langt den er nået med installationen (træls hvis det den installerer tager lang tid), eller også fejler den totalt uden af brugeren opdager det (velkendt problem hvis man anvender editoren TeXnicCenter).

Da rigtigt mange af de pakker vi omtaler i denne bog, ikke er inkluderet i MikTeX basic (f.eks. *memoir* og *mathtools*), anbefaler vi at man *altid* installerer *alt*. Det kan godt være at man ikke kommer til at anvende mere end 10%, men man mangler heller ikke noget fra starten.

Sidebemærkning C.14 (Sprogopsætning).

I tidligere versioner af MikTeX skulle man selv sørge for at dansk orddeling blev inkluderet i L^AT_EX. I MikTeX 2.9 har man heldigvis sørget for at alle sprog er aktiveret som standard (hvilket det har været i T_EX Live i mange år).

Sidebemærkning C.15 (Ikke alle værktøjer i MikTeX virker med det samme).

Det skal bemærkes at der er visse redskaber som kommer med i en MikTeX installation som *ikke* kan køre uden videre. Her har man ofte brug for manuelt at installere et specifikt programmeringssprog på sin Windows, f.eks. Perl, Python eller Ruby. Relevante eksempler til Perl kunne være `latexmk`¹⁷ eller `makeglossaries`.

Anvender man i stedet T_EX Live, så virker disse med det samme da T_EX Live kommer (på Windows) med en (skjult) Perl-installation.

Opdatering af MikTeX

Start → **Programmer** → **MikTeX 2.9** → **Update**. Dette skal man til tider køre to gange: Første gang opdateres programmerne (f.eks. selve opdateringsprogrammet), anden gang henter den de opdaterede pakker.

Man bør sørge for at opdatere en gang i mellem, f.eks. en gang om måneden, eller hvis der er kommet en speciel opdatering man gerne vil have med.

Bemærk at for MikTeX, opdateres pakkerne ca. en gang om ugen, hvorimod pakkerne i T_EX Live opdateres dagligt (alt efter mirror kan da bagefter dage en dag eller to før et mirror også har en specifik opdatering).

Sidebemærkning C.16 (MikTeX settings – ekstra rødder).

MikTeX har et godt konfigureringsværktøj – »MikTeX settings« som man finder i MikTeX menuen. Her finder man bl.a. et interface til at bede MikTeX om at kigge ekstra steder efter filer. Meget anvendeligt til egne pakker, og et noget nemmere interface end det tilsvarende for T_EX Live.

¹⁷ En af mine personlige favoritter, den anvender jeg dagligt.

C.2 Editorer

I dette afsnit vil vi skrive lidt om nogle af de editorer man kan anvende til at skrive sin L^AT_EX kode. Vi kan naturligvis ikke beskrive alle, men man skulle gerne få et indtryk af hvad man kan bruge. Rækkefølgen er tilfældig. Hvis der ikke står noget ekstra tekst ved en editor, er det fordi undertegnede ikke personligt kender meget til editoren. Har man noget relevant information eller et godt tip til en editor er man meget velkommen til at sende en mail.

Hvordan vælger man en god editor? Det er faktisk et rigtigt godt spørgsmål. Det kommer an på hvilke behov man har. Nogle har f.eks. glæde af knapper med adgang til symboler, andre (som jeg selv) sætter større pris på kraftfulde features til editering af tekst. Helt tredje personer har andre krav til deres editor.

TeXworks

Hjemmeside: <http://tug.org/texworks>

Encoding: Alle relevante

Pris: Fri software

Platform: Linux, Mac og Windows

Som allerede nævnt kommer TeXworks allerede med TeXLive til Mac eller Windows, og følger med MikT_EX til Windows, alternativt kan man hente den fra den nævnte hjemmeside.

På Linux skal man hente TeXworks via sin Linux distribution.¹⁸ På Ubuntu er dette nemt via (eller søg efter den i Synaptics)

```
%> sudo apt-get install texworks
```

Features. TeXworks er en meget simpel editor, uden mange avancerede features. Der er en stor grøn knap til at oversætte med og et par knapper til at gemme med o.lign. Editoren vil normalt fylde halvdelen af skærmen (venstre side), den anden halvdel er forbeholdt TeXworks' indbyggede PDF-fremviser. Fremviseren og editoren er kædet sammen således at man kan hoppe fra PDF til den relevante linie i koden, og omvendt. Der er desuden en ganske fornuftig syntaksfarvning, som man anbefales at anvende.

TeXworks har også en indbygget stavekontrol, bygget på det velfungerende hunspell bibliotek. Det er nemt at tilføje ekstra sprog.¹⁹

Editoren understøtter desuden såkaldte magiske kommentarer, dvs. kommentarer som editoren vil lede efter og så herefter justere sig selv efter de værdier den finder. For nærmere information kan følgende blog-indlæg anbefales: <http://www.texdev.net/2011/03/24/texworks-magic-comments/>. Her forklarer Joseph Wright bl.a. hvordan man kan angive *master* for denne fil, eller hvilket sprog stavekontrollen skal anvende.

Fordele. (a) Nemt at gå til uden at blive distraheret af alt for mange knapper og features. (b) Indbygget PDF-fremviser med *synctex* support (hop mellem PDF og editor). (c) God stavekontrol. (d) På Mac og Windows kommer den med L^AT_EX, lige til at gå til.

¹⁸ Thi den bygger på nogle kode-biblioteker som kan skal have gennem linux-distributionen i den rette version

¹⁹ Den er god til at markere ord som den mener er stavet forkert, men nogle af de forslag den kommer med kan være lidt misvisende.

Ulemper. (a) I og med det er en meget simpel editor, kan den være for simpel til meget krævende editeringsopgaver. (b) Har pt. ikke noget interface til at håndtere labels eller citeringer.

Tip C.17 (Ekstra sprog på Ubuntu).

Da TeXworks internt anvender biblioteket hunspell til stavekontrol, skal man bare installere hunspell-filerne for det pågældende sprog. F.eks. for dansk

```
%> sudo apt-get install hunspell-da
```

Tip C.18 (Ekstra sprog på Windows).

Her er tingene ikke helt så nemme. Der er to ting at finde ud af (i) hvor skal sprogfilerne ligge, og (ii) få fat i sprogfilerne.

Jf. (i) så antager vi at man anvender TeXworks 0.4 eller deromkring. Gå til **Scripts** → **Scripting TeXworks** → **Show Scripts Folder**. Gå en folder over dette og man skulle gerne finde folderen »Dictionaries«. Her skal sprogfilerne placeres.

Sprogfilerne hentes via <http://extensions.services.openoffice.org/en/dictionaries>, vælg sproget du vil have og hent filen. F.eks. hedder den dict-da-current.oxt for de danske filer. Dette er faktisk en ZIP-fil, som man skal pakke ud. Man kan bruge to filer herfra, » .aff«- og » .dic«-filen (de har samme fornavn). Disse to filer kopierer man til den folder vi fandt tidligere.

Genstart så TeXworks, og har adgang til det nye sprog.

Tip C.19 (Ekstra sprog på Mac).

Samme procedure som for Windows. Dog ser det ud til at folderen man skal placere de to filer i, er

```
<brugerkonto>/Library/TeXWorks/dictionaries
```

For at pakke .oxt-filen ud, kan det være en ide at omdøbe endelsen til » .zip«, for så kan Mac selv finde ud af at pakke den ud.

Tip C.20 (Encoding 1).

Som standard vil TeXworks gemme i UTF8-encoding²⁰ (husk dette når du skal vælge option til inputenc). Desværre er der visse editorer på Windows som ikke understøtter UTF8,²¹ med det resultat at en tekst med æøå skrevet i TeXworks og gemt i UTF8, ikke åbner korrekt i denne editor (æøå bliver til ubrugelige tegn).

Hvis man formoder at dette kan blive et problem, bør man overveje at bede TeXworks om at gemme alle fremtidige dokumenter i en anden encoding, f.eks. iso-8859-1 (også kendt som latin1).

²⁰ Som er den mest anbefalede encoding.

²¹ F.eks. TeXnicCenter.

Tip C.21 (Encoding 2).

Det vigtigste tip i Joseph Wrights blog om magiske kommentarer er

```
% !TeX encoding = UTF-8
```

eller hvilken encoding man nu anvender. Denne lille kommentar (sat i starten af filen) får TeXworks til at arbejde på denne fil i den angivne encoding uanset hvilken encoding TeXworks normalt vil arbejde med.

TeXMaker

Hjemmeside: <http://www.xmlmath.net/texmaker/>

Pris: Fri software

Encoding: Alle relevante

Platform: Linux, Mac og Windows

Fordele. Et roligt og overskueligt interface, med et sidevindue med adgang til symboler, labels etc. Seneste version kommer nu med sin egen indbyggede PDF-fremviser som understøtter hop mellem fremviser og editor. Arbejder som standard i UTF8. Lækker feature med »forslag-mens-du-skriver«, dvs. mens man skriver kan der dukke en menu op med forslag til det man er ved at skrive. Skriver man f.eks. »\use« så foreslår den noget med »\usepackage« og viser samtidig syntaksen for denne.

Ulemper. I forhold til TeXnicCenter er TeXMaker lidt mindre sofistikeret vedr. *master* konceptet når man arbejder med store projekter.

Tip C.22 (Sætte master for et projekt).

Se i **Options**-menuen, her finder man muligheden for at sætte det nuværende dokument som *master*. Dette betyder at for *denne session*, så vil TeXMaker oversætte master-filen i stedet for den fil man står i i det øjeblik man beder TeXMaker om at oversætte.

Tip C.23 (Hop mellem PDF og editor).

Af en eller anden grund er dette ikke slået til som standard. Gå ind i konfigurationen af TeXMaker, find linien med konfigurationen af pdf_latex, før »%.tex« tilføj »_synctex=1_«.

Tip C.24 (Stavekontrol under Ubuntu).

Her anvendes samme filer som under TeXworks, så se Tip C.17 på foregående side.

Tip C.25 (Stavekontrol på Windows).

Vi anvender de samme filer som under TeXworks, se Tip C.18 på forrige side. Her skal vi bare placere dem et andet sted.

Stedet kan findes via **Options** → **Configure TeXMaker** → **Editor**. Her vil man se en henvisning til en folder med sprogfiler. Læg de to filer herved, og så skulle man gerne have dansk stavekontrol i TeXMaker.

TeXstudio

TeXstudio

Hjemmeside: <http://texstudio.sourceforge.net/>

Pris: Fri software

Platform: Linux, Mac og Windows

Dette er egentlig bare en såkaldt *fork* af TeXMaker, altså nogle programmører som mente at de kunne gøre tingene bedre end de som vedligeholder TeXMaker.

Man kan se TeXstudio som TeXMaker på steroider, den har fået en del features som TeXMaker ikke har.

Ulemper. Af en eller anden grund er den indbyggede PDF-fremviser ikke slået til som standard. Fikses nemt via **Options** → **Configure TeXstudio** → **Commands**, hvor man så sætter flueben ved »*internal viewer*«.

En lidt ekstrem default syntaksfarvning.

Fordele. For meget at nævne her. Sammenlig i stedet selv.

TeXnicCenter

Hjemmeside: <http://www.texniccenter.org/>

Pris: Fri software

Platform: Windows

Ulemper. Meget rodet og uforståeligt interface. Understøtter *ikke* UTF8. Har en indbygget ”forslag-mens-du-skriver” feature, som er *meget* uintuitiv!²² Stavekontrollen (som anvender de samme filer som man anvender til TeXworks) er håbløs.

Installerer man selv TeXnicCenter så skal man selv forklare den hvor L^AT_EX er henne. De fleste andre editorer finder selv ud af dette. (Hvis du er i tvivl, så lav en søgning efter latex.exe, det bibliotek som denne findes i er det bibliotek man skal angive i TCs konfigurerings Wizard.)



Under fremviseren Sumatra PDF (se side 394), er det forklaret hvordan man justerer TC til at anvende Sumatra PDF som PDF-fremviser.

TeXShop

Hjemmeside: <http://pages.uoregon.edu/koch/texshop/>

Pris: Fri software

Platform: Mac

Stamfader til TeXworks, har nok en smule flere features end TeXworks, da den er så tæt knyttet til MAC OS X.

Da den allerede kommer med MacTeX, er det de færreste som installerer TeXShop fra hjemmesiden. Men det an være nyttigt, hvis man sidder med en Mac som anvender en NFS monteret L^AT_EX-installation (hvilket en del af vore sekretærer gør).

²² Skriver man »\begin{doc« så foreslår den fint »\begin{document}... \end{document}«-parret. Takker man ja (det er vist mellemrumstasten), så får man »\begin{doc}\begin{document}... \end{document}« som resultat. Ikke just det man forventede. Sammenlign selv med hvad samme tekststump giver af forslag i TeXMaker.

C. L^AT_EX-distributioner samt andre redskaber

Ulemper. Er som standard sat til at gemme tekst i Mac-roman encoding (svarer til APPLMAC option til inputenc), hvilket kan gøre det besværligt at udveksle filer med andre. Overvej at omkonfigurere TeXShop til at gemme i utf8.

Gnu Emacs+AUCT_EX+refT_EX

Pris: Fri software
Platform: Linux, Mac og Windows, for Mac se også Aquamacs

Der er her tale om den kraftige editor Emacs sammen med udvidelserne AUCT_EX²³ og refT_EX.²⁴ På Linux fås kombinationen normalt gennem Linux distributionen (for Ubuntu se Tip C.26), på Mac kan Aquamacs være et nemmere valg end Gnu Emacs. Officielle urler:

<http://www.gnu.org/software/emacs/>
<http://www.gnu.org/software/auctex/>

Lige pt. ved jeg ikke om man kan få en Emacs til Windows hvor AUCT_EX allerede er inkluderet, det har man tidligere kunnet, men den er væk nu. Ellers må man selv installere Emacs til Windows, og så følge vejledningen på AUCT_EX's hjemmeside.

Efter min personlige mening, er Emacs+AUCT_EX+refT_EX den ondeste editorkombination til editering af L^AT_EX-kode. Der er mange editeringsfeatures i AUCT_EX som jeg slet ikke forstår at der ikke er andre editorer som har kopieret. Er man først blevet god til Emacs, vil man ikke bruge andet. Der findes faktisk både en browser, en mailklient og en psykiater indbygget i Emacs.

Ulemper. Emacs kan være svær at vænne sig til. Skal man ændre en konfiguration, er det ikke altid lige nemt at gennemskue.

Tip C.26 (Ubuntu Emacs+AUCT_EX – installation vs. alternativ T_EX Live).

I Ubuntu er Emacs og Auctex nemt at installere via

```
sudo apt-get install --no-install-recommends emacs auctex
```

Anvender man TUGs version af T_EX Live, så vil installationen fejle. Auctex antager at man anvender Ubuntu's T_EX Live, som er placeret et meget specifikt sted (et andet sted end der vi har installeret T_EX Live).

Her må vi igen snyde en smule. Ubuntu vil klage over at »/usr/share/texmf/« mangler. Det laver vi da bare

```
sudo mkdir -p /usr/share/texmf
sudo apt-get install -f
```

Den sidste linie anvendes til at få Ubuntu til at forsøge at fuldføre den fejlslagne installation.

²³ Projektet er faktisk startet i DK, så vidt huskes står AUC for Aalborg UniversitetsCenter.

²⁴ Som i dag vedligeholdes og distribueres sammen med AUCT_EX.

Aquamacs

Tip C.27 (Emacs vs. Yap).

Hvis man anvender Emacs med AUCT_EX på Windows sammen med MiK_TE_X, så kan det være en god ide at tilføje følgende linie til sin ».emacs«-konfigurationsfil

```
(require 'tex-mik)
```

Dette vil konfigurere AUCT_EX til at blive konfigureret til at anvende en MiK_TE_X-installation (bl.a. med yap i stedet for xdvi).

Sidebemærkning C.28 (Emacs med Windows-feel).

For de mere eventyrlystne har Brian Elmegaard nævnt at følgende Emacs version skulle være nemmere at bruge for Windows brugere (f.eks. mere Windows-agtige genveje). <http://ourcomments.org/Emacs/EmacsW32.html>. Den kommer dog ikke med AUCT_EX præinstalleret, så det må man selv gøre, se AUCT_EXs hjemmeside.

Aquamacs

Hjemmeside: <http://aquamacs.org/>

Pris: Fri software

Platform: Mac

Dette er en Emacs til Mac med et mere Mac-agtigt udseende og genveje mere passende til Mac OS. Passer fint sammen med PDF-fremviseren Skim.

Fordele. Auctex og venner er *indbygget!*

WinEdt

Hjemmeside: <http://www.winedt.com/>

Pris: Shareware, test er gratis, ellers skal der købes en licens (\$30 til \$70)

Platform: Windows

Eftersigende en temmelig hardcore (på den gode måde) editor. Mange anvendelige editerings features og muligheder for selv at justere hvordan editoren skal opføre sig.

Notepad++

Hjemmeside: <http://notepad-plus-plus.org>

Pris: Fri software

Platform: Windows

TeXmacs

Hjemmeside: <http://www.texmacs.org>

Pris: Fri software

Platform: Linux, Mac og Windows

Spøjs editor som kan vise formlen mens man skriver.

C. L^AT_EX-distributioner samt andre redskaber

Led

Hjemmeside: <http://www.latexeditor.org/>

Platform: Windows

Ulemper. Har men indbygget fremviser, men det er en DVI-fremviser, hvilket de færreste anvender i dag. Er ikke opdateret siden 2009.

Kile

Hjemmeside: <http://kile.sourceforge.net/>

Pris: Fri software

Platform: Linux (og Windows)

En editor på linie med TeXmaker, bare designet til KDE-miljøet.

Lyx

Hjemmeside: <http://www.lyx.org>

Pris: Fri software

Platform: Linux, Mac og Windows

Dette er ikke helt en editor som sådan, den skal mere ses som et værktøj for de som ikke ønsker selv at skrive L^AT_EX-koden selv. Nogle vil måske se det som en kobling mellem Word og L^AT_EX.

Lyx kræver at man har en eksisterende L^AT_EX-installation, MikT_EX eller T_EX Live fungerer fint.

Fordele. Man behøver ikke kende til L^AT_EX.

Ulemper. Man kan kun tilgå de features som nogen har lavet interfaces for. Loader man f.eks. en ny pakke, så er det ikke noget interface til at anvende features fra denne pakke. Man kan dog sagtens indsætte L^AT_EX-kode i et Lyx-dokument.

Lyx gemmer dokumentet i sit eget Lyx-format. Dette betyder at skal man udveksle dokumentet med andre, skal man huske at eksportere det som L^AT_EX før man sender dokumentet til modtageren.

Lyx er udemærket til at eksportere Lyx-kode til L^AT_EX, men er ofte dårlig den anden vej. Man kan være heldig med at et eksisterende dokument kan importeres ind i Lyx, men ofte går det galt.

Scientific Workplace

Hjemmeside: <http://www.mackichan.com/>

Pris: Kommerciel (dyr)

Platform: Windows

Her er vi endnu mere ovre i Word-med-LaTeX, end vi er i Lyx. SWP kommer med sin egen LaTeX-installation, har indbyggede matematikprogrammer, tidligere var den koblet med Maple, men i dag kommer den med en andre indbyggede matematikting.

Ulemper. Lider af de samme problemer som Lyx, vedr. pakker som er understøttet, samt importen. Det kan være svært at arbejde sammen med en SWP-bruger hvis man ikke selv er SWP-bruger. En anden ting man skal være opmærksom på er, at SWP kommer med deres egen L^AT_EX-installation, som ikke har noget til fælles med MikT_EX eller T_EX Live.

SWP anvender også sit eget dokument-format. Husk at lave en *gem som...* før man sender dokumentet til andre, man skal gemme det som formatet »Portable L^AT_EX«.

C.3 Fremvisere

Hvad kendetegner en god fremviser? Tja, det kommer an på formatet og ens ønsker. Til PDF vil jeg nok sige at en fremviser med SyncT_EX support er at foretrække (sammen med en editor som også understøtter dette).

Sidebemærkning C.29 (SyncT_EX).

SyncT_EX er en feature som er indbygget i moderne L^AT_EX-versioner. Det er en feature som man evt. skal slå til mens man oversætter (det kan ikke slås til inde fra et dokument). Oftest er det bare et ekstra argument til L^AT_EX-kaldet som ens editor laver:

```
-synctex=<tal>
```

Her er <tal> lig 0 (standard), hvilket betyder at SyncT_EX er slået fra, 1 betyder SyncT_EX aktiveret og dets output skal ikke komprimeres mens -1 betyder det samme, men med GZ-komprimeret output.

Det omtalte output giver en kobling mellem linienummeret (og filnavnet) og *x-y*-koordinater i PDF-filen. Dette kan anvendes af kapable editorer og PDF-fremvisere, således at man kan hoppe fra den ene til den anden. En *yderst* nyttig feature når man retter sin tekst. Om outputtet skal komprimeres eller ej kommer an på fremviseren.

Man vil kunne se om SyncT_EX er slået til ved at der ved oversættelse, dukker en »<filnavn>.synctex(.gz)«-fil op.

En anden rar feature vil være at kunne genindlæse (reloade) filen, som skal fremvises. Nogle kan gøre dette automatisk (ikke altid lige godt), andre skal fortælles at den skal genindlæse (f.eks. ved at aktivere fremviservinduet, eller et tryk på en knap).

Advarsel C.30 (Mac Preview).

Mac har et fremviser program kaldet preview som sikkert er ganske udemærket. Har dog set meget spændende problemer med det som vi gerne vil advare imod: Har man fremvist en PDF i preview så *undlad* at gemme en kopi via preview. I flere tilfælde har vi oplevet at de indlejrede fonte i PDF-filen bliver totalt ødelagt således at filen ser her anderledes ud når man printer den. (Mega træls hvis det er eksamensopgaver.)

Man kan godt undre sig over hvorfor programmet vil ændre indholdet af PDF-filen når brugeren ikke har ændret noget.

Det kan være en ide at tjekke PDF-filens interne egenskaber (anvend Adobe Reader). Der vil man normalt kunne se hvilket program der senest har pillet ved filen.

Lad os starte med den nok mest kendte fremviser.

C. L^AT_EX-distributioner samt andre redskaber

Adobe Reader

Hjemmeside: <http://www.adobe.com>
Synctex: Nej
Format: PDF
Platform: Linux, Mac og Windows

Fordele. AR er rigtig pænt at se på på skærmen, understøtter som den eneste fremviser Javascript indlejret i PDF-filer.

Ulemper. På Windows kan AR ikke genindlæse PDF-filer, faktisk låser AR PDF-filen, således at pdfL^AT_EX ikke kan opdatere PDF-filen før den er blevet lukket i AR. I T_EX Live og MikT_EX findes forskellige redskaber som kan anvendes af editorer til eksternt at bede AR om at lukke PDF-filen, og andre til at genåbne filen. Så vidt jeg ved vil genåbning af PDF-filen altid ske på den første side.

Der er en reload featue i AR på Linux (tastaturkombination).

Javascript i PDF-filer kan være ret farlig, der findes ofte fejl i AR som så anvendes som angrebsvektor af crackere.²⁵ Omvendt kan den være nyttig til visse effekter i AR.

AR bruger utroligt meget hukommelse sammenlignet med de alternative fremvisere.



Jeg anbefaler at man normalt anvender en alternativ PDF-fremviser til sit daglige L^AT_EX-arbejde og så evt. anvender AR til at fremvise med (ved foredrag og lignende) eller til når man skal printe.

Sumatra PDF

Hjemmeside: <http://blog.kowalczyk.info/software/sumtrapdf/free-pdf-reader.html>
Synctex: Ja
Format: PDF
Platform: Windows

Alternativ *build* kan hentes fra <http://william.famille-blum.org/software/sumatra/index.html>.

Sumatra PDF bruger meget lidt hukommelse og låser ikke PDF-filen. Når PDF-filen ændres (f.eks. når man har genoversat sit dokument) så genindlæses PDFen automatisk. Man kan endda konfigurere sin editor til at få Sumatra PDF til at gøre meget skumle ting (f.eks. hoppe fra PDF tilbage til den tilsvarende kildekode).

Tip C.31 (Integration i TeXMaker).

Justeres via Options → Configure TeXMaker, vi skal rette i feltet Pdf Viewer. Start med at vælge browse således vi kan fortælle TeXMaker hvor Sumatra PDF findes. Det er oftest i Programmer eller Program Files, i underfolderen SumatraPDF. Det kunne resultere i "c:/ProgramFiles/SumatraPDF/SumatraPDF.exe", efter dette tilføjes

```
_-reuse-instance_%.pdf
```

husk mellemrummene.

²⁵ Javascript anvendes i den forbindelse til at få udført ting på ens computer hvis crackeren får hul gennem en fejl i AR. Sørg derfor altid at anvende en opdateret AR og anvend forsigtighed overfor tilsendte PDF-filer fra afsendere som man ikke kender.

Skim

Tip C.32 (Integration i TeXnicCenter).

Her går vi gennem Build → Define Output Profiles... Vælg LaTeX => PDF og gå til fanebladet Viewer. Igen starter vi med at sætte Path of executable, metoden er som beskrevet ved TeXMaker.

Herefter skal man under View projects output, markere punktet Commandline argument og i tekstfeltet skriver man

```
-reuse-instance_%bm.pdf
```

igen skal man huske mellemrummet.

Under *Forward search* kan man vælge *Command line argument* og lade den være tom (eller se Tip C.33).

Under *Close document before...* vælger man

```
do not close
```

Tip C.33 (Synctex med TeXnicCenter).

På <http://william.famille-blum.org/blog/static.php?page=static081010-000413> beskrives det, hvordan man får TeXnicCenter til at kunne hoppe til SumatraPDF samt omvendt.

Det skal nævnes at det kan være lidt svært at få til at virke, blandt andet kan det hjælpe undervejs, hvis hop fra PDF til TC (via dobbeltklik i SumatraPDF) ikke virker, at slette en SumatraPDF konfigurationsfil, som Windows holder skjult for en (her kan Google hjælpe).

Skim

Hjemmeside: <http://skim-app.sourceforge.net/>

Synctex: Ja
Format: PDF
Platform: Mac

Har mulighed for at tilføje noter til en given PDF, se mere på hjemmesiden.

Foxit

Hjemmeside: <http://www.foxitsoftware.com/products/reader/>

Synctex: Nej
Format: PDF
Platform: Windows

Understøtter muligheden for at tilføje noter til en given PDF-fil. Meget anvendt som et mere sikkert alternativ til AR.

Evince

Format: PDF og andre
Synctex: Ikke helt officielt, men der skulle eksistere patches så den kan
Platform: Linux

C. L^AT_EX-distributioner samt andre redskaber

Dette er standard dokument fremviser på Ubuntu. Genindlæser automatisk PDF-filer når de ændres.

Okular

SyncTex: Ja
Platform: Linux

Dokumentfremviser hørende til KDE²⁶, kan dog nemt installeres under f.eks. Ubuntu, med det forbehold at der skal installeres temmeligt mange ekstra ting (thi den anvender KDE miljøet, og afhænger derfor af mange af dets biblioteker).

Xpdf

Format: PDF
SyncTex: Ikke officielt
Platform: Linux

Primitiv men velfungerende PDF-fremviser. I samme familie finder man de to meget nyttige CLI²⁷ værktøjer: `pdftotext` og `pdffonts` som kan vise hhv. info om en given PDF-fil samt en oversigt over de anvendte fonte.

PDF-Xchange Viewer

Hjemmeside: <http://www.tracker-software.com/product/pdf-xchange-viewer>
SyncTex: Nej
Format: PDF
Platform: Windows

Denne er måske ikke så L^AT_EX-relateret, men den har et par tricks:

- Kan tilføje noter til vilkårlige PDF-filer.
- Forstår Javascript.
- Generel mulighed for at tegne ovenpå en PDF-fremvisning, ganske nyttigt i en undervisningssituation.

Til gengæld er der forlydender om at gratisversionen faktisk installerer Spy-ware!

GV

Format: PostScript og PDF
SyncTex: Nej
Platform: Linux

Baseret på GhostScript. Automatisk reload når filen ændres. Har den ulempe at hvis dokumentet er langt og oversættelsen er langsom, så vil GV opdage at filen er ændret, genindlæse og kan måske ikke nå frem til den pågældende side fordi oversættelsen ikke er nået så langt. Automatisk reload kan dog slås fra.

²⁶ På Linux er der to store desktop miljøer, Gnome og KDE. Ubuntu anvender normalt Gnome.

²⁷ Command Line.

Gsview32

Fordele. Hvis man fremviser en (korrekt) EPS-fil, så skrumper den hvide baggrund til at vise størrelsen af EPS-filen. Nyttigt hvis man skal tjekke om en EPS-fil har for meget luft omkring sig.

Gsview32

Format: PostScript og PDF
Synctex: Nej
Platform: Windows

Fremviser a la GV, bare til Windows.

Fordele. Har indbygget feature til at kunne oversætte EPS/PS til PDF.

Ulemper. Har ikke samme indsnævringsfeature som GV.

Xdvi

Format: Dvi
Platform: Linux og Mac (kræver kørende X-server)

Denne kommer med T_EX Live. Der er ikke så mange nye L^AT_EX-brugere som anvender dette format.²⁸

Dviout

Format: Dvi
Platform: Windows

Kommer med T_EX Live på Windows.

Yap

Format: Dvi
Platform: Windows

Standard Dvi-fremviser på MikT_EX.

C.4 Andet

Et par nyttige redskaber på Linux

Et par redskaber som også kunne være nyttige på Linux (fås sikkert også til andre styresystemer):

pdftk

Eller rettere PDF Toolkit. Godt kommandolinie værktøj til at manipulere PDF-filer.

²⁸ Jeg gør dog fortsat, idet jeg nemt kan hoppe mellem Xdvi og min Emacs. Der er desværre fortsat langt mellem PDF-fremviserne med Synctex-support på Linux.

C. \LaTeX -distributioner samt andre redskaber

Imagemagick

Denne er normalt en del af de fleste Linux-distributioner. Kort sagt lær at bruge `convert`. Rart at kende hvis man lige skal bruge en thumbnail af en PDF-fil,²⁹ eller skal konvertere noget fra TIF til PNG.

jpeg2eps

Normalt vil en konvertering fra JPG til EPS via `convert` resultere i meget store filer. `jpeg2eps` virker bedre. Programmet kan hentes via <http://www.pdflib.com/download/free-software/jpeg2eps/>.

Printerdrivere

Mange Windows programmer er ikke gearet til at gemme f.eks. figurer i formater som er velegnet til \LaTeX . F.eks. er det ikke vildt godt at anvende PNG som format til en stregtegning (især ikke hvis denne skal skaleres op). Og man kan ofte ikke gemme i PDF-format med mindre man har Adobe Destiller installeret.

Der findes alternativer. Man kan installere virtuelle printere, dvs. printere som faktisk printer til en fil.

EPS <http://www.adobe.com/support/downloads/product.jsp?product=44&platform=Windows>

PDF En Google søgning giver <http://www.primopdf.com/> og <http://www.pdf995.com/> som steder hvor man kan finde en gratis PDF-printerdriver.



Pga. kontroverser med Adobe kan Office 2007 ikke automatisk gemme som PDF (kan ikke huske om Office 2010 kan). Men Microsoft har lavet en plugin således at man kan gemme i PDF fra Office 2007, den kan findes via

<http://www.microsoft.com/downloads/en/details.aspx?displaylang=en&FamilyID=4d951911-3e7e-4ae6-b059-a2e79ed87041>.

²⁹ F.eks. via »`convert -size x150 fil.pdf fil.png`«

Om fejlmeddelelser

Som nybegynder er en af de mest frustrerende ting at lære at forstå de forskellige fejlmeddelelser man kommer ud for. Vi vil i dette kapitel forklare nogle af de mest almindelige af dem. Yderligere fejlforklaringer kan f.eks. findes i [Mittelbach og Goossens \(2004\)](#), appendix B.

Det vil også være en god ide at kigge FAQ'en igennem, se PDF-versionen af [FAQ \(2007\)](#), kapitlerne T–V er ret så informative.

D.1 Fejlmeddelelser

`<makro>` allowed only in math mode

man har anvendt et symbol eller en makro i den almindelige tekst, men den må kun anvendes i matematikmode.

`\begin{<env>}` on input line `<nummer>` ended by `\end{<and et env>}`

denne ser man også ofte hvis man skriver det hele pr. håndkraft. Reglen er at enhver `\begin{<env>}` skal afsluttes af en tilsvarende `\end{<env>}`. Situationer så som `\begin{<env1>} ... \begin{<env2>} ... \end{<env1>} ... \end{<env2>}` må ikke forekomme.

`\caption` outside float

`\caption` må normalt *kun* anvendes *indeni* float-environments. Se dog afsnittet [En figur og en tabel i samme float](#), side 153.

Command `<navn>` already defined

kommer når man via `\newcommand` vil oprette en ny makro, men det valgte navn er allerede i brug.

Command `<navn>` invalid in math mode

forekommer når man skriver noget i matematik som kun må anvendes i den almindelige tekst, f.eks. æ, ø, eller å. Skriv dem i stedet med `\text{<tekst>}`, eller `\textup{<tekst>}`.

Command `\<navn>` unavailable in encoding T1

¹ kan forekomme hvis man har en fil med æøå men gemt i en anden encoding end den som er angivet til inputenc.

Double super/subscript

som nævnt tidligere (se afsnit [3.4 på side 68](#)) må man f.eks. ikke lave »x_i^j« der mangler `{}`-er.

Environment `<name>` undefined

gæt selv. Tjek om man har stavet det rigtigt, eller har husket at load den pakke environmentet kommer fra.

¹ Fixme Note: huh???

D. Om fejlmeddelelser

Extra `},` or forgotten `$`
fejlen er lige hvad den siger.

Dog ikke altid. Denne meddelelse kan man også få hvis man har anvendt en såkaldt skrøbelig (fragile) makro i argumentet til en 'bevægelig' makro. Dette kan være `\caption` eller `\section's`. Se <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=extrabrace> for mere information vedrørende denne fejl.

Extra alignment tab has been changed to `\cr`
ses ofte i tabeller (eller *`eqnarray`*, *`split`*) hvor man har anvendt flere `&`-er på en linie end der skulle være. Problemet er ofte at man har glemt en `\\` i rækken før.

Extra alignment tab has been changed to `\cr`. `\endtemplate ...`
Dette betyder ofte at man har en konstruktion hvor man har specificeret flere søjler end konstruktionen kan klare. F.eks. understøtter *`pmatrix`* normalt højst 10 søjler. Noget man dog kan ændre, se Bemærkning 3.35 på side 72.

Extra `\right`
den tilsvarende fejl af at have glemt en `»\left«`.

Float(s) lost
her er man kommet til at placere en float (*`figure`* eller *`table`*) et sted hvor man ikke kan placere noget som flyder, eksempelvis inden i en *`minipage`*, `\parbox`, `\marginpar` eller `\footnote`. Det kan være en ret svært at finde tilbage til det sted hvor det er gået galt.

Husk også at `\marginpar` og `\footnote` faktisk også er floats. Dette betyder at man ikke kan placere en `\marginpar` inden i en `\footnote`.

Pas også på med konstruktioner som f.eks. skjuler at den anvender en `\marginpar`, eksempelvis `\fixme` (her kan man komme ud over det via `inline` option til `\fixme`).

Graphics division by zero
Dette kommer som oftest hvis man har drejet en inkluderet figur samtidigt med at man sætter højden. Alt efter hvor meget man har drejet kan disse beregninger ende op med at gå galt. Anvend i stedet `totalheight`-optionen til `\includegraphics` i stedet for `height`.

Illegal parameter number in definition of $\langle command \rangle$
som nævnt i et andet kapitel anvendes tegnet $\# \langle num \rangle$ til at angive hvor de enkelte parametre skal anvendes i en makrodefinition. Denne fejl kan forekomme hvis de makroer man har angivet ikke anvender det rigtige antal argumenter. Se [Eksempel D.1](#).

Eksempel D.1

```
\newcommand\test[1]{#1 og så #2}
```

Her er fejlen naturligvis, at `[1]` jo angiver at denne makro kun har ét argument, hvorfor `#2` ikke giver mening.

Bemærk dog at man godt kan have makroer som bliver defineret med f.eks. brug af `#1` men hvor man *ikke* har angivet noget antal af argumenter. I et sådant tilfælde fungerer det faktisk idet makroen anvendes indeni en anden makro som så sørger for at sætte `#1` til en brugbar værdi.

Illegal unit of measure (pt inserted).
angiver at man har stavet en længde enhed forkert, eller måske har man anvendt \pm -syntaksen, som man får via `calc`-pakken, et sted hvor denne ikke kan anvendes.

Fejlmeddelelser

Misplaced alignment tab character &

denne forekommer når man anvender & udenfor de områder hvor & anvendes til at angive alignment (eller adskille søjler). Har man brug for tegnet & så anvend `\&` i stedet.

Missing \$ inserted

skyldes oftest at man har forsøgt at anvende noget i den almindelige tekst som ellers kun findes i matematikmode. Et eksempel kunne være »_« som i stedet skrives med `»_«` eller det kan være et symbol. Hvis man er ved at angive et filnavn hvori der indgår `»_«`, da kan `\path` fra `url`-pakken være en fordel. Her må man gerne anvende `»_«`.

Dette kan også skyldes at man er kommet til at efterlade en blank linie i et environment til fremhævede formler – det må man nemlig ikke.

Missing `\begin{document}`

ses ofte når man har lavet en fejl i preamble, kunne være man har glemt nogle `{}`-er et sted i preamble.

Dette kan også forekomme hvis der har været en outputfejl ved skrivning til `.aux`-filen. Tjek efter om der i `.log`-filen står `./fil.aux` før fejlen forekommer. I dette tilfælde kan det normalt løses ved at slette `.aux`-filen og så starte oversætningen forfra.

Missing delimiter (. inserted)

man har her brugt `\left`, `\right` eller en af `\big`'erne på noget som ikke kvalificerer som en delimiter (hegn).

Missing `\right .` inserted

man har glemt den tilhørende `\right` til en `\left`. Bemærk at den som L^AT_EX indsætter er den blanke delimiter `»\right.«`.

Multiple `\label's`: label *<tekst>* will be lost

man må kun have en label pr. nummereret formel (når man anvender `amsmath`).

No room for new `\count`

Standard T_EX har kun plads til et bestemt antal tællere. Har man indlæst en masse ekstra pakker så kan man nemt komme i bekneb. Løsningen er at anvende udvidelsen eT_EX. Det gør de fleste allerede, men man skal lige aktivere de ekstra tællere, hvilket kan ske ved at tilføje

```
\RequirePackage{etex}
```

før `\documentclass` linien, øverst i sit dokument. I mange tilfælde kan man også slippe afsted med bare at anvende den i preamble med `\usepackage{etex}`, *memoir*-klassen loader automatisk denne pakke, da *memoir* anvender masser af længder og tællere.

No room for new `\dimen`

Samme som ovenfor, nu bare vedr. længder. Løsningen er den samme.

No room for new `\write` (eller) `\read`

T_EX har kun plads til at kunne skrive til 16 filer samtidigt, dvs. holde 16 filer åbne samtidigt for at kunne skrive i dem. Den kan også kun læse fra 16 filer samtidigt. Der er ikke noget at gøre i dette tilfælde. Dog kan man nogle gange opdage at nogle af de filer som der skrives til slet ikke skal bruges til noget. F.eks. vil pakken `ntheorem` lave en `.thm`-fil som man ofte ikke behøver. Finder man sådanne, kan man forsøge at omdefinere noget pakke kode (i sin preamble) således at denne fil allokering ikke kommer til at finde sted.

Eksempel
D.2

D. Om fejlmeddelelser

Runaway argument

her har man sandsynligvis glemt en afsluttende } et sted. Meddelelsen findes i mange varianter.

T_EX capacity exceeded *(forklaring)*

I forklaringen vil man ofte se at L^AT_EX forslår at man får forøget hukommelsen i T_EX. I 95 % af tilfældene skyldes denne fejl *ikke* at L^AT_EX mangler hukommelse, men i stedet en makrofejl eller manglende indsigt i hvordan L^AT_EX virker.

- (i) Ved makrofejl, er det ofte en dårligt designet makro som giver problemet. F.eks. hvis der er kommet en situation med rekursion som ikke stopper.
- (ii) En anden fejl som er set et par gange er at man glemmer at L^AT_EX arbejder med teksten et afsnit ad gangen. Men der er en grænse for hvor lange afsnit der kan være i L^AT_EXs hukommelse. Her løses problemet naturligvis ved at sørge for at starte nye afsnit. På det seneste var der en person som ville lave en lang adresse liste. Hvert sæt af adresse oplysninger var pakket ind i en *minipage*, så de ikke blev delt over en side. Men vedkommende havde glemt at starte et nyt afsnit efter hver adresse, så L^AT_EX løb tør for hukommelse, da den blev nødt til at læse hele dokumentet ind i hukommelsen før den begyndte at kigge på linie- og sideombrydninger.

There's no line here to end

ses ofte når folk gerne vil have en ekstra blank linie, den laver de så ved at indsætte \\ to gange eller f.eks. efter en paragraf, environment eller blank linie. Det er bare brugerens dårlige vane. Oftest har disse blanke linier ikke nogen berettigelse.

Token not allowed in a PDF string (PDFDocEncoding)

(faktisk en warning, ikke en error), ses hvis man anvender hyperref-pakken og anvender f.eks. matematik i overskrifter, disse vil nemlig blive kopieret til en bookmarkliste i PDF-filen og disse bookmarkstrengte er ret strikse med hvad de må indeholde. Løsningen er at anvende \texorpdfstring, se [Tip 10.13 på side 287](#).

Too deeply nested

ses i forbindelse med nestning² af *enumerate* eller *itemize*. L^AT_EX listerne kan kun nestes ned til niveau 4.

Too many unprocessed floats

betyder at L^AT_EX har haft mere end 18 floats liggende i hukommelsen som den ikke har kunnet placere. Kunne f.eks. være forårsaget af en meget stor float som spærrer det hele for en masse små. Indsæt evt. en \FloatBarrier eller \clearpage et sted. Eller prøv at anvende det ekstra argument til \begin{figure} etc.

Kan også skyldes at man har anvendt for mange \marginpars i en paragraf (der må højst være 9!).

Undefined control sequence

er nok den aller mest almindelige fejlmeddelelse man kommer ud for – betyder at en makro man har anvendt ikke findes. Ofte skyldes det bare at man har stavet det forkert eller har glemt et mellemrum. Fejlmeddelelsen indeholder også information om hvor fejlen er, ofte endda et udsnit af teksten hvori man for det meste kan finde makroen som er forkert.

² At *nest* betyder at man har en konstruktion lavet inden i en konstruktion af samme type.

Advarsler

- `\verb` ended by end of line
argumentet til en `\verb` skal stå på samme linie!
- `\verb` illegal in command argument
`\verb` må ikke anvendes i argumenter til andre makroer. Se evt. `fancyvrb` for hjælp til alligevel at kunne gøre noget lignende.
- `\vref` or `\vpageref` at page boundary A-B (may loop)
kommer når man har anvendt pakken `varioref` og man har anvendt f.eks. `\vref` når et sideskift. Pakken `varioref` tjekker nemlig om starten og slutningen af resultatet af `\vref{<label>}` står på samme side, hvis de ikke gør, kan det betyde at man ikke kan få dokumentet til at stabilisere sig (man får hele tiden at vide at der labels som er ændret). Det vil jo heller ikke være smart med referencen »figur 2.3 på næste side« hvis »næste side« faktisk stod på samme side som figure 2.3.
- You can't use 'macro parameter #' in ...
man har sandsynligvis glemte at `#<n>` anvendes til at angive parametre med i makro definitioner – ønskede man at skrive `#` anvendes `\#` i stedet.
- You can't use '`\spacefactor`' in vertical mode
denne fejl skyldes oftest at man har prøvet at lave om på en makro hvor der indgår `@` i navnet og man samtidigt ikke har gjort opmærksom på at man ved hvad man gør. Løsning, sæt et `\makeatletter... \makeatother`-par omkring.
- You haven't defined the language `<prog>` yet
denne fejl ses ofte når man lige har skiftet sprogopsætning på et allerede kompileret dokument, f.eks. fra `danish` til `english`. Fejlen skyldes at `babel` skriver sprogopsætningen til `.aux`-filen, så nu har man et dokument som siger at den skal anvende `english` men `.aux`-filen siger at den skulle anvende `danish`. Løsning: Kompiler én gang mere så overskrives `.aux`-filen og problemet forsvinder.

D.2 Advarsler

Vi har ikke tænkt os at sige så meget vedrørende warnings, bare lidt om de advarsler de fleste kommer ud for.

L^AT_EX advarsler er på formen

```
LaTeX Font Warning: Font shape `U/stmry/b/n' undefined
(Font)                using `U/stmry/m/n' instead on input line 17.
```

Eller det kan være direkte fra pakker. Desuden har man de to meget almindelige advarsler/informationer

```
Overfull \hbox (4.28769pt too wide) in paragraph at lines 189--195
\T1/cmr/m/n/10.95 man ved hvad man gør. Løs-ning, sæt et []\T1/cmtt/m/n/10.95 \
makeatlet-ter[]\T1/cmr/m/n/10.95 ... []\T1/cmtt/m/n/10.95 \makeatother[]\T1/
cmr/m/n/10.95 -
```

og

```
Underfull \hbox (badness 10000) in paragraph at lines 189--189
```

Der kan også være tale om »underfull \vbox«

D. Om fejlmeddelelser

D.2.1 Ikke helt så alvorlige advarsler

Font advarsler

Font relaterede advarsler kan man normalt bare ignorere. De skyldes oftest at man har anvendt en blanding af form, familie og serie (shape, family og series) som den nuværende skrifttype ikke understøtter. F.eks. har Computer Modern ikke nogen fed skrivemaskine variant. Man ser normalt også sjældent en sans serif font som har small caps.

Underfull advarsler

De to underfull advarsler vedrørende `\hbox` og `\vbox` skyldes at L^AT_EX har set sig nødsaget til at strække tingene mere end den selv har syntes om.

Underfull `\hbox` med en badness på under 10000 betyder at L^AT_EX er blevet nødt til at strække størrelsen på ordmellemrummet til en sådan grad at man faktisk kan se det. Det ses ofte i litteraturlister. Badness-tallet viser hvor slet L^AT_EX mener det er, jo højere værdi, jo værre er det. Værdien 10000 er det maksimale. De ses ofte når folk har været slemme og har lavet følgende konstruktion

Eksempel D.3

```
tekst tekst\  
tekst
```

Dette er virkelig slemt og vi ser desværre en del brugere som anvender denne konstruktion. Hvilket er årsagen til at vi slet ikke omtaler tvungen lineskift i teksten. Det er slet ikke noget man har brug for og man skal derfor holde sig fra det.

I de få tilfælde hvor man gerne ville indskyde en blank linie for at starte på noget nyt, så kan man anvende

Eksempel D.4

```
\fancybreak{} % eller  
\fancybreak{${}\quad*\quad*}
```

hvis man altså anvender *memoir*-klassen. Dette med at ville have en blank linie mellem afsnit, hører ikke hjemme i god typografi og ønsket herom er en vane man bør aflægge sig hurtigst muligt. Afstand mellem afsnit er noget som er tilladeligt i lyrik, breve og opslag/posters.

Underfull `\vbox` kommer når L^AT_EX bliver nødt til at strække afstanden mellem elementerne på siden for at opnå en fast højde på alle sider. Der er to almindelige årsager

- (a) En afsnitstitel (`\section` etc.) er kommet for tæt på bunden af en side og flyttes derfor om på næste side. Men så må siden jo strækkes for at opnå den korrekte højde. Med *memoir*-klassen kan dette afhjælpes ved at placere følgende i preamblen

Eksempel D.5

```
\raggedbottomsectiontrue
```

Så vil denne side løbe kort i stedet for at blive strakt, det ser pænere ud.

- (b) Den anden almindelige årsag er længere matematiske udregninger. L^AT_EX bryder sig ikke om at starte en side med et matematisk udtryk, så her vil f.eks. *align* flytte udtrykket om på næste side, hvilket efterlader nuværende side strakt mere end L^AT_EX kan lide. Det er ofte også ret tydeligt. Man har umiddelbart to muligheder: (i) Skrive teksten om således at tingene enten er fyldt bedre ud eller så udregningen kan være på siden. (ii) Eller hvis man går fra en venstre- til en

Advarsler som man bør rette

højreside, så kan man overveje at dele udregningen over to sider, se afsnit [4.2.6 på side 98](#). Det noget man helst bør holde sig fra, og det må aldrig anvendes ved sideskift fra en højreside til en venstre.

D.2.2 Advarsler som man bør rette

Man bør i hvert fald sørge for at de er rettet i den endelige version af dokumentet.

Labels og citeringer

Vi har allerede set på labels og citeringer og de meddelelser de kan give. Her er først et udvalg til labels

```
LaTeX Warning: Label(s) may have changed. Rerun to get cross-references right.
LaTeX Warning: Label `key' multiply defined.
LaTeX Warning: There were multiply-defined labels.
LaTeX Warning: Reference `key1' on page 1 undefined on input line 6.
LaTeX Warning: There were undefined references.
```

Samt et par stykker til citeringer (her i en situation hvor vi har anvendt pakken natbib):

```
Package natbib Warning: Citation `foo' on page 218 undefined on input line 284.
Package natbib Warning: There were undefined citations.
```

Dette er naturligvis ting man altid skal udbedre inden man afleverer sit færdige dokument.

Den notoriske 'overfull \hbox'

Denne advarsel er den mest almindelige advarsel og der er *ingen* som undslipper den. Bemærk at man med klasseoptionen *draft* kan få vist de overfull \hbox'e markeret med sorte kasser i højremargin.

Advarslen betyder at her er noget som er blevet for langt til at kunne være på linien. Dette kan enten være tekst som L^AT_EX ikke kan orddele, hvilket så kommer ud som

```
Overfull \hbox (18.68895pt too wide) in paragraph at lines 6--7
\T1/cmr/m/n/10 hæn-gende sæt-
```

dvs. L^AT_EX fortæller hvor stor overskridelsen er samt den fortæller hvor de steder den kunne lave orddeling lå i dette tilfælde. Man får desuden at vide i hvilke liner (sådan nogenlunde) i kildekoden problemet findes. Løsningen er her at gå ind og hjælpe med orddelingen, enten via \- eller ved at skrive lidt af teksten om. Overskridelser lavet med matematik ser cirka ud på samme måde.

Generelt kan man sige at overskridelser på op til 3–4 pt godt kan accepteres, man kan få L^AT_EX til kun at brokke sig over overskridelser på over 3 pt via \hfuzz, placér bare følgende i preambelen:

```
\hfuzz=3pt
```

Eksempel
D.6

Kommentarer

E.1 The Intent to indent

Helt tilbage i starten af bogen nævnte vi at vi på det kraftigste fraråder at man fjerner afsnitsindenteringen og anvender luft mellem afsnit i stedet. Lad os forklare hvorfor.

At skrive en opgave har to formål: (i) At demonstrere at man kan løse en given opgave, og (ii) demonstrere, at man kan kommunikere dette på en forståelig vis. Det er det andet formål som er i spil her. Læseren må aldrig komme i tvivl om hvad skribenten mener med teksten. Hele humlen er hvordan læseren kan se om der et givet sted i teksten er nyt afsnit.

For flere hundrede år siden var der ikke noget problem, man skrev ud i en køre (pergament var dyrt) og markerede nyt afsnit med »¶«. Nogle gange ville man endda markere mellemrum via »•«.

Er der tale om ren tekst, er det ikke noget problem at man markerer starten af et nyt afsnit med en blank linie, men generelt går det galt. I de følgende eksempler skal man forestille sig at man står over for et dokument uden afsnitsindentering samt hvor der er luft mellem afsnit.

- bla
- bla bla.

Sætning lige efter en liste...

**Eksempel
E.1**

Er der nyt afsnit efter en liste (der er jo luft efter)?

Et billede.



Figur 1: En figur

Fra figur 1 ses...

**Eksempel
E.2**

Er der nyt afsnit før den sidste sætning?

$$a^2 + b^2 = c^2.$$

Starten på en ny sætning...

**Eksempel
E.3**

Er der nyt afsnit her? Her er så et eksempel på at tolkningen af en sætning kan være helt forskellig alt efter om der er nyt afsnit før sætningen eller ej:

Matematik bemærkninger



Det er ikke muligt at lave en en til en oversættelse til Word (eller den anden vej). Det som en del anbefaler er at man går over HTML (L^AT_EX kan oversættes til nogenlunde til HTML med `htlatex`), dette kan så importeres i Word. Matematikken bliver dog til billeder som ikke kan redigeres. OpenOffice/LibreOffice skulle også kunne lave visse ting sammen med L^AT_EX.

E.3 Matematik bemærkninger

Dette afsnit omhandler små finurligheder folk er kommet ud for og som vi derfor lige vil vende så andre også er klar over dem. Afsnittet er kun ment for de interesserede.

Lidt mere om align til højre for lighedstegn

Vi har tidligere nævnt at ønsker man at aligne med *align*, men til højre for f.eks. et lighedstegn, i stedet for det mere normale, til venstre. Så skal man huske at indsætte nogle {}-par på udvalgte steder. Hvorfor nu det?

For bedre at illustrere begrundelsen har vi forstørret det almindelige mellemrum omkring »=«.

```
\begin{align*}
X = & X \\
& + X \\
= & X \\
\end{align*}
versus
\begin{align*}
X = {} & X \\
& + X \\
= {} & X \\
\end{align*}
```

	$X = X$
	$+ X$
	$= X$
versus	
	$X = X$
	$+ X$
	$= X$

Eksempel
E.5

Luften før og efter »=« bør være den samme.

Matematisk mellemrum mellem symboler

Vi har tidligere set at L^AT_EX selv sørger for at indsætte det rigtige mellemrum mellem de forskellige symboler, operatorer osv. Desværre er dette ikke helt rigtigt. Der er et par eksempler hvor man bliver nødt til at hjælpe til selv. Et eksempel er:²

```
\begin{gather*}
\sin \circ h \\
\end{gather*}
```

$\sin \circ h$

Eksempel
E.6

som man kan se er mellemrummet mellem »o« og »h« forkert. Forklaringen er som følger. Teknisk set deler L^AT_EX matematikken op i otte grupper³ og tildeler spacingen efter deres indbyrdes rækkefølge. I vores sammenhæng er det kun grupperne Ord(inary), Op(erator), Bin(ary) og Rel(ation) som er interessante. Den specielle regel er, at et Bin symbol i visse kombinationer internt laves om til en Ord, fordi dette Bin symbol parret

² Indrapporteret af Rasmus Villemoes.

³ Se [Mittelbach og Goossens \(2004\)](#), side 524–526 specielt tabel 8.7.

E. Kommentarer

med det foregående ikke giver logisk mening at sætte ved siden af hinanden. Blandt de fire typer vi ser på, er de problematiske kombinationer

Op–Bin, Bin–Bin, Rel–Bin og Bin–Rel.

Altså giver det ikke mening (ud fra et logisk synspunkt) at sætte `\sin` (Op) ved siden af `\circ` (Bin), hvorfor `\circ`'s type ændres til Ord og da h også er Ord, kommer der intet mellemrum mellem dem.

Men for en matematiker er der jo ingen forskel, $(\sin \circ h)(\theta)$ er jo bare funktionssammensætningen $\sin(h(\theta))$. Løsningen, hvad angår spacingen, er at tvinge h eller `\sin` over i en anden gruppe. Dette gøres via `\mathbin`, `\mathop`, `\mathord` eller `\mathrel`. I vores eksempel skal vi anvende `\mathord{\sin}`.

`\mathbin`
`\mathop`
`\mathord`
`\mathrel`

Eksempel E.7

<pre>\begin{equation*} \mathord{\sin} \circ h \end{equation*}</pre>	$\sin \circ h$
---	----------------

De fire kommandoer ovenfor kan også anvendes i mange andre sammenhænge. F.eks. benyttes symbolet `\square` (`\square`) til tider som operator eller som binær operation, men $x \square y$ (`\square y`) har forkert spacing som en binær relation, så man kan jo lave sin egen version hvor man ændrer `\square`'s klasse til Bin.

Eksempel E.8

<pre>\begin{gather*} x\mathbin{\square} y \quad \quad \text{vs.} \quad x \square y \end{gather*}</pre>	$x \square y \quad \text{vs.} \quad x \square y$
--	--

Forkert mellemrum efter `\left(` eller `\bigl(`

Mads Sørensen har gjort opmærksom på at det modsatte – altså for meget spacing, også forekommer. Dette kan anvendes som argument til *altid* at sørge for at anvende `l` og `r` varianterne af `\bigl` makroerne.

Eksempel E.9

<pre>\begin{gather*} \sin\bigl(\quad \quad \text{vs.} \quad \quad \quad \sin\bigl(\end{gather*}</pre>	$\sin(\quad \quad \text{vs.} \quad \sin($
---	--

Problemet er her at `\bigl(` (« ikke klassificeres som et ‘Open’ objekt, men som en ‘Inner’, se [Mittelbach og Goossens \(2004\)](#)), hvilket igen betyder at `\sin` (med rette) indsætter ekstra spacing hvilket vi kan se er forkert i dette tilfælde. Problemet forekommer specielt ved brug af `\left... \right`-konstruktioner.

Det generelle råd er at man ved små størrelser skalerer manuelt, og ved store gør mellemrummet ikke så meget. Hvis man er meget pernitten, så kan man anvende

Eksempel E.10

<pre>\begin{gather*} \sin\mathopen{}\left(\frac{1}{2}\right) \quad \quad \text{vs.} \quad \sin\left(\frac{1}{2}\right) \end{gather*}</pre>	$\sin\left(\frac{1}{2}\right) \quad \text{vs.} \quad \sin\left(\frac{1}{2}\right)$
--	--

E.4 Makrobemærkninger

Vi har allerede nævnt at kigger man i kildekoden til forskellige pakker så vil man ofte se makroer så som `\def`, `\edef` etc. Det er også bemærket at disse bør man *ikke* anvende medmindre man direkte ved hvad man gør. Anvend i stedet `\newcommand` eller lignende.

Der er dog en ting som man kun kan lave via `\def`, nemlig specielle afgrænsede argumenter. Antag f.eks. at vi ville lave en smart makro til at lave en retteliste til en bog som denne. Her er et forslag:

```
% #1 = hvor
% #2 = hvem kom med rettelsen
% #3 = fejlen
% #4 = det rigtige
\def\rettelse#1 by #2 : #3 -> #4\par{%
#1\quad(#2):\quad
>>#3<< rettes til >>#4<<\par\noindent
}

\noindent
\rettelse s. 2005, l. 45 by daleif : hest -> æsel

\rettelse s. 2006, l. 32 by daleif : $a+b$ -> $A+B$
```

Eksempel
E.11

<p>s. 2005, l. 45 (daleif): »hest« rettes til »æsel « s. 2006, l. 32 (daleif): »$a + b$« rettes til »$A + B$ «</p>

Tricket her er brugen af `\par` som en del af argument specifikationen. `\par` er det den blanke linie laves om til når \LaTeX læser koden. Så `\rettelse` læser faktisk et helt afsnit. Så hvis man bare sørger for at have blanke linier mellem sine rettelser, så kan man uden problemer have ret lange rettelser. Bemærk også at `»_by_«`, `»_:_«` samt `»_->_«` er en del af argumentet, de er afgrænsere (inklusive mellemrummene).

Dette er et af de små tricks man kan lave hvis man forstår lidt mere vedrørende \TeX -makroprogrammering. Men det er langt ude over grænserne for denne bog.

E.5 Lidt om filtyperne anvendt af \LaTeX

Når man arbejder med \LaTeX vil man opdage at \LaTeX faktisk laver en hel del forskellige filer. Her vil vi lige opsummere betydningen af nogle af disse filer.

- .tex \LaTeX fil (kildekode til dokument eller del af et dokument).
- .aux Den vigtigste fil \LaTeX anvender (ud over selve dokumentkildekoden), den anvendes bl.a. til at holde informationer om krydsreferencer og citeringer, mellem de enkelte \LaTeX -oversættelser.
- .dvi Binært output format fra standard- \LaTeX .
- .ps PostScript-fil.
- .pdf PDF (portable document format)-fil.
- .log Forskellige former for informationer om oversættelsen, mere udførlig end den man ser som output fra \LaTeX .
- .toc Ekstern datafil til indholdsfortegnelsen.
- .lof Samme type, bare til listen over figurer.
- .lot Samme type, til listen over tabeller.

E. Kommentarer

- .sty Udvidelsespakker, se afsnit 1.6.3 på side 12.
- .cls Dokumentklassefiler, se afsnit 1.6.2 på side 10.
- .dtx Dokumenteret kildekode til pakker eller klasser.
- .ins Driver fil til at få den rensede pakke/klasse kode ud af en .dtx fil. Kører man L^AT_EX på en .ins fil får man .sty-filen hevet ud af .dtx-filen.
- .bbl Litteraturliste lavet via BibT_EX. Se afsnit 8.3 på side 199.
- .idx Rå uformatteret index data (indekser behandles ikke yderligere i denne bog).
- .ind Formateret index.
- .out Filtype som anvendes af *comment*-environmentet.
- .end Endnotes skrives til denne fil.

Der er desuden mange andre filtyper i spil blandt de filer som L^AT_EX og venner gør brug af. Specielle pakker kan sagtens anvende deres helt egne eksterne filer til et eller andet. I skrivende stund har dette dokument's masterfil, 16 forskellige filendelser.

E.6 Lidt om de pakker som anvendes til at lave denne bog

Beskriver kun de offentligt tilgængelige pakker som er anvendt til denne bog.

amsmath

bedre matematik

afterpage

til at indsætte noget kode direkte efter den nuværende side

babel f.eks. dansk opsætning af standard ord. Iværksætter desuden orddelingsmønstrene.

berasans

en anden type sans serif

beramono

tilsvarende for mono-space font

bm ordentlig fed matematik. Så man f.eks. kan få et fedt x i den rigtige form (kursiv). Kan desuden gøre symboler fede, selv om de ikke findes i en fed version.

calc mulighed for at lave beregninger direkte i f.eks. `\set length`

caption

til konfigurering af hvordan captions ser ud⁴

color farver, til f.eks. links ved hyperlinks, samt baggrunde ved kode eksempler og overskrifter.

comment

til at kunne udkommentere store stykker tekst, jeg anvender det normalt ikke ret meget, da min editor, nemt kan udkommentere store områder med % i starten af alle linier.

csquotes

pakke som giver en sprogafhængige makro til håndtering af anførselstegn.

dfltxbmarkup

giver makroen `\markup` som anvendes i bogen til at skrive nøgleord/pakkenavne/etc. i teksten, margin og i indekset samtidigt.

dfltxbmisc

forskellige makroer anvendt i bogen. F.eks. til at give `{\{tekst}}` eller .

⁴ FixMe Note: så vidt jeg ved loader vi faktisk slet ikke caption pakken direkte, men gennem subfig

Lidt om de pakker som anvendes til at lave denne bog

dlfltxbcodetips

når vi nu har lavet en pakke til at holde nogle centrale kodetips, så kan vi lige så godt selv gøre brug af pakken.

dlfltxbtocconfig

pakken som håndterer den måde vi sætter indholdsfortegnelserne op

enumitem

bedre kontrol over standardlisterne.

etex L^AT_EX har et begrænset antal tællere, længder etc. til sin rådighed. Denne pakke forøger bl.a. dette antal

fix-cm

Computer Modern (og andre skrifttyper) i vilkårlige størrelser. Nogle gange er den brugbar og andre gange er den ikke. Burde kunne erstattes med pakken anysize.

fixltx2e

retter forskellige bugs i L^AT_EX-kernen. Denne pakke bør man altid bruge ved større projekter.

fixme noter til en selv

fontenc

gør at æøå nu er »rigtige« bogstaver, og tillader dermed orddeling efter æøå

graphicx

inklusion af ekstern grafik samt metoder til at forstørre tekst samt rotere eller spejle

hyperref

gør at alle krydsreferencer, citeringer og indholdsfortegnelser bliver lavet om til interne hyperlinks. Alle `\url` makroer bliver til eksterne hyperlinks. Giver desuden andre features til hyperlinks.

ifsym flere tekstsymboler, se evt. [Pakin \(2003\)](#).

inputenc

tillader direkte brug af bl.a. æøå i teksten

keyval

anvendes internt i diverse pakker til at kunne give $\langle \text{nøgle} \rangle = \langle \text{værdi} \rangle$ -syntaks

kpfonts

skrifttypen anvendt i denne bog. Giver både matematik og tekstfonte, hvilket er forklaringen på hvorfor vi ikke anvender `amssymb` i opsætningen af denne bog.

lipsum

giver adgang til en masse dummy tekst.

listings

til behandling af kildekode

longtable

tabel som kan side ombrydes

mathtools

ekstra matematik konstruktioner

mdframed

giver en anden mere konfigurerbar version af framed environmentet, som giver en *åben frame* ved deling over sider

mflogo

giver logoerne `\MF` (METAFONT) og `\MP` (METAPOST)

microtype

forskellige mikrotypografiske effekter

multicol

giver bl.a. environmentet *multicols* som man kan benytte til at skrive i flere kolonner inde midt i ens dokument. Også anvendeligt til posters.

E. Kommentarer

- natbib**
udvidede citerings muligheder
- ntheorem**
til opsætning af sætninger
- paralist**
flere ekstra typer lister, indlæst således at den ikke rører de originale lister.
- placeins**
giver `\FloatBarrier`
- ragged2e**
giver panderter til f.eks. `\raggedright` og `\raggedleft` som tillader en hvis grad af ordde-
ling. Meget anvendeligt hvis man i en tabel har en søjle med tekst over flere linier.
- soul** kan blandt andet give letterspacing (i.e. tekst hvor der er forøget afstand mellem
bogstaverne)
- stmaryrd**
flere matematiske symboler
- svn-multi**
giver adgang til Subversion data indlagt i kildefilerne. Vi anvender blandt andet dette
til at give oplysningerne om hvornår de enkelte kapitler sidst er blevet opdateret. Det
skal dog siges at den version som anvendes til bogen er en speciel version som Martin
Scharrer har stillet til rådighed. Denne udgave anvender ikke eksterne filer til at gemme
data.
- subfig**
under-nummerering af figurer, bemærk at denne pakke automatisk loader pakken `cap-
tion`.
- varioref**
lettere intelligente krydsreferencer
- textcomp**
tekstsymboler
- threeparttablex**
udvidelse til `threeparttable`, til noter i tabeller.
- tikz** skummelt tegnesystem til LaTeX, anvendes i vores tilfælde til at give specielle effekter
- url** giver `\url` til at skrive URLer med. Er i stand til at linieombryde URLer på en hensigts-
mæssig måde.
- verbatimcopy**
egen udviklet pakke (nu også tilgængelig via `CTAN`), tillader LaTeX at tage kopier af
tekst-filer. Den anvendes internt i en anden (endnu) ikke-offentlig pakke til at lave de
eksempel-filer man kan downloade.
- xparse**
parser pakke til LaTeX3 syntaks. Gør det meget nemmere at lave makroer med komplekse
argumenter, så som `{\[]{}[]}...`
- xspace**
giver makroen `\xspace` som kan sørge for at en makro som anvendes i teksten, og som
ikke tager et argument, nu faktisk *ikke* spiser et eventuelt mellemrum efter makroen. Jeg
har defineret en `\ctan` makro således:

Kodetip

E.1



```
\newcommand\ctan{\textsc{ctan}\xspace}
```




Listen over anvendte pakker bag på forsiden er lavet via

```
\begin{multicols}{4}
  \raggedright
  \ttfamily
  \obeylines
  afterpage
  ...
\end{multicols}
```

Eksempel
E.12

`\obeylines` gør her at lineskift i kildekoden overholdes i output, normalt bliver et enkelt lineskift jo bare oversat til et mellemrum.

E.7 Ordbog

Dette er en miniordbog over nogle af de termer vi anvender i bogen.

Master

Et *master*-dokument eller *master*-fil, er det dokument som kan oversættes med L^AT_EX. I mange tilfælde er dette bare en enkelt fil, men ved større projekter vil *masteren* inkludere kapitler fra eksterne filer.

Bemærk at de fleste gode editorer til L^AT_EX, kender til *master*-begrebet, dvs. de kan konfigureres således at sidder man og skriver i et kapitel, og så ønsker at oversætte dokumentet, da ved editoren hvilken fil den skal oversætte i stedet for den nuværende fil.

Tekstblok

Typografisk set er *tekstblokken* det område hvori selve teksten befinder sig. Det er et rektangel af bredde `\textwidth` og højde `\textheight`. Disse to længder man kan henvise til, men man bør ikke ændre dem per håndkraft (med mindre man har meget god grund til det).

Oversætte

Betegnelse for det at fodre en L^AT_EX-variant med et dokument. Til tider kan oversættelsen også involvere brug af B_IB_TE_X eller `makeindex`.

Kompilere

Se *oversætte*.

Environment

Navn for en konstruktion som indenfor et begrænset område kan ændre hvordan tingene ser ud eller opfører sig. Af pædagogiske årsager anvender man normalt ikke den danske oversættelse *miljø*.

Header/footer

Er den engelske betegnelse for det vi på dansk normalt kalder sidens hoved og fod. Disse er normalt placeres udenfor tekstblokken.

Pagestyle

Det koncept som i L^AT_EX styrer hvordan sidehoved og -fod skal se ud.

Inline

Når noget står indeni noget andet (dansk: indlejret). Vi anvender det mest vedrørende matematik for at skelne mellem matematik skrevet blandt ordene i den almindelige tekst samt matematik som er fremhævet på linier for sig selv.

Klasse

L^AT_EXs pendant til skabeloner.

Pakke

Samling af makroer som udvider funktionaliteten i L^AT_EX, eller som omkonfigurerer dele af den.

Editor

Program man anvender til at skrive tekst med.

Teksteditor

Se *editor*.

Typesetting System

Dette er den engelske betegnelse for hvad L^AT_EX faktisk er. Det danske ord tekstbehandlingssystem passer bedre med den engelske betegnelse *Word processor*. Udfra den engelske betegnelse kan man lære at det L^AT_EX gør er det samme som trykkere gjorde i 'gamle' dage når de skulle sætte bøger op. L^AT_EX gør bare dette elektronisk.

Cropping

Er en fordanskning for den engelske betegnelse for at beskære et billede eller lignende.

Crop marks

Dette er noget man kan tilføje sit dokument for at markere hvor papiret skal beskæres. Dette anvendes oftest når papirstørrelsen er mindre end f.eks. A4.

Alignment

Kan oversættes med »*justering på en linie*«, i vores tilfælde er der oftest tale om en vertikal linie.

Letterspacing

Betegner afstanden mellem bogstaverne i et ord. Normalt er denne nul, men til visse effekter sættes den op.

E. Kommentarer

Prefix

Dette er noget man skriver før noget andet. De filer som vi anvender til at lave de eksterne eksempler med hedder alle sammen noget med `ext-master-`, dette er deres *prefix*.

Postfix

Svarende til *prefix* bare skrevet i slutningen.

Option

Engelsk ord for valgmulighed, skriver ofte det

engelske ord fordi det er kortere.

Upright

Vi anvender betegnelsen *upright* vedrørende skrifttyper til at betegne det som ikke er kursiv. En god dansk oversættelse er *opretstående*, men da den tilsvarende L^AT_EX-kommando er `\textup` anvender vi oftest det engelske *upright*.

Sub- eller superscript

Et andet ord for hhv. sænket eller hævet skrift.

E.8 Pakker som *ikke anbefales*

Dette er en liste over pakker som vi ikke anbefaler brugen af. Enden fordi pakken ikke distribueres mere eller fordi der er kommet en ny pakke som er smartere. Er man glad for at bruge en pakke af ældre dato og gerne vil fortsætte med det, så gør man naturligvis bare det.

subfigure

er overhalet af `subfig`-pakken. Husk også at visse klasser har deres egne systemer til subnummerering af delfigurer.

tlenc

brug i stedet `\usepackage[T1]{fontenc}`, om ikke andet så er det noget klarere at se hvad der sker. Desuden er det nemmere at skifte font encoding hvis det er nødvendigt.

isolatin1

pakken distribueres slet ikke mere. Brug `\usepackage[latin1]{inputenc}`, så er det også hurtigt at skifte til `utf8`-encoding om nødvendigt.

a4 brug `a4paper`-klasseoption i stedet. Det er der flere pakker som kan drage nytte af.

a4wide

lær selv at sætte marginerne op. Så er de nemmere at rette bagefter.

glossary

brug `glossaries` i stedet. Langt mere konfigurerbar.

numprint

funktionalitet overtaget af `siunitx`

sistyle

funktionalitet overtaget af `siunitx`

siunit

funktionalitet overtaget af `siunitx`

epsfig

selv om syntaksen er fuldt kompatibel med `graphicx`, så anbefales det at man anvender `graphicx` og `\includegraphics` for en mere standardiseret tilgang til ekstern grafik.

psfig

distribueres ikke mere, så gamle dokumenter som anvender `psfig` vil ikke kunne kompileres med de nye L^AT_EX-distributioner. Har man anvendt `\psfig` syntaksen i et gammelt dokument, så kan man få dokumentet til at oversætte igen ved at erstatte `\usepackage{psfig}` med `\usepackage{epsfig}`.

Pakker som ikke anbefales

latexsym

pakken er irrelevant, og tåbelig da den optager et helt matematikalfabet for bare 11 symboler (L^AT_EX kan kun udnyttet et begrænset antal matematikalfabeter).

ae irrelevant i dag hvor vi har ordentlige fontsæt. Da den blev skrevet tilbage i 90'erne var der en del problemer med nogle af standardfontene.

aecompl

samme grund som for ae

times

sætter ikke matematikfonten, og skalerer sans serif fonten forkert. Hvis man absolut vil bruge times⁵ så anvend pakken mathptmx.

mathptm

brug mathptmx i stedet.

pslatex

virker lidt som mathptmx men anvender en meget smal Courier.

palatino

mangler matematik, brug mathpazo eller kombiner i det mindste palatino med eulervm.

⁵ Anbefales ikke

Skabeloner

Dette kapitel skulle gerne ende op med at indeholde referencer til skabeloner til forskellige opgaver, såsom bachelorprojekter, specialer, ph.d.-afhandlinger, undervisningsmaterialer.

Indtil videre er der ikke rigtigt inkluderet nogen relevante skabeloner, men de skal nok komme.

Da vi gerne vil kommentere skabelonerne og da visse af skabelonerne er splittet op i flere filer, har vi valgt at skabelonerne kun skal kunne downloades, de vil *ikke* blive gengivet her i bogen. Vi vil i stedet forklare lidt om hver skabelon, måske forklare nogle specielle detaljer, men ellers lade kommentarerne i skabelonerne tale for sig selv.

Alle eksemplerne fra bogen samt alle skabelonerne er at finde via

<http://data.imf.au.dk/system/latex/bog/eksempler/>

Kommentarerne vil i alle skabelonerne være på engelsk, dette gør er skabelonerne er encoding neutrale, hvorved skabelonerne fungerer i alle editorer til at starte med. Man skal altså ikke rode med at få æøå til at virke.¹

Vore skabeloner vil primært være baseret på *memoir*-klassen.

URL: data.imf.au.dk/cgi-bin/latexbog/eksempler.cgi?action=single&file=S-F.1

Skabelon

F.1



Kommentar: Meget simpel skabelon bygget over *memoir*-klassen. Er velegnet som udgangspunkt for en aflevering med matematik.

Feature: Skabelonen er sat op således at hvis dokumentet kun fylder én side, så fjernes sidetallet automatisk (kræver mindst to oversættelser).

URL: data.imf.au.dk/cgi-bin/latexbog/eksempler.cgi?action=single&file=S-F.2

Skabelon

F.2



Kommentar: Stort set det samme som Skabelon F.1, men her demonstrerer vi hvordan man kan smide metadata op i sidehovedet for at spare plads.

¹ Dette kan fortsat være et problem med mange nuværende editorer til Windows.

Litteratur og pakkeinformation

CTAN, kort for *Comprehensive T_EX Archive Network* er et samarbejde hvor pakkeforfattere og andre placerer deres filer så andre kan få glæde af dem. Adressen er <http://www.ctan.org>.

Det er her pakkerne i MiK_TE_Xs og T_EX Lives pakkemanagere stammer fra, og kun herfra. Bemærk dog at det ikke er alle pakker man kan få gennem en pakkemanager. Der kan være licens konflikter eller forfatterens eget ønske, som sætter en begrænsning på hvorvidt en pakke må tages med. Disse pakker må man så installere pr. håndkraft.

De fleste af URLerne herunder er på formen CTAN: $\langle sti \rangle$, dette skal oversættes til <http://mirror.ctan.org/⟨sti⟩>. I PDF-udgaven er der dog tale om et aktivt link med den rette adresse. Man anvender mirrors for at mindske trykket på den centrale CTAN server, hvorpå stien også er en smule anderledes.

AMS, *User's Guide for the **amsmath** Package*, American Mathematical Society, 1999.
Følger med enhver L^AT_EX distribution under navnet `amsl`.

CTAN: `/macros/latex/required/amslatex/math/`

AMS, *Using the **amsthm** Package*, American Mathematical Society, 2. udgave, 2004.
Følger normalt med L^AT_EX distributionen som `amsthdoc.pdf`.

CTAN: `/macros/latex/required/amslatex/classes/`

Donald Arseneau, *placeins.sty*, 1999. Pakken som giver `\FloatBarrier`, lidt brugerinformation findes i `placeins.sty`. Der er også kommet en simpel brugermanual »`placeins.txt`« på CTAN.

CTAN: `/macros/latex/contrib/placeins/`

Donald Arseneau, *truncate.sty*, 2001. Denne pakke giver forskellige muligheder for automatisk at forkorte tekst, f.eks. i et sidehoved. Pakken findes via CTAN og brugervejledningen findes i starten af `truncate.sty`.

CTAN: `/macros/latex/contrib/misc/`

Donald Arseneau, *cite.sty*, 2003a. Pakken `cite` giver bl.a. sammentrækning af numeriske citeringer. Som altid, ved denne pakkeforfatter, finder man brugsvejledningen i slutningen af filen `cite.sty`.

CTAN: `/macros/latex/contrib/cite/`

Donald Arseneau, *framed.sty*, 2003b. Bemærk iøvrigt at hele denne pakke er bygget direkte ind i `memoir`-klassen.

CTAN: `/macros/latex/contrib/misc/`

Donald Arseneau, *threeparttable.sty*, 2003c. Brugervejledningen står i selve filen.

CTAN: `/macros/latex/contrib/misc/`

Donald Arseneau, *url.sty*, 2003d. Brugervejledningen til denne pakke findes i toppen (og bunden) af filen `url`. Pakken er en standard del af enhver L^AT_EX distribution.

CTAN: `/macros/latex/contrib/misc/`

Donald Arseneau, *varwidth.sty*, 2003e. Giver et *minipage*-lignende environment som selv justerer sin bredde ned til at passe med indholdet.

CTAN: `/macros/latex/contrib/misc/`

Litteratur og pakkeinformation

- Donald Arseneau, *chapterbib.sty*, 2009. Pakke som gør det muligt at lave litteraturlister for kapiptler. Nyttigt for ph.d.-afhandlinger samt f.eks. til proceedings. Brugermanualen ses i selve .sty-filen. CTAN: [/macros/latex/contrib/cite/](#)
- Jens Berger, *The jurabib package*, 2004. Pakke til at lave litteraturlister indenfor f.eks. humaniora, baseret på BibTEX. Se desuden <http://www.berger-on.net/jurabib/> hvor man vil finde mange forskellige eksempler på brugen af jurabib. CTAN: [/macros/latex/contrib/jurabib/](#)
- Javier Bezos, *The titlesec and titletoc Packages*, 2002. Brugervejledning til især titlesec-pakken. CTAN: [/macros/latex/contrib/titlesec/](#)
- Javier Bezos, *Customizing lists with the enumitem package*, 2011. Brugermanual for enumitem-pakken. Gældende for version 3+. CTAN: [/macros/latex/contrib/enumitem/](#)
- Preben Blæsild og Jørgen Granfeldt, *Statistics with applications in biology and geology*, Chapman & Hall/CRC, 2003.
- Johannes Braams, *Creating a mailing*, 1994. Simpel pakke til at lave næsten enslydende breve. CTAN: [/macros/latex/contrib/ mailing/](#)
- Johannes Braams, *Babel, a multilingual package for use with L^AT_EX's standard document classes*, 2005. Bruger og implementationsmanual for \babel-pakken. CTAN: [/macros/latex/required/babel/](#)
- Johannes Braams, David Carlisle, Alan Jeffrey, Leslie Lamport, Frank Mittelbach, Chris Rowley og Rainer Schöpf, *The L^AT_EX 2_ε Sources*, 2009. Dette er den dokumenterede kildekode til L^AT_EX-kernen. Den følger normalt med L^AT_EX-installationen. På en T_EX Live-baseret installation kan man se den via »texdoc source2e«, eller man kan anvende linket herunder. CTAN: [/macros/latex/doc/source2e.pdf](#)
- Robert Bringhurst, *The Elements of Typographic Style*, Hartley & Marks, 2002, 2. udgave. Version 2.5.
- David Carlisle, *The hline package*, 1994. Brugervejledning og dokumentation findes i hline.pdf. CTAN: [/macros/latex/required/tools/](#)
- David Carlisle, *The afterpage package*, 1995. Pakken er en standard del af L^AT_EX og giver mulighed for af indsætte noget som det aller første på næste side. CTAN: [/macros/latex/required/tools/](#)
- David Carlisle, *blkarray.sty*, 1999a. Pakke til at give specielle array og tabel konstruktioner. Forklaring vedrørende selve brugen findes i blkarray.sty. Spøjt nok kan man faktisk få en fin formateret version hvis man kører L^AT_EX på blkarray.sty og bare trykker Enter ved alle fejlene. CTAN: [/macros/latex/contrib/carlisle/](#)
- David Carlisle, *Packages in the 'graphics' bundle*, 1999b. Følger med enhver L^AT_EX distribution under navnet grfguide. Indeholder f.eks. information om brugen af color- og graphicx-pakken. CTAN: [/macros/latex/required/graphics/](#)
- David Carlisle, *The keyval package*, 1999c. Pakke til at give *key = value* option syntaks. For at få dokumentationen må man kompilere keyval.dtx. Se også pakken xkeyval. CTAN: [/macros/latex/required/graphics/](#)
- David Carlisle, *The tabularx package*, 1999d. Pakken er en standarddel af enhver L^AT_EX installation. CTAN: [/macros/latex/required/tools/](#)

Litteratur og pakkeinformation

- David Carlisle, *The colortbl package*, 2001a. Pakken giver support for farver i tabeller.
CTAN: [/macros/latex/contrib/colortbl/](#)
- David Carlisle, *The ifthen package*, 2001b. Er en standard del af enhver L^AT_EX-installation, med dokumentation `ifthen.pdf`.
- David Carlisle, *The longtable package*, 2004. Pakken er den standard del af L^AT_EX distributionerne. Manualen hedder normalt `longtable.pdf`.
CTAN: [/macros/latex/required/tools/](#)
- David Carlisle, *The xr package*, 2006. Pakke som gør det muligt at referere til labels i eksterne dokumenter. Pakken er den del af L^AT_EX-tools bundtet og derfor en del af enhver L^AT_EX-installation.
CTAN: [/macros/latex/required/tools/](#)
- Chicago, *The Chicago manual of style*, The University of Chicago Press, 15. udgave, 2003.
- Steven Douglas Cochran, *The subfig Package*, 2004. Pakken anvendes til at angive subnummerering af figurer, samt subcaptions. Pakken `subfig` afløser den tidligere `subfigure` pakke.
CTAN: [/macros/latex/contrib/subfig/](#)
- Patrick W. Daly, *Customizing Bibliographic Style Files*, 2003a. Dokumentationen til programmet `makebst`. Dokumentationen fortæller ikke så meget om hvordan programmet anvendes, se i stedet [Mittelbach og Goossens \(2004\)](#) afsnit 13.5.2. Det er desuden en god ide at kigge på filen `merlin.mbs` som er driverfilen til `custom-bib` og `makebst`. Kompileres denne fil får man yderligere forklaring om brugen.
CTAN: [/macros/latex/contrib/custom-bib/](#)
- Patrick W. Daly, *Natural Sciences Citations and References*, 2003b. Manual til pakken `natbib`, følger normalt med L^AT_EX distributionen, men kan ellers findes via CTAN. Brugermanualen fås ved at kompilere `natbib.dtx`.
CTAN: [/macros/latex/contrib/natbib](#)
- Marco Daniel og Elke Schubert, *The mdframed pacakge*, 2010.
- Michael Downes, *Short Math Guide for L^AT_EX*, American Mathematical Society, 2002. Kort introduktion til `amsmath`.
URL: <http://www.ams.org/tex/amslatex.html>
- Victor Eijkhout, `comment.sty`, 1999. Brugerguiden til denne pakke findes i form af kommentarer i starten af `comment.sty`.
CTAN: [/macros/latex/contrib/comment/](#)
- Victor Eijkhout, *T_EX by Topic—A T_EXnician's Reference*, Addison-Wesley, 2001. Bogen er skrevet i 1991, men nu udsolgt fra forlaget. Forfatteren har i stedet overtaget alle rettigheder og har herefter stillet bogen frit til rådighed. Bogen er et godt alternativ hvis man ønsker at lære mere om T_EX.
URL: <http://www.eijkhout.net/tbt/>
- FAQ, *UK-TUG L^AT_EX FAQ*, 2007. FAQ over de almindeligste spørgsmål. FAQen kan desuden downloades som PDF.
URL: <http://www.tug.org/faq/>
- Simon Fear, *Publication quality tables in L^AT_EX*, 2003. Manual til pakken `booktabs` som hjælper med at give pænere tabeller. Findes som `booktabs.pdf`. Manualen er desuden værd at læse for dens diskussion af hvordan man designer tabeller. Funktionaliteten af pakken er desuden indbygget i `memoir`-klassen.
CTAN: [/macros/latex/contrib/booktabs/](#)

Litteratur og pakkeinformation

Melchior Franz, *The soul package*, 2003. Pakken soul kan give tekst hvor afstanden mellem bogstaverne er større end normalt, samt give en anden metode til understregning af tekst, hvor denne metode faktisk understøtter at teksten automatisk kan deles over flere linier.
CTAN: [/macros/latex/contrib/soul/](#)

Federico Garcia, *opcit a package for footnote-style bibliographic references*, 2006.
CTAN: [/macros/latex/contrib/opcit/](#)

Michel Goossens, Sebastian Rahtz og Frank Mittelbach, *The L^AT_EX Graphics Companion – Illustrating documents with T_EX and PostScript*, Addison-Wesley, 1997, ISBN 0-201-85469-4.

Michel Goossens, Frank Mittelbach, Sebastian Rahtz, Denis Roegel og Herbert Voß, *The L^AT_EX Graphics Companion*, Addison-Wesley, 2007, 2. udgave, ISBN 978-0-321-50892-8. Helt ny bog. Meget større og mere omfattende end den første version fra 1997. Blandt andet står der en del om METAPOST, PSTricks og X_Y-pic.

Michel Goossens, Frank Mittelbach og Alexander Samarin, *The L^AT_EX Companion*, Addison-Wesley, 1993, ISBN 0-201-54199-8.

Alexander Grahn, *The animate package*, 2008. Pakke som kan anvendes til at lave forskellige typer animationer til f.eks. præsentationer.
CTAN: [/macros/latex/contrib/animate/](#)

Michael Grant, David Carlisle og Craig Barratt, *psfrag.sty and psfrag.pro*, 1998. Pakken psfrag anvendes til at placere L^AT_EX kode i stedet for nøgleord i EPS-filer. Manualen hedder pfgguide.pdf.
CTAN: [/macros/latex/contrib/psfrag/](#)

Enrico Gregorio, *Horrors in L^AT_EX: How to misuse L^AT_EX and make a copy editor unhappy*. *TUGboat*, 26(3), 2005. Artikel med en masse rædselseksemples, god lektie.
URL: <http://tug.org/TUGboat/Contents/contents26-3.html>

Indian T_EX Users Group, *L^AT_EX Tutorials—A Primer*, 2003.
URL: <http://www.tug.org.in/tutorials.html>

Thorsten Hansen, *The multibib Package*, 2004. Pakke til at lave emneopdelt litteraturliste.
CTAN: [/macros/latex/contrib/multibib/](#)

Patrick Happel, *lipsum – access to 150 paragraphs of Lorem ipsum dummy text*, 2005. Pakken giver adgang til en masse fyld tekst som man frit kan benytte.
CTAN: [/macros/latex/contrib/lipsum/](#)

Harald Harders, *Multilingual bibliographies: The babelbib package*, 2009. Pakke som i modsætning til de fleste BibT_EX-stilfiler giver brugeren mulighed for at styre specialordene.
CTAN: [/biblio/bibtex/contrib/babelbib/](#)

Emil Hedevang, *Serres spektralfølge: med henblik på opståen og opbygning*, Bachelorprojekt, Institut for Matematiske Fag, Det Naturvidenskabelige Fakultet, Aarhus Universitet, 2004. Forefindes ved henvendelse på bibliotekskontoret samme sted. Det specielle ved denne er at den faktisk er skrevet i plain T_EX, hvorimod specialet (Hedevang, 2006) er skrevet i L^AT_EX, via memoir-klassen. Flot bachelorprojekt som viser hvad man kan opnå hvis man tager sig tiden til det.

Emil Hedevang, *Homologi af dikubestier: et forsøg på homologi af kombinatoriske monotone stier med henblik på anvendelse af algebraisk topologiske metoder inden for samtidigt*

Litteratur og pakkeinformation

- samvirke*, Speciale, Institut for Matematiske Fag, Det Naturvidenskabelige Fakultet, Aarhus Universitet, 2006. Forefindes ved henvendelse på bibliotekskontoret samme sted. Som ved bachelorprojektet ([Hedevang, 2004](#)) af samme forfatter, et godt eksempel på et gennemført design.
- Carsten Heinz, *The listings Package*, 2004. Pakken `listings` er uundværlig, når det kommer til kildekode fremvisning. Understøtter et hav af forskellige programmeringsprog.
CTAN: [/macros/latex/contrib/listings/](#)
- Martin Hensel, *The mhchem Bundle*, 2005. Gældende for version 2.
CTAN: [/macros/latex/contrib/mhchem/](#)
- John D. Hobby et al., *METAPOST – a user’s manual*, 2011.
CTAN: [/graphics/metapost](#)
- Morten Høgholm, *Empheq—Emphasized Equations*, 2008a. Dokumentationen til `empheq` pakken.
CTAN: [/macros/latex/contrib/mh/](#)
- Morten Høgholm, *Mathtools*, 2008b. Dokumentationen til `mathtools`-pakken.
CTAN: [/macros/latex/contrib/mh/](#)
- Klaus Höppner, *Introduction to MetaPost*, i *TUGBoat – The Communications of the T_EX Users Group*, bind 30:2, pp. 209–213, The T_EX Users Group, 2009. Indtil 11. oktober 2010 er artiklen kun tilgængelig for medlemmer af TUG. Herefter kan findes findes via linket herunder.
URL: <http://tug.org/TUGboat/Contents/contents30-2.html>
- Szász János, *The algorithmicx package*, 2005. Brugermanualen for seneste version af `algorithmicx`-pakken.
CTAN: [/macros/latex/contrib/algorithmicx/](#)
- Jens Ledet Jensen, *Et Nanokursus i Statistik*, Institut for Matematiske Fag, 2007. Statistikbog for Nano-studerende, designet og sat op af Lars Madsen. Bogen kan bruges som illustration af: Brugen af Utopia og Fourier tekst- og matematikfontene, brugen af dækblad og forside samt brugen af et lidt andet design til forsiden på omslaget. Bogen kan ses/købes i Stakbogladen, Naturvidenskab, Aarhus Universitet.
- Arne Jørgensen, *Dk-bib – danske varianter af BibT_EXs standardstilarter*, 2006. Gældende for version 0.6 maj 2006.
CTAN: [/biblio/bibtex/contrib/dk-bib/](#)
- Manuel Kauers, *gauss.sty – A Package for Typesetting Matrix Operations*, 2002. Udover pakkemanualen man finder i linket her under, kan man tage et kig i Herbert Voß’s artikel om pakken `gauss`, artiklen er godt nok på tysk, men eksemplerne er gode. Artiklen findes her: [CTAN: /info/math/voss/gauss/](#).
CTAN: [/macros/latex/contrib/gauss/](#)
- Uwe Kern, *Extending L^AT_EX’s color facilities: the xcolor package*, 2006. Pakke som giver udvidet farvesupport i L^AT_EX, i.e. flere muligheder end `color`-pakken.
CTAN: [/macros/latex/contrib/xcolor/](#)
- Donald E. Knuth, *The T_EXbook*, Addison-Wesley, 1986, ISBN 0-201-13447-0.
- Markus Kohm og Jens-Uwe Morawski, *KOMA-Script – a versatile L^AT_EX 2_ε bundle*, 2006. Bruger vejledning til KOMA-Script klasserne. Manualen findes ofte under navnet `scrguien.pdf`.
CTAN: [/macros/latex/contrib/koma-script/](#)
- Jørgen Larsen, *IMFUFA-L^AT_EX*, 2010. IMFUFA-L^AT_EX er nogle forskellige dansk udviklede dokumentklasser baseret på *memoir*-klassen. Klasserne indeholder desuden andre små guldkorn som gør dansk L^AT_EX nemmere.
URL: <http://dirac.ruc.dk/imfufalatem/>

Litteratur og pakkeinformation

- John Lavagnino, *endnotes.sty*, 2003. Pakken giver muligheden for endnotes, hvilket er det samme som fodnoter, men som i stedet printes samlet til slut. Brugervejledning findes som kommentarer i sterten af *endnotes.sty*.
CTAN: [/macros/latex/contrib/misc/](#)
- Philipp Lehman, *The Font Installation Guide*, Findes på CTAN, 2004. Fremragende eksempel på at en teknisk manual sagtens kan se pæn ud.
CTAN: [/info/Type1fonts/fontinstallationguide/](#)
- Philipp Lehman, *The csquotes package*, 2006. Pakke til intelligente anførselstegn. Versionen 3.4 er fra april 2006.
CTAN: [/macros/latex/contrib/csquotes/](#)
- Philipp Lehman, *The etoolbox package*, 2009. Pakke som [Lehman \(2010\)](#) er bygget over. Giver mange spændende værktøjer til makroskribenter.
CTAN: [/macros/latex/contrib/etoolbox/](#)
- Philipp Lehman, *The biblatex package*, 2010. Version 0.9, og nu med i de nyeste L^AT_EX distributioner. Helt ny implementation af måden man håndterer litteraturlister og citeringer.
CTAN: [/macros/latex/exptl/biblatex/](#)
- Jerry Leichter og Piet van Oostrum, *multirow.sty*, 2004. Version 1.6. Brugerinformation findes i starten af *multirow.sty*.
CTAN: [/macros/latex/contrib/multirow/](#)
- LTX3, *L^AT_EX 2_ε for class and package writers*, The L^AT_EX3 Project, 1999. Skrevet af L^AT_EX3 projektet og følger med enhver L^AT_EX distribution under navnet *clsguide.pdf*.
CTAN: [/macros/latex/doc/](#)
- Lars Madsen, *Memoir-klassen – en stor og fleksibel dokumentklasse*, 2003. DK-TUG-foredrag om *memoir*-klassen. Slides (med kildekoder) kan findes via DK-TUGs hjemmeside.
URL: <http://www.tug.dk/memoir.html>
- Lars Madsen, *Avoid eqnarray! The PracT_EX Journal*, 4, 2006.
URL: <http://tug.org/pracjourn/2006-4/madsen/>
- Lars Madsen, *Users manual for the SASdisplay package*, 2007. Simpel lille pakke til at fremvise SAS kode eller output via pakken *listings*. Version 0.81, af september 2007.
URL: <http://www.imf.au.dk/system/latex/lokalepakker/>
- Lars Madsen, *Page Styles on steroids (or, memoir makes page styling easy)*. *The PracT_EX Journal*, 2, 2008a. Artiklen er en gennemgang af page style håndteringen i *memoir*-klassen, specielt forklarer den det nye interface som gør konfigurering af sektionmarkører til en leg.
URL: <http://tug.org/pracjourn/2008-2/madsen/>
- Lars Madsen, *Various chapter styles for the memoir class*, 2008b. Showcase dokument som fremviser forskellige kapitelforside designs til brug ved *memoir*-klassen. Filen (med kildekode) er tilgængelig via CTAN. Bidrag er velkomne. Seneste version er 1.7b fra 2010/05/10.
CTAN: [/info/MemoirChapStyles/](#)
- Lars Madsen, *A small extension to threeparttable*, 2009. En lille udvidelse til [Arseneau \(2003c\)](#) som gør det muligt at anvende den sammen med *longtable*. Desuden kan den anvendes til at muliggøre referencer til tabelnoter.
CTAN: [/macros/latex/contrib/threeparttablex/](#)
- Nicolas Markey, *Tame the BeasT – The B to X of BibT_EX*, 2009. Note som beskriver sproget anvendt til BibT_EX-stilfiler. Filen findes sikkert allerede i de fleste L^AT_EX-distributioner som *tbt_en.pdf*.
CTAN: [/info/bibtex/tamethebeast/](#)

Litteratur og pakkeinformation

Andreas Matthias, *The pdfpages Package*, 2004. Pakke som gør det muligt at inkludere fler-sidede PDF-dokumenter, direkte i ens eget dokument. Anvendes bl.a. på `\imf` til at lime preprints sammen med deres respektive forsider.

CTAN: [/macros/latex/contrib/pdfpages/](#)

Wolfgang May, *An Extension of the L^AT_EX-Theorem Environment*, 2002. Brugermanual for pakken `ntheorem`.

CTAN: [/macros/latex/contrib/ntheorem/](#)

Andrew Mertz og William Slough, *A TikZ tutorial: Generating graphics in the spirit of T_EX*, i *TUGboat – The Communications of the T_EX Users Group*, bind 30:2, pp. 214–226, The T_EX Users Group, 2009. Indtil 11. oktober 2010 er artiklen kun tilgængelig for medlemmer af TUG. Herefter kan findes findes via linket herunder.

URL: <http://tug.org/TUGboat/Contents/contents30-2.html>

Henrik Skov Midtby, *The todonotes package*, 2009. Pakke til at lave meget grafiske todo noter til sit dokument. Det er en god ide at læse manualen før man bruger pakken.

CTAN: [/macros/latex/contrib/todonotes/](#)

Frank Mittelbach, *The varioref package*, 2004. Pakken giver `\vref` makroen til at lave lidt smartere krydsreferencer. Burde følge med enhver L^AT_EX installation, men kan ellers findes på CTAN.

CTAN: [/macros/latex/required/tools/](#)

Frank Mittelbach og David Carlisle, *A new implementation of L^AT_EX's tabular and array environment*, 2003. Brugermanual og dokumentation for pakken `array`. Pakken følger med enhver L^AT_EX distribution. Manualen hedder normalt `array.pdf`.

CTAN: [/macros/latex/required/tools/](#)

Frank Mittelbach og Michel Goossens, *The L^AT_EX Companion*, Addison-Wesley, 2004, 2. udgave, ISBN 0-201-36299-6. Med Johannes Braams, David Carlisle og Chris Rowley samt bidrag af Christine Detig og Joachim Schrod. Kildekoden til mange af eksemplerne i bogen findes på CTAN.

CTAN: [/info/examples/tlc2/](#)

Lee Netherton og CV Radhakrishnan, *nomencl A Package to Create a Nomenclature*, 2005. Pakke til at lave tegn- eller terminologiforklaringer.

CTAN: [/macros/latex/contrib/nomencl](#)

Rolf Niepraschk, *Pakken overpic*, 1997. Pakke som giver mulighed for at skrive ovenpå inkluderede figurer. Pakken er dokumenteret via to eksempelfiler som man kan finde via URLen her under.

CTAN: [/macros/latex/contrib/overpic/](#)

Rolf Niepraschk, *The eso-pic package*, 2002. Pakken findes på CTAN, hvor man også finder eksempler samt manualen.

CTAN: [/macros/latex/contrib/eso-pic/](#)

Josselin Noirel, *The cellspace package*, 2006. Pakke med konstruktioner til at give mere luft til cellerne i en tabel.

CTAN: [/macros/latex/contrib/cellspace/](#)

Josselin Noirel, *The xifthen package*, 2007. Udvidelser til `ifthen`-pakken (Carlisle, 2001b).

CTAN: [/macros/latex/contrib/xifthen/](#)

Heiko Oberdiek, *The grffile package*, 2009. Pakke som udvider reglerne for hvordan filnavne for grafik kan skrives. F.eks. kan den håndtere filnavne med mellemrum.

CTAN: [/macros/latex/contrib/oberdiek/](#)

Tobias Oetiker, *An Acronym Environment for L^AT_EX2_ε*, 2005.

CTAN: [/macros/latex/contrib/acronym/](#)

Litteratur og pakkeinformation

- Tobias Oetiker, *The Not So Short Introduction to L^AT_EX 2_ε*, 2006. Introduktion til L^AT_EX som mange har lært L^AT_EX fra. Bemærk dog at en stor del af matematikken i denne note ikke bør anvendes. [CTAN: /info/lshort/english/](#)
- Scott Pakin, *The Comprehensive L^AT_EX Symbol list*, 2003. Følger normalt med LaTeX distributionerne under navnet »symbols-a4.pdf« eller »symbols-letter.pdf«, ellers kan den findes via CTAN. [CTAN: /info/symbols/comprehensive/](#)
- Paul Pichaureau, *The mathdesign package*, 2006. Pakke med opsætning af matematik til visse fonte. URL: <http://www.tug.org/tex-archive/fonts/mathdesign/>
- Ebbe Thue Poulsen, *Funktioner af en og flere variable. Indledning til matematisk analyse*, Gads Forlag, 2001.
- Sebastian Rahtz og Heiko Oberdiek, *Hypertext marks in L^AT_EX: a manual for hyperref*, 2004. Manual for hyperref pakken til interne hyperlinks i PDF. Desværre hedder manualen ofte manual.pdf. [CTAN: /macros/latex/contrib/hyperref/](#)
- Keith Reckdahl, *Using Imported Graphics in L^AT_EX 2_ε*, 3. udgave, 2005. Da kildekoden ikke er inkluderet, er filen ikke inkluderet i f.eks. T_EX Live. Hent den i stedet fra linket herunder. [CTAN: /info/epslatex/english/](#)
- Axel Reichert, *Fancy Cross-referencing*, 1999. Manual til pakken fancyref, som tilbyder en mere intelligent måde at referere på. Hvis dokumentationen ikke findes på systemet skal man bare finde og kompilere »fancyref.dtx«.
[CTAN: /macros/latex/contrib/fancyref/](#)
- R. M. Ritter, *The Oxford Guide to Style*, Oxford University Press, 2002, ISBN 0-19-869175-0.
- Kristoffer H. Rose, *X_y-pic User's Guide*, 1999. Findes sammen med pakken xy under navnet xyguide.pdf. [CTAN: /macros/generic/diagrams/xy-pic/xy-3.7/doc/](#)
- Kristoffer H. Rose og Ross Moore, *X_y-pic Reference Manual*, 1999. Findes sammen med pakken xy under navnet xyrefer.pdf. Kan være meget svær at forstå.
[CTAN: /macros/generic/diagrams/xy-pic/xy-3.7/doc/](#)
- Bernd Schandl, *paralist – Extended List Environments*, 2002.
[CTAN: /macros/latex/contrib/paralist/](#)
- Martin Scharrer, *Version Control of LaTeX Documents with svn-multi. The PracT_EX Journal*, 3, 2007, ISSN 1556-6994. URL: <http://www.tug.org/pracjourn/2007-3/scharrer>
- Martin Scharrer, *The svn-multi package*, 2009. Pakke som giver adgang til versionsdata indsat af subversion versionskontrollsystemet. [CTAN: /macros/latex/contrib/svn-multi/](#)
- Walter Schmidt, *The icomma package for L^AT_EX 2_ε*, 2002. Giver et aktivt komma til decimaltal i matematik. Dokumentationen fås ved at oversætte »icomma.dtx«.
[CTAN: /macros/latex/contrib/was/](#)
- Martin Schröder, *ragged2e.sty*, 2003. Brugervejledningen findes indeni »ragged-2e.sty«.
[CTAN: /macros/latex/contrib/ms/](#)
- Axel Sommerfeldt, *Typesetting captions with the caption package*, 2005. Seneste version findes på CTAN. [CTAN: /macros/latex/contrib/caption/](#)

Litteratur og pakkeinformation

- Victoria Squire, *Getting it Right with Type*, Laurence King Publishing, 2006, ISBN 1-85669-474-7.
- Ellen Swanson, *Mathematics into Type*, American Mathematical Society, 1999, ISBN 0-8218-1961-5. Hæftet er af ældre dato men opdateret i 1999 af Arlene O'Sean og Antoinette Schleyer. Faktisk er dette hæfte citeret i [Knuth \(1986\)](#), side 197.
- Nicola L.C. Talbot, *Datatool – Databases and data manipulation*, 2009a. Pakke som f.eks. gør det muligt at hente formaterede tabeller fra eksterne kilder. Pakken er veldokumenteret, med masser af eksempler. [CTAN: /macros/latex/contrib/datatool/](#)
- Nicola L.C. Talbot, *glossaries.sty v 2.04: L^AT_EX Package to Assist Generating Glossaries*, 2009b. Meget gennemført pakke til generering af begrebsforklaringer etc. Manualen er velskrevet og pakken kommer med mange eksempel filer man kan tage udgangspunkt i. Hvis man anvender MikTeX, vil det være en fordel at installere Perl for at få det fulde udbytte af pakken. [CTAN: /macros/latex/contrib/glossaries/](#)
- Till Tantau, *Users's Guide to the Beamer Class*, 2005. Bruger manual til Beamer klassen. Filen hedder normalt »beameruserguide.pdf«. Se desuden [CTAN: /macros/latex/contrib/beamer/examples/](#). [CTAN: /macros/latex/contrib/beamer/doc/](#)
- Till Tantau, *The TikZ and PGF packages*, 2007. Manual til det grafiksystemet TikZ/PGF som giver samme resultat i PostScript og PDF. De grafiske muligheder fra dette system anvendes bl.a. i *beamer*-klassen. Manualen findes via [CTAN: /graphics/pgf/base/doc/generic/pgf/version-for-pdftex/en/](#), pakkerne kan findes via linket herunder. Man kan desuden findes man eksempler ved at google »pgf examples«. [CTAN: /graphics/pgf/](#)
- Kresten Krab Thorup og Frank Jensen, *The calc package – Infix notation arithmetic in L^AT_EX*, 1998. Standard del af enhver L^AT_EX installation. Giver f.eks. mulighed for at man kan lægge længder sammen i argumentet til `\setlength`. Pakken vedligeholdes i dag af L^AT_EX3-projektet. [CTAN: /macros/latex/required/tools/](#)
- Geoffrey Tobin, *setspace.sty*, 2000. Denne pakke giver forskellige muligheder til generering af tekst med dobbelt linieafstand. Brugervejledningen for pakken findes som kommentarer i »setspace.sty«. Bemærk at funktionaliteten af pakken nu er indbygget i *memoir*-klassen. [CTAN: /macros/latex/contrib/setspace/](#)
- Mark Trettin, *An essential guide to L^AT_EX 2_ε usage – Obsolete commands and packages*, 2007. Engelsk overstættelse ved Jürgen Fenn. [CTAN: /info/l2tabu/english/](#)
- Stefan Ulrich, *bibtopic.sty*, 2006. Pakke til at lave emneopdelt litteraturliste. [CTAN: /macros/latex/contrib/bibtopic/](#)
- Hideo Umeki, *The geometry package*, 2002. Pakken giver nemt interface til konfigurering af marginer i L^AT_EX. Alt efter hvilken version af pakken man har kan manualen til denne pakke godt hedde »manual.pdf«, hvilket ikke er særligt sigende. [CTAN: /macros/latex/contrib/geometry/](#)
- Universitetsavisen for Aarhus Universitet, *Campus*, 2006. Nummer 10, 2006, indeholder en artikel om normalsider. URL: <http://www.au.dk/campus/avis/2006/arkiv06/>
- Universitetsavisen for Aarhus Universitet, *Campus*, 2007. Nummer 19, 2007, indeholder en diskussion vedrørende brugen af PowerPoint og lignende i undervisningen. URL: <http://www.au.dk/campus/avis/2007/arkiv07/>

Litteratur og pakkeinformation

- Dominique P. G. Unruh, *usc.sty – Unicode Support*, 2004.
CTAN: [/macros/latex/contrib/unicode/](#)
- Piet van Oostrum, *Page layout in L^AT_EX*, 2004. Brugermanual for pakken fancyhdr. Manualen er meget lærerig og tilbyder mange løsninger på sidehoved og -fod relaterede problemer.
CTAN: [/macros/latex/contrib/fancyhdr/](#)
- Timothy Van Zandt, *The ‘fancyvrb’ package – Fancy Verbatims in L^AT_EX*, 1998.
CTAN: [/macros/latex/contrib/fancyvrb/](#)
- Didier Verna, *Fixme – Collaborative annotation tool for L^AT_EX*, 2009. Hørende til version 4.1. Bemærk at version 4 ændrede en del af syntaksen i forhold til tidligere versioner, så husk at læse manualen til pakken.
CTAN: [/macros/latex/contrib/fixme/](#)
- Ulrik Vieth, *The concmath package*, 1999. Udvidelse af Concrete fonten. Brugermanualen fås ved at oversætte filen contmath.dtx som kan findes på URLen herunder.
CTAN: [/macros/latex/contrib/concmath/](#)
- Jürgen Vollmer, *The draftcopy package*, 2002. Manualen (hvis den mangler) kan findes på CTAN under navnet draftcopy.doc, som så bare skal kompileres med L^AT_EX.
CTAN: [/macros/latex/contrib/draftcopy/](#)
- Herbert Voß, *Math mode*, 2008. Stor note omkring matematikrelaterede ting i L^AT_EX, både med og uden amsmath.
CTAN: [/info/math/voss/mathmode/](#)
- Mark A. Wicks, *Dvipdfm User’s Manual*, 1999. Teknisk brugermanual til dvipdfm programmet. Burde følge med L^AT_EX distributionen som .
CTAN: [/dviware/dvipdfm/](#)
- Peter Williams og Thorsten Schnier, *The Harvard Family of Bibliography Styles*, 1996. Brugermanual til harvard pakken.
CTAN: [/macros/latex/contrib/harvard/](#)
- Peter Wilson, *The tocloft package*, 2003. Pakken omkonfigurerer indholdsfortegnelsen og giver brugeren mange konfigurations muligheder. Funktionaliteten af pakken er indbygget i *memoir*.
CTAN: [/macros/latex/contrib/tocloft/](#)
- Peter Wilson, *L^AT_EX for the Initiated – A Design Class*, Endnu ikke udgivet, 2004a. Bog om dokumentdesign med *memoir* klassen. Forhåbentlig bliver dette 500+ siders monster snart udgivet så andre også kan få glæde af den.
- Peter Wilson, *The tocbibind package*, 2004b. Pakken sørger for at punkter som bibliografi, indeks og indholdsfortegnelse indgår i indholdsfortegnelsen. Indbygget i *memoir*.
CTAN: [/macros/latex/contrib/tocbibind/](#)
- Peter Wilson, *Addendum – The Memoir Class for Configurable Typesetting; User Guide*, 2007a. Addendum til Memoir manualen. Seneste version fra december 2007.
CTAN: [/macros/latex/contrib/memoir/](#)
- Peter Wilson, *Some Examples of Title Pages*, 2007b. Mange eksempler på hvordan en forside kan se ud. Lavet med *memoir* klassen. I sidste del af dokumentet er koden til de forskellige eksempler gennemgået. Filen hedder normalt titlepages.pdf og kan findes på linket herunder.
CTAN: [/info/latex-samples/](#)
- Peter Wilson, *The printlen package*, 2009. Giver metoder til at skrive værdien af en længe ud i for skellige enheder.
CTAN: [/macros/latex/contrib/printlen/](#)

Litteratur og pakkeinformation

Peter Wilson, *The Memoir Class for Configurable Typesetting*, The Herries Press, 2010, 8. udgave. Dette er brugermanualen for *memoir*-klassen. Manualen følger normalt med installationen som `memman.pdf`, men kan desuden hentes fra CTAN.

CTAN: [/macros/latex/contrib/memoir/](#)

Joseph Wright, *siunitx – A comprehensive (SI) units package*, 2010. Pakken er under fortsat udvikling og JW svarer ofte på spørgsmål på CTT eller på <http://tex.stackexchange.com>. Bemærk at denne pakke er baseret på L^AT_EX₃-kode, denne ændres en gang i mellem fundamentalt, så når man opdaterer er det bedst at man opdaterer hele installationen og ikke bare en enkelt pakke. Pakke og manual kan findes via

CTAN: [/macros/latex/contrib/siunitx/](#)

Stikordsregister

Ikke-bogstaver

- ~
 - tie, non-breaking space, 9
 - (BIBTEX felt), 207
 - \: (medspace), 46
 - . (punktum, hegn)
 - som tomt hegn, 65, 67
 - \; (thickspace), 46
 - \, (thinspace), 46
 - \~ (accent tilde), 30
 - !
 - anvendt i indeks, 254
 - "
 - specialtegn under dansk babel, 25
 - "_
 - orddeling ved bindestreg, 27, 228
 - \" (umlaut), 30
 - "/
 - orddeling ved skråstreg, 228
 - „tekst“ (anførselstegn), 26
 - \#
 - argumentangiver, 9
 - \\$
 - matematikmode, 9
 - &, 48
 - ved brug i tabeller, 168
 - ,
 - apostrof, 8
 - ping, 8
 - ((hegn), 65
 -) (hegn), 65
 - *, 86
 - \' (accent aigu), 30
 - \. (accent prik over), 30
 - \- (orddeling), 24, 26, 227
 - / (hegn), 65
 - 10pt (klasse option), 11
 - 11pt (klasse option), 11
 - 12pt (klasse option), 11
 - _ (ordmelletrum), 46
 - \= (accent macron), 30
 - »tekst« (anførselstegn), 26
 - @
 - anvendt i index, 254
 - i makronavne, 358
 - [(hegn), 65, 90
 - \$. . \$, 45, 47
 - \%
 - kommentartegn, 9
 - \&
 - søjleadskillers, 9
 - \[. . . \], 47, 48, 50, 119
 - _
 - subscriptangiver, 9
 - { }
 - som beskyttelse, 31
 - \{ (hegn), 65
 - \} (hegn), 65
 -] (hegn), 65, 90
 - ^, 68
 - hævet skrift, 68
 - potens, 68
 - superscript, 68
 - superscriptangiver, 9
 - \^ (accent circumflex), 30
 - , 68
 - indices, 68
 - subscript, 68
 - sænket skrift, 68
 - '
 - accent-grave, 8, 9
 - back-ping, 8, 9
 - \' (accent grave), 30
 - 'tekst' (anførselstegn), 26
 - "tekst" (anførselstegn), 26
 - *, 98
 - \\, 48, 49, 71, 76, 91, 94, 98
 - |, 66, 254
 - { }
 - obligatorisk argument, 9
- ## A
- A (søjle specifier), 173
 - a4 (pakke, brug ikke), 416
 - a4paper (klasse option), 6, 11, 15, 334, 416
 - a4wide (pakke, brug ikke), 416
 - abbrv (BIBTEX stil), 209, 218
 - abbrvnat (BIBTEX stil), 212
 - \abovecaptionskip, 152
 - \abs, 103
 - lavet på den rigtige måde, 103
 - \abs*, 103
 - abstract (environment), 303, 321
 - \abstractname (sprogspecifikt navn), 230

- accenter, 29
- aigu (á), 30
 - bind efter (ô), 30
 - bolle (å), 30
 - breve (ă), 30
 - cedille (ç), 30
 - circumflex (â), 30
 - grave (à), 30
 - háček (č), 30
 - macron (ā), 30
 - matematiske, 63
 - ogonek (ą), 30
 - prik over (â), 30
 - prik under (ã), 30
 - tilde (ñ), 30
 - umlaut (ä), 30
 - understreg (ǎ), 30
 - ungarsk umlaut (ǎ), 30
- \acute, 63
- \add (gauss), 74
- \addcontentsline, 286, 324, 346, 347
- \addgls (glossaries), 261
- \addlinespace (booktabs), 175, 176, 185
- address (BibTeX felt), 205, 208
- \addto, 229
- \addtocontents, 346
- \addtocounter, 365
- \addtolength, 186, 315, 363, 364
- \addtopsmarks (memoir), 311, 312
- \AddToShipoutPicture, 156
- \adjustwidth, 320
- adjustwidth (env, memoir), 320, 321
- adjustwidth* (env, memoir), 191, 320
- Adobe Reader, 394
- \AE (Æ), 29, 30
- \ae (æ), 29, 30
- ae (pakke, brug ikke), 417
- aecompl (pakke, brug ikke), 417
- afgrænser, *Se* hegn
- afgrænsere, 65
- afledt af en funktion, 54
- \afterpage, 146, 147, 153, 193
- afterpage (pakke), 146, 412
- \aleph (ℵ), 53
- algorithm (environment), 285
- algorithm (pakke), 285
- algorithmic (environment), 283
- algorithmicx (pakke), 283–285, 425
- algpseudocode (pakke), 283
- \aliaspagestyle (memoir), 309, 310
- align (enumitem), 243
- align (environment), 47–51, 76, 90–92, 95–98, 101, 104, 127, 133, 275, 332, 404, 409
- align* (environment), 49, 50, 54, 64, 87–91, 93, 94, 97, 99, 104–106, 275, 282
- alignat (environment), 76, 91–93, 96, 105, 132, 133
- alignat* (environment), 92
- aligned (environment), 96, 97, 99, 101
- alignedat (environment), 89, 96, 98, 99
- alignment, 90
- align, 91
 - alignat, 92
 - aligned, 96
 - alignedat, 96
 - ensbetyderpile mellem linier, 105
 - hack ved brede grænser, 102
 - indsæt kommentar, 93
 - split, 95
 - til højre for relation, 91
- alignmentsøjle, 90
- \allowbreak, 85, 333
- \allowdisplaybreak, 98
- \allowdisplaybreaks, 98, 99
- \Alph, 366
- \alph, 366
- \Alph* (enumitem), 242
- \alph* (enumitem), 242
- \alpha (α), 52, 282
- alpha (BibTeX stil), 201, 209
- \alsoname (sprogspecifikt navn), 230
- alt for store hegn
- eksempel, 67, 77
- \ampere (siunitx enhed), 266
- amsart (klasse), 42, 45, 120, 335
- amsbook (klasse), 45, 120
- amsfonts (pakke), 51
- amsmath (pakke), 6, 12, 27, 45, 50, 51, 59, 71, 72, 74, 90, 95, 100–102, 104, 107, 120, 124, 125, 133, 178, 186, 231, 269, 282, 334, 366, 401, 412, 423, 430
- amssymb (pakke), 6, 12, 45, 51, 59, 75, 107, 125, 172, 231, 334, 413
- amsthm (pakke), 115, 117–122, 124, 131, 331, 334
- \and, 42
- anførselstegn, 25
- „tekst“, 26
 - »tekst«, 26
 - ‘tekst’, 26
 - “tekst”, 26
- \angle (∠), 53
- angle (graphicx), 138, 139
- \angstrom (siunitx enhed), 266
- animate (pakke), 277
- \animategraphics, 277
- annotate (BibTeX felt), 208
- ansinew (encoding), 14
- ansize (pakke), 413
- apalike (BibTeX stil), 211, 213, 216

- apostrof, 9
- appendices (environment), 224
- \appendix, 223, 224, 294, 322, 329
- \appendixname, 224
- \appendixname (sprogspecifikt navn), 230
- \appendixpage (memoir), 224, 322
- \appendixpagename (memoir), 224, 322
- \appendixtocname (memoir), 224, 322
- applemac (encoding), 14, 390
- \approx (\approx), 53
- Aquamacs, 391
- \ar (X_y-pic), 90, 107–109
- labels, 109
 - pilens udformning, 109
 - placering, 108
- \arabic, 366
- \arabic* (enumitem), 242
- \arccos (mat. operator), 60
- \arcminute (siunitx enhed), 266
- \arcsecond (siunitx enhed), 266
- \arcsin (mat. operator), 60
- \arctan (mat. operator), 60
- \arg (mat. operator), 60
- argument, 9
- array (environment), 72–74, 186
- array (pakke), 169–172, 175, 176, 185, 427
- \arraybackslash, 171, 172, 182
- \ArrowBetweenLines, 352
- \ArrowBetweenLines (mathtools), 105
- \ArrowBetweenLines*, 105
- \ArrowBetweenLines* (mathtools), 105
- \ArrowBetweenLines**, 105
- article (BIB_TE_X type), 205, 208
- article (klasse), 6, 12, 15, 23, 42, 142, 223, 230, 240, 305, 314, 323, 326, 334, 335, 341–343, 347
- \ast (*), 53
- \asymp (\asymp), 53
- \AtEndDocument, 327
- \atomicmassunit (siunitx enhed), 266
- \atto (siunitx prefix), 267
- authblk (pakke), 335
- authoblk (pakke), 369
- \author, 41, 42, 274, 335
- author (BIB_TE_X felt), 205, 206, 208
- authoryear (natbib option), 204
- B**
- \b (accent understreg), 30
- \babel, 422
- babel (pakke), 6, 11, 12, 19, 24–28, 38, 110, 119, 216, 227–231, 315, 321, 357, 403, 412
- babelbib (pakke), 216
- bachelorprojekt
- fortløbende figurnummerering, 326
- \backmatter (memoir), 322
- \backprime (\backprime), 53
- \backsim (\backsim), 53
- \backsimeq (\backsimeq), 53
- \backslash, 9, 53
- badness, 404
 - baggrundsbilleder, 156
- \bar, 63
- \bar (siunitx enhed), 266
- \barn (siunitx enhed), 266
- \barwedge (\barwedge), 53
- \baselineskip, 320
- \bdot, 280
- \beamer, 278
- beamer (klasse), xv, 272–278, 429
- \author, 274
 - \date, 274
 - frame, 273–276
 - \frametitle, 273, 276
 - Frankfurt, 273, 274, 276
 - handout, 277
 - infolines, 274
 - \institute, 274
 - Luebeck, 275, 276
 - miniframes, 273, 276
 - \mode, 277
 - notheorem, 276
 - notheorems, 276
 - \pause, 275
 - \pdfpageuselayout, 278
 - \subtitle, 274
 - \title, 274
 - \titlepage, 274
 - trans, 278
 - \uncover, 275
- \becquerel (siunitx enhed), 266
- \begingroup, 98, 132, 356
- \bel (siunitx enhed), 266
- \belowcaptionskip, 152
- Bera Sans, 236
 - BeraMono, 236
 - beramono (pakke), 236, 412
 - berasans (pakke), 236, 412
 - beskrivende liste, 38
- \beta (β), 52, 282
- beviser
- med amsthm, 119
 - med ntheorem, 125
 - slutmarkør forsvinder, 124
- \bf (brug ikke), 32
- \bfseries, 32, 33, 40, 121, 127, 246
- bfseries (environment), 32
- biber (program), 222
- \bibitem, 198, 199
- biblatex (pakke), 216, 222, 360
- bibliografi, 197–222

- bibliografi fortsat...*
- konstruktion af, 197
- \bibliography, 200, 201, 214, 216, 254, 294, 323
- \bibliographysec (multibib), 218
- \bibliographystyle, 200–203, 208–214, 218, 219, 294, 323
- \bibliographystylesec (multibib), 218, 219
- \bibmark (memoir), 311
- \bibname (sprogspecifikt navn), 230
- \bibsection (memoir), 324
- BibTeX
- and, 206
- felter
- , 207
- address, 205, 208
- annotate, 208
- author, 205, 206, 208
- booktitle, 206, 208, 213
- chapter, 205
- edition, 205
- editor, 205, 208
- file, 208
- howpublished, 205
- issn, 205
- journal, 205, 208
- month, 205
- note, 205–207
- number, 205
- organization, 205
- pages, 205, 208
- publisher, 205, 206, 208
- school, 205
- series, 205, 208
- shorttitle, 208
- title, 205, 206, 208
- type, 205
- url, 205, 207
- volume, 205, 208
- year, 205, 206, 208
- forfattersortering, 206
- få alle punkter med i litteraturlisten, 201
- mere end en forfatter, 206
- stile
- abbrv, 209, 218
- abbrvnat, 212
- alpha, 201, 209
- apalike, 211, 213, 216
- chicago, 202, 203, 212, 213
- dk-abbrv, 210
- dk-alpha, 210
- dk-apali, 210
- dk-plain, 209
- dk-unsrt, 210
- dfltxbbibtex, 213
- jox, 214
- jurabib, 214
- kluwer, 213
- plain, 200, 208
- plainnat, 211
- unsrt, 201, 209
- unsrtnat, 212
- store bogstaver bliver til små, 206
- titel opfører sig underligt, 206
- titelfelt syntaks, 206
- typer
- article, 205, 208
- book, 205, 206, 208
- booklet, 205
- inbook, 205
- incollection, 208, 213
- inproceedings, 206
- phdthesis, 205
- proceedings, 205
- URL i en indgang, 207
- bibtex (program), 200, 204, 205, 219, 222, 256, 323, 379
- bibtex8 (program), 205, 216
- bibtopic (pakke), xiii, 217–219
- \btPrintAll, 220
- \btPrintCited, 220
- \btPrintNotCited, 220
- btSect, 220
- bibunits (pakke), 217
- \Big (hegnsforstørrer), 66, 103, 234
- \big (hegnsforstørrer), 66, 103, 410
- \bigbox (\square), 56
- \bigcap (\cap), 56
- \bigcup (\cup), 56
- bigfoot (pakke), 225, 226, 249
- \Bigg (hegnsforstørrer), 66, 103
- \bigg (hegnsforstørrer), 66, 103, 234
- \Biggl (hegnsforstørrer), 66
- \biggl (hegnsforstørrer), 66–68
- \Biggm (hegnsforstørrer), 66, 68
- \biggm (hegnsforstørrer), 66, 68
- \Biggr (hegnsforstørrer), 66
- \biggr (hegnsforstørrer), 66–68
- \biginterleave (\llcorner), 56
- \Bigl (hegnsforstørrer), 66, 104
- \bigl (hegnsforstørrer), 66, 71, 99, 332
- \Bigm (hegnsforstørrer), 66, 68
- \bigm (hegnsforstørrer), 66, 68
- \bigoplus (\oplus), 56
- \bigotimes (\otimes), 56
- \bigparallel (\parallel), 56
- \Bigr (hegnsforstørrer), 66, 104
- \bigR (hegnsforstørrer), 66, 71, 99, 332
- \bigsqcap (\sqcap), 56
- \bigsqcup (\sqcup), 56
- \bigstar (★), 53

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
Æ
Ø
Å

\bigtimes, 56
 \bigvee (\vee), 56
 \bigwedge (\wedge), 56
 bindestreg (-), 26
 \binom, 70
 binomialkoefficient
 \binom, 70
 \dbinom, 70
 \tbinom, 70
 bitmap
 skalering af, 161
 \blacklozenge (\blacklozenge), 53
 \blacksquare (\blacksquare), 53
 \blacktriangle (\blacktriangle), 53, 125
 \blacktriangledown (\blacktriangledown), 53
 blindtext (pakke), 231
 blkarray (pakke), 74
 block (environment), 74
 blockarray (environment), 74
 \bm, 57–59, 233
 bm (pakke), 6, 12, 59, 231, 334, 412
 Bmatrix (environment), 71
 bmatrix (environment), 71
 bmatrix* (environment), 71
 \bmod, 70
 \boldmath, 59
 book (BIB_{TEX} type), 205, 206, 208
 book (klasse), 11, 12, 42, 230, 301, 303, 305, 314, 341, 343, 344
 booklet (BIB_{TEX} type), 205
 bookmark (pakke), 287
 booktabs (pakke), 174–179, 185, 189, 193, 195, 423
 \addlinespace, 175
 \bottomrule, 175
 \cmidrule, 175, 176
 \cmidrulewidth, 175
 \heavyrulewidth, 175
 \lightrulewidth, 175
 \midrule, 175
 \specialrule, 175
 \toprule, 175
 booktitle (BIB_{TEX} felt), 206, 208, 213
 \bot (\perp), 53
 \bottomfraction, 144
 bottomnumber (tæller), 144
 \bottomrule (booktabs), 175, 176, 178, 179, 182, 183, 195
 \Box (\square), 53
 boxedminipage (environment), 372, 374
 boxedminipage (pakke), 372
 \bracket, 103
 break (theoremstyle, nthm), 122, 123, 132
 breakurl (pakke), 286
 breqn (pakke), 68, 269
 \breve, 63

brug ikke
 a4, 416
 a4wide, 416
 ae, 417
 aecompl, 417
 \bf, 32
 epsfig, 416
 glossary, 416
 isolatin1, 416
 \it, 32
 latexsym, 417
 mathptm, 417
 numprint, 416
 palatino, 417
 psfig, 416
 pslatex, 417
 \sc, 32
 sistyle, 416
 siunit, 416
 subfigure, 154, 416, 423
 tlenc, 416
 times, 417
 brøker
 \cfraction, 69
 \dfraction, 69
 \fraction, 69
 \tfraction, 69
 Bsmallmatrix (environment), 72
 bsmallmatrix (environment), 72
 \btPrintAll, 220
 \btPrintCited, 220
 \btPrintNotCited, 220
 btSect (environment), 220
 \bullet (\bullet), 53

C
 \c (accent cedille), 30
 C (søjle specifier), 179
 c (søjle specifier), 168, 171–174, 176, 179, 182, 186
 calc (pakke), 364, 365, 400, 412
 \calccentering (memoir), 320
 \Call, 284
 \candela (siunitx enhed), 266
 \cap (\cap), 53
 \caption, 143, 146–148, 150–154, 169, 185, 188, 189, 191, 192, 321, 399, 400
 emulere \caption udenfor floats, 153
 mindre fontstørrelse, 150
 spacing før og efter, 152
 ændring af bredde, 151
 caption (pakke), 150–154, 169, 188, 412, 414
 \captionof, 153
 \captionsetup, 150, 151, 154
 \captionof (caption), 153

 A
B
C
 D
 E
 F
 G
 H
 I
 J
 K
 L
 M
 N
 O
 P
 Q
 R
 S
 T
 U
 V
 W
 X
 Y
 Z
 Æ
 Ø
 Å

- \captionsdanish, 229
- \captionsetup, 152
- \captionsetup (caption), 150, 151, 154
 - cases (environment), 67, 75, 76
 - med formelnumre, *Se*
 - empheq|hyperpage
 - cases* (environment, mathtools), 76
- \cdot (·), 53, 62, 86
- \cdots (…), 62
- \ce, 281, 282
- \cee, 282
 - cellspace (pakke), 179, 186
- \cellspacebottomlimit (længde), 186
- \cellspacetoplimit (længde), 186
- \celsius (siunitx enhed), 266
- Center (environment), 38
- center (environment), 33, 37, 191, 321
- \Centering, 38, 171
- \centering, 37, 143, 146–148, 150–154, 169, 171, 172, 174, 188, 191, 192
- \centi (siunitx prefix), 267
- \centimetre (siunitx enhed), 267
- centrere indhold mht. papiret, 320
- centeret :=, 106
- \cfrac, 69
- \cftappendixname, 224, 329
- \cftchaptername, 224
- \cftfigurename, 328
- \cfttablename, 328
- \cftXnumwidth (memoir), 315
 - change (theoremstyle, nthm), 122
 - changebreak (theoremstyle, nthm), 122
- \@chapapp, 312
- \chapnamefont (memoir), 314
- \chapnumfont (memoir), 314
- \chapter, 11, 23, 36, 146, 223–225, 230, 291, 292, 305, 306, 311, 326, 332, 340, 341, 343–348
 - chapter (BibTeX felt), 205
 - chapter (pagestyle, memoir), 308–310, 313
 - chapter (tæller), 100, 116, 365
 - chapterbib (pakke), 217, 323
- \chaptermark, 311, 343
- \chaptername, 224
- \chaptername (sprogspecifikt navn), 230
- \chapterstyle (memoir)
 - section, 326
- \chaptitelfont (memoir), 314
- \check, 63
- \checkandfixthelayout, 308
- \checkandfixthelayout (memoir), 308, 324
- \checkandtixthelayout, 308
- \chi (χ), 52
- chicago (BibTeX stil), 202, 203, 212, 213
- chngcitr (pakke), 101, 366
- chngpage (pakke), 191
- \circ (o), 53
- \circeq (≐), 53
- \circlearrowleft (↶), 55
- \circlearrowright (↷), 55
- \cite, 198–204, 214, 218, 260
 - cite (pakke), 199, 204, 421
- \cite* (natbib), 202
- \citealias (natbib), 203
- \citealp (natbib), 202
- \citealp* (natbib), 203
- \citealt (natbib), 202, 203
- \citeauthor (natbib), 202
- \citeauthor* (natbib), 202
- \citep (natbib), 202, 204
- \citep* (natbib), 202
- \citepalias (natbib), 203
- citering
 - nummerisk
 - automatisk sortering, 199
- \citesec (multibib), 218
- \citet (natbib), 202–204
- \citet* (natbib), 202
- \citetalias (natbib), 203
- \citetext (natbib), 203
- \citeyear (natbib), 203
- \citeyearpar (natbib), 203
- \cleardoublepage, 22, 23, 145, 146, 294, 321
 - cleared (pagestyle, memoir), 308
- \clearpage, 22, 145, 147, 402
- \cleartorecto (memoir), 325
- cleveref (pakke), 298, 299
- \cline, 174
 - clip (graphicx), 138, 139
- \clubpenalty, 334
- \clubsuit (♣), 53
- \cm (siunitx enhed), 267
- \cmath, 282
- \cmidrule (booktabs), 175, 176
- \cmidrulewidth (booktabs), 175
- colon (natbib option), 203
- \colops (gauss), 74
- \color, 246
 - color (pakke), 245, 412, 422, 425
- \colorbox, 246
- \colorlet, 130, 245
 - colortbl (pakke), 193
- column specifier, 178
- \columnsep (længde), 365
- comma (natbib option), 203
- \Comment, 284
 - comment (environment), 295, 412
 - comment (pakke), 295, 412
 - compactenum (environment), 241
 - compactitem (environment), 241
- \CompileMatrices (Xy-pic), 110
- \complement (C), 53

- Concrete, 235
- `\cong` (\cong), 53, 57
- `\contentsname`, 322
- `\contentsname` (sprogspecifikt navn), 230
- ConTeXt, 5
- `\ContinuedFloat`, 154
- `\ContinuedFloat` (subfig), 155
- contmath (pakke), 235
- convert (program), 160, 161, 398
- cool (pakke), 352
- `\coprod` (\coprod), 56
- `\copypagestyle` (memoir), 309, 310, 313
- `\copyright` (\copyright , textcomp), 30, 320
- `\cos`, 48, 60
- `\cosh` (mat. operator), 60
- `\cot` (mat. operator), 60
- `\coth` (mat. operator), 60
- `\coulomb` (siunitx enhed), 266
- `\counterwithin` (memoir), 366
- `\counterwithout` (memoir), 101, 366
- `\createmark` (memoir), 311, 312
- `\createplainmark` (memoir), 311, 312
- `\csc` (mat. operator), 60
- `\csdef`, 361
- csquotes (pakke), 25, 412
- CSV data, 194
- `\ctan`, 414
- `\cubed` (siunitx prefix), 267
- `\cubic` (siunitx prefix), 267
- `\cup` (\cup), 53
- `\curvearrowleft` (\curvearrowleft), 55
- `\curvearrowright` (\curvearrowright), 55
- `\cymatrix`, 111
- D**
- `\d` (accent prik under), 30
- D (søjle specifier), 178, 179
- d (søjle specifier), 178
- `\dagger` (\dagger), 53
- danish (klasse option), 11
- `\danishhyphenmins`, 229
- dansk opsætning, 19
- danske bogstaver, 29
- danske vokaler
- i matematik, 59
- `\dashleftarrow` (\dashleftarrow), 55
- `\dashrightarrow` (\dashrightarrow), 55
- datatool (pakke), 194, 195, 248
- `\date`, 41, 42, 274
- `\day` (siunitx enhed), 266
- `\dbinom`, 70
- dcases (environment), 75
- dcases (environment, mathtools), 75
- dcases* (environment, mathtools), 76
- dcolumn (pakke), 71, 177–181
- `\ddagger` (\ddagger), 53
- `\ddddot`, 63
- `\dddot`, 63
- `\ddot`, 63
- `\ddots` (\ddots), 62
- deactivate (environment), 111
- `\deca` (siunitx prefix), 267
- `\deci` (siunitx prefix), 267
- `\decibel` (siunitx enhed), 266
- `\DeclareDocumentCommand`, 130
- `\DeclareGraphicsExtensions`, 136
- `\DeclareGraphicsRule`, 140
- `\DeclareMathOperator`, 60, 61, 280, 352
- `\DeclareMathOperator*`, 54, 60
- `\DeclarePairedDelimiter`, 103, 333, 352
- `\DeClarePairedDelimiter` (mathtools), 103
- `\DeclarePairedDelimiter` (mathtools), 103
- `\DeclarePairedDelimiterX` (mathtools), 103
- `\DeclarePairedDelimiterX` (mathtools), 103
- `\def`, 335, 351, 411
- `\defcitealias` (matbib), 203
- `\definecolor`, 245
- `\DefineSequence`, 361
- `\defineshorthand`, 27, 228
- defn (environment), 118, 123
- `\deg` (mat. operator), 60
- `\degree` (siunitx enhed), 266
- `\degreeCelsius` (siunitx enhed), 266
- deklarationsmakroer, 30, 31
- delimiter, *Se* hegn
- `\Delta` (Δ), 52, 282
- `\delta` (δ), 52
- `\depth` (længde), 373
- `\depthof`, 364
- description (environment), 41, 239, 242, 244, 356
- `\descriptionlabel`, 356
- `\dfrac`, 69, 186
- `\DH` ($\text{\textcircled{D}}$), 30
- `\dh` ($\text{\textcircled{d}}$), 30
- `\Diamond` (\diamond), 53
- `\diamond` (\diamond), 53
- `\diamondsuit` (\diamond), 53
- `\diff`, 71
- `\dim` (mat. operator), 60
- `\discretionary`, 86
- `\displaybreak`, 98
- `\displaystyle`, 45, 75, 112, 132
- `\DistEq`, 352
- `\div` (\div), 53
- `\DJ` ($\text{\textcircled{D}}$), 30
- `\dj` ($\text{\textcircled{d}}$), 30
- dk-abbrev ($\text{\textcircled{B}}\text{\textcircled{T}}\text{\textcircled{E}}\text{\textcircled{X}}$ stil), 210
- dk-alpha ($\text{\textcircled{B}}\text{\textcircled{T}}\text{\textcircled{E}}\text{\textcircled{X}}$ stil), 210
- dk-apali ($\text{\textcircled{B}}\text{\textcircled{T}}\text{\textcircled{E}}\text{\textcircled{X}}$ stil), 210
- dk-apali (pakke), 213
- dk-bib (pakke), 210, 211, 216

- dk-plain (BIBTEX stil), 209
 dk-unsrc (BIBTEX stil), 210
 dlfltxbbibtex (BIBTEX stil), 213
 dlfltxbcodetips (pakke), xviii, 413
 dlfltxbmarkup (pakke), 254, 412
 dlfltxbmarkupbookkeys (pakke), 254
 dlfltxbmisc (pakke), 412
 dlfltxbtocconfig (pakke), 413
 document (environment), 6, 10, 15, 142
 \documentclass, 6, 10, 11, 15, 18, 142, 294
 dokumentklasse, 10
 amsart, 42, 45, 120, 335
 amsbook, 45, 120
 article, 6, 12, 15, 23, 42, 142, 223, 230,
 240, 305, 314, 323, 326, 334, 335,
 341–343, 347
 beamer, xv, 272–278, 429
 book, 11, 12, 42, 230, 301, 303, 305,
 314, 341, 343, 344
 letter, 12, 22, 247, 305
 memoir, xv, xvi, 11, 12, 22, 33, 35, 39,
 42, 87, 101, 140, 143–146, 150,
 154, 156, 169, 170, 172, 174, 177,
 182, 185, 191, 193, 223–226, 230,
 238, 240, 242, 246, 250, 253, 255,
 258, 263, 270, 272, 278, 286, 299,
 301, 303, 305–316, 320–324,
 326–328, 330, 334, 337, 339–342,
 344–347, 353–355, 364, 366, 368,
 379, 385, 401, 404, 419, 421,
 423–426, 429–431
 options
 10pt, 11
 11pt, 11
 12pt, 11
 a4paper, 6, 11, 15, 334, 416
 danish, 11
 draft, 333, 405
 fleqn, 11, 331
 leqno, 11
 letter, 11
 onecolumn, 11
 oneside, 11, 326
 openany, 11
 openright, 11
 reqno, 11
 twocolumn, 11
 twoside, 11, 324, 326, 339, 340
 powerdot, 272
 prosper, 272
 report, 11, 12, 42, 230, 305, 341
 revtex4, 12
 scrldr2, 247
 scrldr2, 12, 22
 dokumentklasser
 KOMA-script
 scrartcl, 12, 305
 scrbook, 12, 305
 scrletter, 305
 scrletter2, 12
 scrldr2, 305
 scrreprt, 12, 305
 \dot, 63
 \dotseq (≐), 53
 \dotfill, 183
 \dots, 29, 61, 62, 85, 90, 118, 132
 \DoubleSpacing (memoir), 327
 \Downarrow (⇓), 55, 66
 \downarrow (↓), 54, 55, 66
 \dp, 375
 draft (klasse option), 333, 405
 \dtldisplayafterend, 195
 \dtldisplayafterhead, 195
 \DTLdisplaydb, 195
 \dtldisplaystarttab, 195
 \DTLforeach, 195
 \DTLiffirstrow, 195
 \DTLloaddb, 195
 Dviout, 397
 dviptfm (program), 430
 dviptfm (program), 136, 137, 306
 dviptng (program), 235
 dvips (program), 6, 8, 135–137, 139, 155,
 159, 161–164, 286
 dækblad, 302

 E
 ebb (program), 137
 \edef, 411
 edition (BIBTEX felt), 205
 editor (BIBTEX felt), 205, 208
 editorer
 Emacs, 390
 Lyx, 392, 393
 TeXMaker, 381, 388, 389
 TeXmaker, 392
 texmaker, 380
 TeXnicCenter, 14, 387–389
 texniccenter, 384, 385
 TeXShop, 14, 389, 390
 texshop, 378
 TeXstudio, 389
 TeXWorks, 14
 TeXworks, 386–389
 texworks, 378, 380
 \electronvolt (siunitx enhed), 266
 \ell (*ℓ*), 53
 ellipsis (pakke), 29
 \Else, 285
 \ElseIf, 285
 \em, 32, 46
 Emacs (editor), 390

em-dash (—), 26	\GHz, 267	
\emminershape, 33	\Hz, 267	#
\emph, 32–34, 198	\kA, 267	A
empheq (environment), 106, 352	\keV, 267	B
empheq (pakke), 76, 87, 106, 127, 425	\kg, 267	C
empty (pagestyle), 42, 308, 341	\kHz, 267	D
empty (pagestyle, memoir), 308	\kJ, 267	E
empty (theoremstyle, nthm), 123, 126, 127	\km, 267	F
emptybreak (theoremstyle, nthm), 123	\kV, 267	G
\emptyset (∅), 53	\mA, 267	H
encoding, 13	\MeV, 267	I
ansinew, 14	\meV, 267	J
applemac, 14, 390	\mg, 267	K
iso-8859-1, 14	\MHz, 267	L
latin1, 6, 12, 14, 19	\mHz, 267	M
latin9, 14	\mL, 267	N
utf8, 13–15, 390	\mm, 267	O
\end{document}, 16	\ms, 267	P
en-dash (–), 26	\mV, 267	Q
\endfirsthead (longtable), 185	\nm, 267	R
\endfoot (longtable), 185	\s, 267	S
\EndFor, 284, 285	\um, 267	T
\EndFunction, 285	\us, 267	U
\endgroup, 98, 132, 356	\gigahertz, 267	V
\endhead (longtable), 185	\gray, 266	W
\EndIf, 285	\hectare, 266	X
\endinput, 16, 291, 293, 294	\henry, 266	Y
\endlastfoot (longtable), 185	\hertz, 266, 267	Z
\EndLoop, 285	\hour, 266	Æ
\endnote, 226	\joule, 266	Ø
endnotes, 35, 225	\katal, 266	Å
endnotes (pakke), 226	\kelvin, 266	
\EndProcedure, 285	\kg, 266	
\EndWhile, 284, 285	\kiloampere, 267	
enhed (siunitx)	\kiloelectronvolt, 267	
\ampere, 266	\kilogram, 266, 267	
\angstrom, 266	\kilohertz, 267	
\arcminute, 266	\kilojoule, 267	
\arcsecond, 266	\kilometre, 267	
\atomicmassunit, 266	\kilovolt, 267	
\bar, 266	\liter, 266	
\barn, 266	\litre, 266	
\becquerel, 266	\lumen, 266	
\bel, 266	\lux, 266	
\candela, 266	\megaelectronvolt, 267	
\celsius, 266	\megahertz, 267	
\centimetre, 267	\meter, 266	
\coulomb, 266	\metre, 266	
\day, 266	\MHz, 266	
\decibel, 266	\micrometre, 267	
\degree, 266	\microsecond, 267	
\degreeCelsius, 266	\milliampere, 267	
\electronvolt, 266	\millielectronvolt, 267	
\farad, 266	\milligram, 267	
forkortet	\millihertz, 267	
\cm, 267	\millilitre, 267	

<i>enhed (siunitx) fortsat...</i>	\ensuremath, 86, 121, 125	
\millimetre, 267	enumerate (environment), 39–41, 119, 132, 239, 240, 242, 244, 402	#
\millisecond, 267	enumerate (pakke), 40, 239, 240, 242	A
\millivolt, 267	enumi (tæller), 243	B
\minute, 266	enumii (tæller), 243	C
\mmHg, 266	enumitem (pakke), 39, 41, 190, 239, 240, 242, 245, 413	D
\mole, 266	align, 243	E
\nanometre, 267	\Alph*, 242	F
\neper, 266	\alph*, 242	G
\newton, 266	\arabic*, 242	H
\ohm, 266	noitemsep, 243	I
\pascal, 266	ref, 242	J
\percent, 266	resume*, 243	K
\radian, 266	\Roman*, 242	L
\second, 266, 267	\roman*, 242	M
\siemens, 266	\setdescription, 244	N
\sievert, 266	\setenumerate, 244	O
\steradian, 266	\setitemize, 244	P
\tesla, 266	\setlist, 244	Q
\tonne, 266	start, 243	R
\volt, 266	style, 244	S
\watt, 266	environment, 9	T
\weber, 266	definere egne, 357	U
enhedspotens (siunitx)	environments	V
\cubed, 267	abstract, 303, 321	W
\cubic, 267	adjustwidth, 320, 321	X
\square, 267	adjustwidth*, 191, 320	Y
\squared, 267	algorithm, 285	Z
enhedsprefix (siunitx)	algorithmic, 283	Æ
\atto, 267	align, 47–51, 76, 90–92, 95–98, 101, 104, 127, 133, 275, 332, 404, 409	Ø
\centi, 267	align*, 49, 50, 54, 64, 87–91, 93, 94, 97, 99, 104–106, 275, 282	Å
\deca, 267	alignat, 76, 91–93, 96, 105, 132, 133	
\deci, 267	alignat*, 92	
\exa, 267	aligned, 96, 97, 99, 101	
\femto, 267	alignedat, 89, 96, 98, 99	
\giga, 267	appendices, 224	
\hecto, 267	array, 72–74, 186	
\kilo, 267	bfseries, 32	
\mega, 267	block, 74	
\micro, 267	blockarray, 74	
\milli, 267	Bmatrix, 71	
\nano, 267	bmatrix, 71	
\peta, 267	bmatrix*, 71	
\pico, 267	boxedminipage, 372, 374	
\tera, 267	Bsmallmatrix, 72	
\yocto, 267	bsmallmatrix, 72	
\yotta, 267	btSect (bibtopic), 220	
\zepto, 267	cases, 67, 75, 76	
\zetta, 267	cases* (mathtools), 76	
enker, 333	Center, 38	
\enlargethispage, 334	center, 33, 37, 191, 321	
\enquote, 25	comment, 295, 412	
ensbetyderpile mellem linier, 105		
\enskip, 46		
\enspace, 183		
\Ensure, 284		

environments fortsat...

compactenum, 241
compactitem, 241
dcases, 75
dcases (mathtools), 75
dcases* (mathtools), 76
deactivate, 111
defn, 118, 123
description, 41, 239, 242, 244, 356
document, 6, 10, 15, 142
empheq, 106, 352
enumerate, 39–41, 119, 132, 239, 240, 242, 244, 402
eqnarray, 50, 51, 99, 331, 334, 400
equation, 6, 47, 50, 51, 95, 97, 99, 101, 111, 127, 133, 332
equation*, 45, 47, 48, 50, 55, 56, 59, 96, 103, 125
falign, 93
falign*, 93
figure, 10, 37, 142, 143, 146–148, 150–154, 168, 285, 400
filecontents*, 195
flalign, 93
FlushLeft, 38
flushleft, 37, 229, 250, 320
FlushRight, 38
flushright, 37, 229
frame (beamer), 273–276
gather, 94–98, 100, 101
gather*, 74, 89, 282
gathered, 97, 99
gmatrix, 74
inparaenum, 241
itemize, 38, 39, 41, 239, 240, 242, 244, 402
landscape, 192, 193
lemma, 117, 121, 123, 125, 275
lgathered, 97, 104
longtable, 172, 183–185, 189, 192, 258, 426
lrbox, 374, 375
matrix, 71, 73
measuredfigure, 152
minipage, 148, 149, 153, 154, 246, 364, 370–372, 400, 402, 421
multicols, 413
multiline, 88, 94, 95, 97
multiline*, 88, 95
multlined, 97
opgave, 127
otherlanguage, 227, 229, 321, 322
otherlanguage*, 228
picture, 156, 270, 365
pmatrix, 71, 74, 400
problem, 133

proof, 115, 119, 124–126, 275, 276
proof (amsthm), 119
proofof, 126
psmallmatrix, 72
quotation, 38, 357
quote, 38, 357
remark, 118
shaded, 193, 225, 246
sideways, 191, 192
sidewaysfigure, 191
sidewaystable, 191
smallmatrix, 70–72
split, 95, 97, 99, 101, 400
subappendices, 224
subequations, 76, 101, 133
table, 142, 152, 168, 183, 186, 188, 192, 285, 400
TableNotes, 189
tablenotes, 188, 189
tabular, 74, 168, 169, 171–176, 178, 179, 182, 183, 186, 188, 191, 371
tabularx, 182, 183, 186, 187
thebibliography, 10, 197–201, 311
theorem, 276
thm, 116–118, 121, 123, 125, 132
thmbreak, 132
ThreePartTable, 189
threeparttable, 188, 189
titlingpage, 308, 320
varwidth, 171, 372
verbatim, 249–252, 273
verbatim*, 249
Vmatrix, 71
vmatrix, 71
Vsmallmatrix, 72
vsmallmatrix, 72
xy, 111
zorn, 118

eps
 boundingbox
 problemer med, 158
 dvipdfmx, 137
 eps (grafik format), 137, 161
 eps2eps (program), 157
 epsfig (pakke), 136
 epsfig (pakke, brug ikke), 416
\epsilon, 51, 52
 epstool (program), 158, 159
 epstopdf (program), 158–160
 eqnarray (environment), 50, 51, 99, 331, 334, 400
 \eqref, 50, 100, 101, 104, 133
 equation (environment), 6, 47, 50, 51, 95, 97, 99, 101, 111, 127, 133, 332
 equation (tæller), 100
 equation* (environment)

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
Æ
Ø
Å

- equation** (environment) fortsat...
- \[...\], 48
 - equation** (environment), 45, 47, 48, 50, 55, 56, 59, 96, 103, 125
 - \equiv, 70
 - esint (pakke), 56
 - eso-pic (pakke)
 - baggrundsbillede, 156
 - eso-pic (pakke), 141, 156
 - \eta, 52
 - etex (pakke), 413
 - etoolbox (pakke), xi, 359, 360, 367, 369
 - \ifblank, 369
 - \ifboolexpr, 368
 - \ifdimequal, 367
 - \ifdimgreater, 367
 - \ifdimless, 367
 - \ifnumequal, 367
 - \ifnumgreater, 367
 - \ifnumless, 367
 - \ifnumodd, 367
 - \iftoggle, 368
 - \newtoggle, 368
 - \togglefalse, 368
 - \toggletrue, 368
 - \whileboolexpr, 368
 - eulervm (pakke), 233, 417
 - Evince, 395, 395, 396
 - evince (program), 6
 - \exa (siunitx prefix), 267
 - Excel (program), 167
 - Excel til L^AT_EX, 194
 - excel2latex (program), 194
 - \excludacomment, 295
 - \exists (∃), 53
 - \exp (mat. operator), 60
 - \externaldocument, 299
 - \extrarowheight (længde), 185
- F**
- falign (environment), 93
 - falign* (environment), 93
 - fancy (pagestyle), 341, 342
 - \fancybreak, 22
 - \fancybreak (memoir), 22, 330
 - \fancyfoot (fancyhdr), 341–344
 - fancyhdr (pakke), 301, 308, 341–345, 430
 - \fancyfoot, 341–344
 - \fancyhead, 341–344
 - \fancyhf, 341–344
 - fjerne linier, 342
 - \footrulewidth, 341, 342
 - \headrulewidth, 341–344
 - ikke anvendes med memoir, 341
 - \nouppercase, 344
 - \fancyhead (fancyhdr), 341–344
 - \fancyhf (fancyhdr), 341–344
 - fancyref (pakke), 298, 299, 428
 - fancyvrb (pakke), 249, 250, 403
 - \farad (siunitx enhed), 266
 - \fbox, 106, 139, 150–154, 158, 246, 372
 - \fboxrule (længde), 246, 372
 - \fboxsep (længde), 246, 372
 - \fcolorbox, 246
 - fed, 32
 - fejlmeddelelse
 - få skrevet filnavn foran, 384
 - fejlmeddelelser, Se afsnit D.1
 - \femto (siunitx prefix), 267
 - \fi, 140, 263
 - figur*
 - floats i to søjler, 143
 - figure (environment), 10, 37, 142, 143, 146–148, 150–154, 168, 285, 400
 - \figurename (sprogspecifikt navn), 230
 - file (BibT_EX felt), 208
 - filecontents* (environment), 195
 - \firmlist, 242, 354
 - \firmlist (memoir), 239, 354
 - \firmlists (memoir), 239
 - fix-cm (pakke), 237, 413
 - fixltx2e (pakke), 33, 413
 - \fixme, 400
 - fixme (pakke), xviii, 295–297, 413
 - \fxerror, 296
 - \fxfatal, 296
 - \fxnote, 296
 - \fxsetup, 296
 - \fxwarning, 296
 - \listoffixmes, 296, 297
 - fjerne sidetal, 341
 - flafter (pakke), 146
 - falign (environment), 93
 - falign*, 93
 - fleqn (klasse option), 11, 331
 - flerliniet matematik
 - kasse om, 106
 - flette PDF-filer, 142
 - float (pakke), 146
 - float(s) lost
 - \fixme, 400
 - marginpar, 400
 - \FloatBarrier, 145–147, 153, 355, 402, 414
 - \FloatBlock (memoir), 145
 - floatflt (pakke), 155
 - \floatpagefraction, 144
 - floats, 142
 - \floatsep (længde), 144
 - FlushLeft (environment), 38
 - flushleft (environment), 37, 229, 250, 320
 - FlushRight (environment), 38
 - flushright (environment), 37, 229

\fnsymbol, 366	\mdseries, 32	
\fodnote, 188	\normalfont, 32	#
fodnoter, 35, 225	\rmfamily, 32	A
font	\scshape, 32	B
antikva, 32	\sffamily, 32	C
\bfseries, 32	\slshape, 32	D
\em, 32	\textbf, 32	E
\emph, 32	\textit, 32	F
fed, 32	\textmd, 32	G
fontskift, 31	\textnormal, 32	H
\footnotesize, 31	\textrm, 32	I
fremhævet, 32	\textsc, 32	J
\Huge, 31	\textsf, 32	K
\huge, 31	\textsl, 32	L
italic, <i>Se kursiv</i>	\texttt, 32	M
\itshape, 32	\textup, 32	N
kursiv, 32	\ttfamily, 32	O
\LARGE, 31	\upshape, 32	P
\Large, 31	fontstørrelser, 30	Q
\large, 31	\footnotesize, 31	R
\mdseries, 32	\Huge, 31	S
monospace, 32	\huge, 31	T
normal tekst, 32	\LARGE, 31	U
\normalfont, 32	\Large, 31	V
\normalsize, 31	\large, 31	W
\rmfamily, 32	\normalsize, 31	X
sans serif, 32	\scriptsize, 31	Y
\scriptsize, 31	\small, 31	Z
\scshape, 32	\tiny, 31	Æ
\sffamily, 32	\footcite, 214	Ø
slantet, 32	footmisc (pakke), 225, 226	Å
\slshape, 32	\footnote, 35, 225, 226, 400	
\small, 31	footnote (tæller), 225	
small caps, 32	\footnotemark, 225	
størrelse, 30	\footnotesize, 31	
\textbf, 32	\footnotetext, 225	
\textmd, 32	\footruleskip (memoir), 310	
\textrm, 32	\footrulewidth (fancyhdr), 341, 342	
\textsf, 32	\For, 284	
\textit, 32	\ForAll, 285	
\textnormal, 32	\forall (V), 47, 53, 84	
\textsc, 32	\foreignlanguage, 228, 229	
\textsl, 32	formelnummer	
\texttt, 32	genbrug af, 100	
\textup, 32	konfigurering af, 100	
\tiny, 31	\numberwithin, 100	
\ttfamily, 32	vis kun refererede, 104	
\upshape, 32	vis kun refererede formler, 104	
fontenc (pakke), 6, 12, 13, 19, 24, 26, 29, 252, 413, 416	formelombrydning, 83–89	
\fontsize, 237	alignment	
fontskift, 31	almindelig, 87	
\bfseries, 32	brudt hegn, 88	
\em, 32	brudte linier, 88	
\emph, 32	multiline, 88	
\itshape, 32	udregning, lange linier, 87	
	automatisk, 85	

formelombrydning fortsat...

automatisk indsættelse af gangetegn, 86
 fremhævede formler, 86–89
 sideskift, 98, 99
 her må ikke deles, *Se* \nobreak
 i teksten, 83
 indsat spacing, 84
 indsættelse af ombrydningspunkt, 85
 integraltegn, 84
 store operatorer, 84
 ved /, 84
 ved omkransning af hegn, 84
 ved operator, 84
 ved par af hegn, 84
 ved relation, 84
 ved usynligt gangetegn, 86
 formelopsætning
 samling af formler, 89
 forside
 IMF krav til, 303
 louises forside, 317
 louises forside – kildekode, 316
 strits forside, 319
 strits forside – kildekode, 318
 forøge afstand
 mellem linier/rækker i matematik, 91
 fourier (pakke), 36, 233, 234, 318, 330
 fouriernc (pakke), 234
 Foxit, 395
 \frac, 45, 69–71, 103
 frame (environment), 273–276
 \framebox, 372
 framed (pakke), 193, 225, 246
 \frametitle (beamer), 273, 276
 Frankfurt (beamer), 273, 274, 276
 fremhævede formler
 centreret, 94
 flere centrerede linier med ét nummer, 97
 fremhævet, 32, 45
 fremvisere
 Xdvi, 397
 \frenchspacing, 28
 \frontmatter, 294, 314, 320, 322
 tekst uden nummer, men med i ToC, 321
 \Function, 284, 285
 funktioner
 navngivne, 59
 \fxerror (fixme), 296
 \fixfatal (fixme), 296
 \fxnote (fixme), 296
 \fxsetup (fixme), 296
 \fxwarning (fixme), 296

G

gaffelfunktion, 75
 \Gamma (Γ), 52
 \gamma (γ), 52
 Garamond, 234
 gather (environment), 94–98, 100, 101
 gather* (environment), 74, 89, 282
 gathered (environment), 97, 99
 gauss (pakke), 74
 \add, 74
 \colops, 74
 gmatrix, 74
 \mult, 74
 \rowops, 74
 \swap, 74
 \geometry, 337
 geometry (pakke), 306, 307, 324, 337–340, 364
 \geq (\geq), 47, 53, 84
 \geqq (\geqq), 53
 \geqslant (\geqslant), 53
 getnonfreefonts (program), 252, 383
 getnonfreefonts-sys (program), 383
 \gg (\gg), 53
 \ggg (\ggg), 53
 ghostscript (program), 160
 \GHz (siunitx enhed), 267
 \giga (siunitx prefix), 267
 \gigahertz (siunitx enhed), 267
 \global, 368, 375
 glossaries (pakke), 258, 259, 261, 262, 416
 \addgls, 261
 \Gls, 261
 \gls, 261
 \glsaddall, 261
 \glsdisp, 261
 \glslink, 261
 \glspl, 261
 \loadglsentries, 259
 \makeglossaries, 259
 \newacronym, 259
 \newglossaryentry, 259, 260
 glossary (pakke, brug ikke), 416
 \Gls (glossaries), 261
 \gls (glossaries), 261
 \glsaddall (glossaries), 261
 \glsdisp (glossaries), 261
 \glslink (glossaries), 261
 \glspl (glossaries), 261
 gmatrix (environment), 74
 Gnu Emacs+AUCTeX+refTeX, 390, 390, 391
 Gnumeric (program), 194
 gnuplot (program), 163
 grafik

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
Æ
Ø
Å

grafik fortsat...	\hecto (siunitx prefix), 267	
anvende grafik i både L ^A T _E X og pdfL ^A T _E X, 139	hegn, 65	#
automatisk vælge format, 139	. (punktum), 65, 67	A
dele en float, 154	(, 65	B
hentet fra andre biblioteker, 140), 65	C
inkludere dele af et andet dokument, 141	/, 65	D
METAPOST og pdfL ^A T _E X, 140	[, 65	E
grafik format	\{, 65	F
eps, 137, 161	\}, 65	G
jbig2, 136, 137], 65	H
jpg, 136	\Downarrow, 66	I
mps, 136	\downarrow, 66	J
pdf, 136, 161	\langle, 65	K
png, 136	\lceil, 65	L
PostScript, 137	\lfloor, 65	M
svg, 161	\lgroup, 65	N
\graphicspath, 140, 141	\llbracket, 65	O
graphicx (pakke), xii, 136, 154, 156, 306, 373, 413, 416, 422	\lVert, 65, 66	P
\includegraphics	\lvert, 65, 66	Q
angle, 138, 139	\rangle, 65	R
clip, 138, 139	\rceil, 65	S
height, 138, 139, 147, 148, 156, 400	\rfloor, 65	T
keepaspectratio, 138	\rgroup, 65	U
origin, 138	\rrbracket, 65	V
page, 139	\rVert, 65, 66	W
scale, 138	\rvert, 65, 66	X
totalheight, 138, 400	skalerering af, 66	Y
trim, 139, 363	skalering af	Z
viewport, 139	\Big, 66	Æ
width, 137–139, 147, 148, 156	\big, 66	Ø
\grave, 63	\Bigg, 66	Å
\gray (siunitx enhed), 266	\bigg, 66	
grffile (pakke), 140	\Biggl, 66	
gsview (program), 6	\biggl, 66–68	
Gsview32, 397	\Biggm, 66, 68	
\guillemotleft (◀), 26	\biggm, 66, 68	
\guillemotright (▶), 26	\Biggr, 66	
GV, 396, 396, 397	\biggr, 66–68	
	\Bigl, 66	
H	\bigl, 66	
\H (ungarsk umlaut), 30	\Bigm, 66, 68	
handout (beamer), 277	\bigm, 66, 68	
harvard (pakke), 199, 430	\Bigr, 66	
\hat, 63	\bigr, 66	
\hbar (\hbar), 53	\left, 66–68	
\hbox, 86	\middle, 68	
headings (pagestyle), 310, 312, 341, 342	\right, 66–68	
headings (pagestyle, memoir), 309, 312, 326	\Uparrow, 66	
\headrulewidth (fancyhdr), 341–344	\uparrow, 66	
\heartsuit (♥), 53	\Updownarrow, 66	
\heavyrulewidth (booktabs), 175	\updownarrow, 66	
\hectare (siunitx enhed), 266	\Vert, 65	
	\vert, 65	
	\vert, 65	
	height (graphicx), 138, 139, 147, 148, 156, 400	
	\height (længde), 373	

\heightof, 364
 heltals-programmering, 92
 helvet (pakke), 235
 Helvetika, 235
 \henry (siunitx enhed), 266
 \hertz (siunitx enhed), 266, 267
 \hfill, 148, 153, 370
 \hfuzz, 333, 405
 \hline, 73, 174, 186, 188, 191, 192
 \hom (mat. operator), 60
 \hookleftarrow (\leftarrow), 55
 \hookrightarrow (\rightarrow), 55
 horeunger, 333
 \hour (siunitx enhed), 266
 howpublished (BibTeX felt), 205
 \hphantom, 97, 99
 \hrule, 152
 \hslash (\hbar), 53
 \hspace, 154, 370, 373
 \hspace*, 370
 \ht, 375
 htlatex (program), 409
 \Huge, 31, 73
 \huge, 31
 hunspell (program), 386, 387
 h-vor, 228
 hyperref (pakke), 43, 254, 261, 285–287, 300, 329, 354, 379, 402, 413
 \hyphenation, 25, 227
 \Hz (siunitx enhed), 267

I

\iddots ($\cdot\cdot\cdot$), 62
 \idotsint ($\int\cdots\int$), 56
 \If, 285
 \ifblank (etoolbox), 369
 \ifboolexpr (etoolbox), 368
 \ifdim, 367
 \ifdimequal (etoolbox), 367
 \ifdimgreater (etoolbox), 367
 \ifdimless (etoolbox), 367
 \iff (\iff), 55
 \iflanguage, 229
 \ifnum, 327, 367
 \ifnumequal (etoolbox), 367
 \ifnumgreater (etoolbox), 367
 \ifnumless (etoolbox), 367
 \ifnumodd (etoolbox), 367
 \ifpdf, 140, 263
 ifpdf (pakke), 140, 263
 ifsym (pakke), 413
 \iftoggle (etoolbox), 368
 \iiint (\iiint), 56
 \iint (\iint), 56
 \iint (\iint), 56

ikke anbefalede pakker
 a4, 416
 a4wide, 416
 ae, 417
 aecompl, 417
 epsfig, 416
 glossary, 416
 isolatin1, 416
 latexsym, 417
 mathptm, 417
 numprint, 416
 palatino, 417
 psfig, 416
 pslatex, 417
 sistyle, 416
 siunit, 416
 subfigure, 154, 416, 423
 t1enc, 416
 times, 417
 \Im (\Im), 53, 60
 Imagemagick (program), 398
 \imath (i), 53
 \in (\in), 53
 inbook (BibTeX type), 205
 \include, 291–294, 323
 \includecomment, 295
 \includegraphics, 137–140, 147, 148, 156, 157, 363, 373, 400, 416
 angle, 138, 139
 clip, 138, 139
 height, 138, 139, 147, 148, 156, 400
 keepaspectratio, 138
 orden af options, 139
 origin, 138
 page, 139
 scale, 138
 totalheight, 138, 400
 trim, 139, 363
 viewport, 139
 width, 137–139, 147, 148, 156
 \includeonly, 293
 \includepdf, 141, 142, 324
 incollection (BibTeX type), 208, 213
 indeks, *Se* stikordsregister
 \indent, 93
 indentfirst (pakke), 24
 \index, 254, 255, 298
 \indexname (sprogspecifikt navn), 230
 \indexspace, 255
 indextitlepagestyle (pagestyle, memoir), 309, 313
 indholdsfortegnelse
 tvunget sideskift, 347
 \inf (mat. operator), 60
 infolines (beamer), 274
 \infty (∞), 53

 A
 B
 C
 D
 E
 F
 G
H
I
 J
 K
 L
 M
 N
 O
 P
 Q
 R
 S
 T
 U
 V
 W
 X
 Y
 Z
 Æ
 Ø
 Å

- inline, 45
 \inner, 103
 inparaenum (environment), 241
 inproceedings (BIBTEX type), 206
 \input, 193, 291–294
 input encoding, *Se* encoding
 inputenc (pakke), 6, 12, 13, 15, 19, 29, 231, 387, 390, 399, 413, 416
 \insertchapterspace, 328
 \insertTableNotes, 189
 \InsertTheoremBreak, 132
 \institute (beamer), 274
 \int (\int), 55, 56
 \intertext, 93, 94, 98, 333
 \intertextfloatsep (længde), 144
 \iota (ι), 52
 iso-8859-1 (encoding), 14
 isolatin1 (pakke, brug ikke), 416
 issn (BIBTEX felt), 205
 \it (brug ikke), 32
 italic, *Se* kursiv
 \item, 38–41, 132, 188, 189, 284
 itemize (environment), 38, 39, 41, 239, 240, 242, 244, 402
 \itemsep (længde), 132
 \itshape, 32
- J**
- jabref (program), 217, 220, 221
 jbig2
 pdflatex, 136
 jbig2 (grafik format), 136, 137
 jpeg2eps (program), 398
 \jmath (j), 53
 \joule (siunitx enhed), 266
 journal (BIBTEX felt), 205, 208
 jox (BIBTEX stil), 214
 jpeg2ps (program), 160, 398
 jpg
 dvi2pdf, 137
 pdflatex, 136
 jpg (grafik format), 136
 jurabib (BIBTEX stil), 214
 jurabib (pakke), 197, 208, 214–216
- K**
- \k (accent ogonek), 30
 \kA (siunitx enhed), 267
 \kappa (κ), 52
 \katal (siunitx enhed), 266
 keepaspectratio (graphicx), 138
 \kelvin (siunitx enhed), 266
 \ker (mat. operator), 60
 \keV (siunitx enhed), 267
 keyval (pakke), 353, 413
 \kg (siunitx enhed), 266, 267
 \kHz (siunitx enhed), 267
 Kile, 392
 \kilo (siunitx prefix), 267
 \kiloampere (siunitx enhed), 267
 \kiloelectronvolt (siunitx enhed), 267
 \kilogram (siunitx enhed), 266, 267
 \kilohertz (siunitx enhed), 267
 \kilojoule (siunitx enhed), 267
 \kilometre (siunitx enhed), 267
 \kilovolt (siunitx enhed), 267
 \kJ (siunitx enhed), 267
 kluger (BIBTEX stil), 213
 \km (siunitx enhed), 267
 kolofon, 302
 kommentar i alignment, 93
 konfigurering af formelnummer, 100
 Konvertering
 Excel til L^AT_EX, 194
 Kp-fonts, 234
 kpfonts (pakke), 56, 234, 413
 kursiv, 32
 \kV (siunitx enhed), 267
 kvadratrod, *Se* \sqrt
- L**
- \L (\mathbb{L}), 30
 \l (\mathfrak{l}), 30
 l (søjle specifier), 168, 172, 173, 175, 176, 188, 191, 192
 \label, 34, 35, 40, 50, 51, 99–101, 104, 133, 143, 146, 148, 150, 154, 298, 366
 \Lambda (Λ), 52
 \lambda (λ), 52
 landscape (environment), 192, 193
 \lang (hegn), 65, 103
 \LARGE, 31
 \Large, 31
 \large, 31, 121, 237
 lastsheet (tæller, memoir), 327
 \LaTeX, 4, 10, 207, 320
 udtalen af, 4
 latex (program), 8, 19, 137, 139, 140, 155, 163, 200, 219, 256, 286
 latexmk (program), 256, 257, 379, 385
 latexsym (pakke, brug ikke), 417
 Latin Modern, 232
 latin1 (encoding), 6, 12, 14, 19
 latin9 (encoding), 14
 \lceil (hegn), 65
 \ldotp (\cdot), 62
 \ldots (\dots), 62
 \leadsto (\leadsto), 55
 Led, 392
 \left, 49, 66–68, 70, 71, 73, 77, 103, 332, 410
 \Leftarrow (\Leftarrow), 55

`\leftarrow` (←), 55
`\lefttharpoonup` (↵), 55
`\lefttharpoonup` (←), 55
`\leftmark`, 309–312, 326, 343, 344
`\leftoverarrow`, 63
`\Leftrightarrow` (⇔), 55
`\leftrightarrow` (↔), 55
`\leftrightharpoonup` (⇔), 55
`\leftrightharpoons` (⇔), 55
`\leftrightharpoons` (⇔), 55
lemma (environment), 117, 121, 123, 125, 275
`\leq` (≤), 53
leqno (klasse option), 11
`\leqq` (≲), 53
`\leqslant` (≲), 53
`\lessapprox` (≲), 53
`\let`, 146, 183, 204, 355
letter (klasse option), 11
letter (klasse), 12, 22, 247, 305
letterspace, 238
letterspacing, 238
`\lfloor` (hegn), 65
`\lg` (mat. operator), 60
lgathered (environment), 97, 104
`\lgroup` (hegn), 65
ligaturer, 26
`\lightning` (⚡), 55
`\lightrulewidth` (booktabs), 175
`\lim` (mat. operator), 60
`\limext`, 60
`\liminf` (mat. operator), 60
`\limits`, 55
`\limsup` (mat. operator), 60
`\linebreak`, 86, 333
`\linespread`, 237, 238, 327
`\linewidth` (længde), 138, 182, 183, 365
linieafstand, 237
`\lipsum`, 230
lipsum (pakke), 230, 231, 413
`\lipsum*`, 231
lister, 38
description, 41
enumerate, 39
itemize, 38
`\listfigurename` (sprogspecifikt navn), 230
listings (pakke), 250–252, 273, 333, 413, 425, 426
`\listoffigures`, 304
`\listoffixmes` (fixme), 296, 297
`\listoftables`, 304
`\listtablename` (sprogspecifikt navn), 230
`\liter` (siunitx enhed), 266
`\litre` (siunitx enhed), 266
litteraturliste, *Se bibliografi*
`\ll` (≪), 53

`\llap`, 111, 112
`\llbracket` (hegn), 65
`\lll` (≪≪), 53
lmodern (pakke), 27, 232, 237
`\ln` (mat. operator), 60
`\loadglsentries` (glossaries), 259
`\log` (mat. operator), 60, 352
`\longleftarrow` (←), 55
`\longleftarrow` (←), 55
`\longleftarrow` (←), 55
`\longmapsto` (↦), 55
`\longmapsto` (↦), 55
longnamesfirst (natbib option), 204
`\longrightarrow` (→), 55
longtable (pakke)
endfirsthead, 185
endfoot, 185
endhead, 185
endlastfoot, 185
longtable (environment), 172, 183–185, 189, 192, 258, 426
longtable (pakke), 183, 185, 188, 189, 413
`\Loop`, 285
Lorem ipsum, 230
`\lozenge` (◇), 53
lrbox (environment), 374, 375
lscap (pakke), 192
`\lstinputlisting`, 251
LuaL^AT_EX, 5
Luebeck (beamer), 275, 276
`\lumen` (siunitx enhed), 266
`\lux` (siunitx enhed), 266
LuxiMono, 236
luximono (pakke), 236, 252
`\lVert` (hegn), 65, 66
`\lvert` (hegn), 65, 66, 103
Lyx, 392
Lyx (editor), 392, 393
længde
cellspacebottomlimit, 186
cellspacetoplimit, 186
columnsep, 365
depth, 373
extrarowheight, 185
fboxrule, 246, 372
fboxsep, 246, 372
floatsep, 144
height, 373
intextfloatsep, 144
itemsep, 132
linewidth, 138, 182, 183, 365
paperheight, 156, 365
paperwidth, 156, 365
parsep, 132
tabcolsep, 186
`\@tempdima`, 358, 365
`\@tempdimb`, 365

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
Æ
Ø
Å

- længde fortsat...*
- \@tempdimc, 365
 - \textfloat, 144
 - \textheight, 138, 365, 415
 - \textwidth, 138, 148, 153, 182, 310, 365, 415
 - \totalheight, 373
 - \unitlength, 365
 - \width, 373
- M**
- \mA (siunitx enhed), 267
 - mailing (pakke), 247
 - \mainmatter, 294, 314, 322, 328
 - \makeatletter, 27, 228, 358, 359, 403
 - \makeatother, 27, 228, 358, 359, 403
 - \makeabox, 372, 373
 - makebst (program), 213, 216, 423
 - \makeevenfoot (memoir), 270, 309, 310, 312
 - \makeevenhead (memoir), 309, 312, 313
 - \makefootrule (memoir), 309, 310
 - \makeglossaries (glossaries), 259
 - makeglossaries (program), 259, 262, 385
 - \makeheadrule (memoir), 309, 310
 - makeidx (pakke), 230, 253
 - \makeindex, 254
 - makeindex (program), ix, 255–258, 379, 415
 - \makeoddfoot (memoir), 270, 309, 312
 - \makeodhead (memoir), 309, 310, 312, 313
 - \makepagestyle (memoir), 309, 310, 312
 - \makepsmarks (memoir), 309, 311, 312
 - \maketitle, 41, 42, 225, 226, 308, 327, 340, 341
 - sidetal, 341
 - \MakeUppercase, 318
 - makro
 - definere egne, 351
 - gemme indhold, 355
 - omdefinere, *Se* \renewcommand
 - makroer, 9
 - \mapsfrom (\leftarrow), 55
 - \mapsto (\mapsto), 55
 - margin (theoremstyle, nthm), 122
 - marginbreak (theoremstyle, nthm), 122, 123
 - \marginpar, 400
 - \marginpars, 402
 - \markboth, 343
 - \markright, 343, 344
 - \markup, 412
 - mateamtik,
 - matematik, 45–113
 - _, 46
 - alignment
 - align, 91
 - alignat, 92
 - mode, 45
 - matematik-mode, 45
 - math unit, 46
 - \mathbb, 47, 57, 58, 75
 - \mathbbm, 58
 - \mathbf, 58, 59
 - \mathbin, 410
 - \mathcal, 58, 352
 - \mathchoice, 70
 - \mathclap, 64, 104
 - mathdesign (pakke), 234
 - mathdots (pakke), 62
 - \mathds, 57, 58
 - \mathfrak, 57, 58
 - \mathit, 57, 58, 280
 - math-mode, *Se* matematik-mode
 - \mathop, 410
 - \mathord, 410
 - mathpazo (pakke), 233, 417
 - mathptm (pakke, brug ikke), 417
 - mathptms (pakke), 233
 - mathptmx (pakke), 232, 417
 - \mathrel, 410
 - \mathring, 63
 - \mathrlap, 105
 - \mathrm, 57–59, 268
 - \mathscr, 57, 58
 - \mathsf, 58
 - \mathsl, 57
 - \mathstrut, 77
 - mathtools (pakke), 6, 12, 54–56, 64–66, 69, 71, 72, 75, 76, 87, 94, 95, 97, 100, 102–106, 333, 334, 385, 413
 - \ArrowBetweenLines, 105
 - \ArrowBetweenLines*, 105
 - cases*, 76
 - dcases, 75
 - dcases*, 76
 - \DeclarePairedDelimiter, 103
 - \DeclarePairedDelimiter, 103
 - \DeclarePairedDelimiterX, 103
 - \DeclarePairedDelimiterX, 103
 - hack ved brede grænser, 102
 - \mathtoolsset, 104
 - \MoveEqLeft, 89, 95, 97, 105, 106
 - showmanualtags, 104
 - showonlyrefs, 104
 - \splitdffrac, 69
 - \splitfrac, 69
 - \mathtoolsset (mathtools), 104
 - matricer
 - array, 72
 - Bmatrix, 71
 - bmatrix, 71
 - matrix, 71
 - pmatrix, 71

- matricer fortsat...*
 Vmatrix, 71
 vmatrix, 71
 matrix
 mere end 10 søjler, 72
 matrix (environment), 71, 73
 \max (mat. operator), 60
 MaxMatrixCols (tæller), 72
 \maxsecnumdepth (memoir), 314, 315
 \mbox, 25, 229, 372, 373
 \mc, 177, 178
 \mdfdefinestyle, 128, 129
 mdframed (pakke), 128–130, 194, 246, 413
 \mdseries, 32
 \measuredangle (\sphericalangle), 53
 measuredfigure (environment), 152
 medspace ($\backslash:$), 46
 \mega (siunitx prefix), 267
 \megaelectronvolt (siunitx enhed), 267
 \megahertz (siunitx enhed), 267
 memoir (klasse), xv, xvi, 11, 12, 22, 33, 35, 39, 42, 87, 101, 140, 143–146, 150, 154, 156, 169, 170, 172, 174, 177, 182, 185, 191, 193, 223–226, 230, 238, 240, 242, 246, 250, 253, 255, 258, 263, 270, 272, 278, 286, 299, 301, 303, 305–316, 320–324, 326–328, 330, 334, 337, 339–342, 344–347, 353–355, 364, 366, 368, 379, 385, 401, 404, 419, 421, 423–426, 429–431
 \addlinespace, 175
 \addtopmarks, 311, 312
 adjustwidth, 320, 321
 adjustwidth*, 191, 320
 \aliaspagestyle, 309, 310
 \appendixpage, 224, 322
 \appendixpagename, 224, 322
 \appendixtocname, 224, 322
 \backmatter, 322
 \bibmark, 311
 \bibsection, 324
 \bottomrule, 175
 \calccentering, 320
 \cftXnumwidth, 315
 \chapnamefont, 314
 \chapnumfont, 314
 \chapterstyle
 section, 326
 \chaptitelfont, 314
 \checkandfixthelayout, 308, 324
 \cleartorecto, 325
 \cmidrule, 175, 176
 \cmidrulewidth, 175
 \copypagestyle, 309, 310, 313
 \counterwithin, 366
 \counterwithout, 101, 366
 \createmark, 311, 312
 \createplainmark, 311, 312
 \DoubleSpacing, 327
 \fancybreak, 22, 330
 \firmist, 239, 354
 \firmists, 239
 \FloatBlock, 145
 \footruleskip, 310
 \heavyrulewidth, 175
 \lightrulewidth, 175
 \makeevenfoot, 270, 309, 310, 312
 \makeevenhead, 309, 312, 313
 \makefootrule, 309, 310
 \makeheadrule, 309, 310
 \makeoddfoot, 270, 309, 312
 \makeoddhead, 309, 310, 312, 313
 \makepagestyle, 309, 310, 312
 \makepsmarks, 309, 311, 312
 \maxsecnumdepth, 314, 315
 \midrule, 175
 \normalrulethickness, 310
 \nouppercaseheads, 312
 \OnehalfSpacing, 327
 overskrive plain pagestyle, 313
 \raggedbottomsectiontrue, 314
 \setFloatBlockFor, 146
 \setlrmargins, 308
 \setlrmarginsandblock, 308
 \setpnumwidth, 315
 \setrmarg, 315
 \setsecheadstyle, 314
 \setsecnumdepth, 314, 315
 \setsubsecheadstyle, 314
 \setsubsubsecheadstyle, 314
 \settocdepth, 314, 315, 346
 \settypeblocksize, 308
 \setulmargins, 308
 \setulmarginsandblock, 308
 \SingleSpacing, 327
 \specialrule, 175
 subappendices, 224
 \tightlist, 239, 354
 \tightlists, 239
 \titleref, 330
 titlingpage, 308, 320
 \tocmark, 311
 \toprule, 175
 METAPOST, 140
 \meter (siunitx enhed), 266
 \metre (siunitx enhed), 266
 \MeV (siunitx enhed), 267
 \meV (siunitx enhed), 267
 \MF, 413
 mflogo (pakke), 320, 413
 \mg (siunitx enhed), 267

- mhchem (pakke), 281, 282
 - \mho (Ω), 53
 - \MHz (siunitx enhed), 266, 267
 - \mHz (siunitx enhed), 267
 - \micro (siunitx prefix), 267
 - \micrometre (siunitx enhed), 267
 - \microsecond (siunitx enhed), 267
 - microtype (pakke), 413
 - \mid ($|$), 53, 66
 - \middle (hegnsforstørrer), 68
 - \midrule (booktabs), 175, 176, 178, 179, 183, 195
 - \milli (siunitx prefix), 267
 - \milliampere (siunitx enhed), 267
 - \millielectronvolt (siunitx enhed), 267
 - \milligram (siunitx enhed), 267
 - \millihertz (siunitx enhed), 267
 - \millilitre (siunitx enhed), 267
 - \millimetre (siunitx enhed), 267
 - \millisecond (siunitx enhed), 267
 - \millivolt (siunitx enhed), 267
 - \min (mat. operator), 60
 - miniframes (beamer), 273, 276
 - minimalt eksempel, 18
 - minipage (environment), 148, 149, 153, 154, 246, 364, 370–372, 400, 402, 421
 - \minute (siunitx enhed), 266
 - \missingfigure, 297
 - mkind (program), 255–257
 - \ml (siunitx enhed), 267
 - \mm (siunitx enhed), 267
 - \mmHg (siunitx enhed), 266
 - MnSymbol (pakke), 54
 - \mod, 70
 - \mode (beamer), 277
 - \models (\models), 53
 - modulus
 - \bmod, 70
 - \mod, 70
 - \pmod, 70
 - \pod, 70
 - \mole (siunitx enhed), 266
 - monospace, 32
 - month (BibTeX felt), 205
 - \MoveEqLeft, 105, 352
 - \MoveEqLeft (mathtools), 89, 95, 97, 105, 106
 - \MP, 320, 413
 - \mp (\mp), 53
 - mpost (program), 162
 - mps
 - dvipdfmx, 137
 - pdflatex, 136
 - mps (grafik format), 136
 - \ms (siunitx enhed), 267
 - \mu (μ), 52
 - \mult (gauss), 74
 - multibib (pakke), xiii, 217–219
 - \bibliographysec, 218
 - \bibliographystylesec, 218, 219
 - \citesec, 218
 - \newcite, 218
 - \newcites, 218
 - \nocitesec, 218
 - multicol (pakke), 11, 413
 - multicols (environment), 413
 - \multicolumn, 170, 172–174, 176–179, 183, 185, 187
 - anvendt som beskyttelse, 177
 - \multirow, 174
 - multirow (pakke), 174
 - \multirowsetup, 174
 - multiline (environment), 88, 94, 95, 97
 - dårligt eksempel, 95
 - \shoveleft, 95
 - \shoveright, 95
 - multiline* (environment), 88, 95
 - multlined (environment), 97
 - \mV (siunitx enhed), 267
- N**
- \nabla (∇), 53
 - \nano (siunitx prefix), 267
 - \nanometre (siunitx enhed), 267
 - natbib (pakke), 199, 202–204, 211–213, 215, 216, 324, 405, 414, 423
 - \cite*, 202
 - \citealias, 203
 - \citealp, 202
 - \citealp*, 203
 - \citealt, 202, 203
 - \citeauthor, 202
 - \citeauthor*, 202
 - \citep, 202, 204
 - \citep*, 202
 - \citepalias, 203
 - \citet, 202–204
 - \citet*, 202
 - \citetalias, 203
 - \citetext, 203
 - \citeyear, 203
 - \citeyearpar, 203
 - \defcitealias, 203
 - pakke options
 - authoryear, 204
 - colon, 203
 - comma, 203
 - longnamesfirst, 204
 - numbers, 204, 211
 - round, 203
 - semicolon, 203

- natbib* (pakke) fortsat...
- sort, 204
 - sort&compress, 204
 - square, 203
 - super, 204, 211
- nath (pakke), 331
- navngivne operatører, 59
- \arccos, 60
 - \arcsin, 60
 - \arctan, 60
 - \arg, 60
 - \cos, 60
 - \cosh, 60
 - \cot, 60
 - \coth, 60
 - \csc, 60
 - definere egne
 - \DeclareMathOperator, 60
 - \DeclareMathOperator*, 60
 - \operatorname, 60
 - \deg, 60
 - \dim, 60
 - \exp, 60
 - \hom, 60
 - \inf, 60
 - \ker, 60
 - \lg, 60
 - \lim, 60
 - \liminf, 60
 - \limsup, 60
 - \ln, 60
 - \log, 60
 - \max, 60
 - \min, 60
 - \Pr, 60
 - \sec, 60
 - \sin, 60
 - \sinh, 60
 - \sup, 60
 - \tan, 60
 - \tanh, 60
- \downarrow, 54
- \nearrow (\nearrow), 55
- \needspace, 255
- needspace (pakke), 255
- \neg (\neg), 53
- negering af symbol, 51
- \negmedspace, 46
- \negthickspace, 46
- \neper (siunitx enhed), 266
- \neq (\neq), 53, 90
- New Century Schoolbook, 234
- \newacronym (glossaries), 259
- \newcite (multibib), 218
- \newcites (multibib), 218
- \newcolumn, 172, 173, 178
- \newcommand, 71, 132, 154, 351–354, 357, 399, 411
- * efter \newcommand, 351
 - med argumenter, 353
 - med standard første argument, 353
 - uden argumenter, 352
 - ændre eksisterende makro, *Se*
 - \renewcommand
- \newcounter, 365
- \newenvironment, 356, 357
- \newglossaryentry (glossaries), 259, 260
- \newlength, 363
- \newline, 86, 148, 170, 171, 223
- \newmdtheoremenv, 128, 130
- \newpage, 22, 185, 299, 320, 334, 347
- \newsavebox, 374
- \NewShadedTheorem, 131
- \newtheorem, 116–118, 121, 122, 125–128, 131–133, 276
- dominerende tæller, 117
 - fælles tæller, 117
 - syntax, 116, 117
 - underlagt anden tæller, 116
- \newtheorem*, 118, 121, 123
- \newtheoremstyle (ntheorem), 123
- \newtoggle (etoolbox), 368
- \newton (siunitx enhed), 266
- \nexists (\nexists), 53
- \NG (D), 30
- \ng (\ng), 30
- \ni (\ni), 53
- \nLeftarrow (\nLeftarrow), 55
- \nleftarrow (\nleftarrow), 55
- \nLeftrightarrow (\nLeftrightarrow), 55
- \nleftrightharrow (\nleftrightharrow), 55
- \nm (siunitx enhed), 267
- \nrightarrow (\nrightarrow), 55
- \nnwarrow (\nnwarrow), 55
- \nobreak, 27, 86, 228
- \nobreakdash, 27
- \nocite, 201, 218
- \nocitesec (multibib), 218
- \noeqref, 104
- \noindent, 22, 152
- noitemsep (enumitem), 243
- \nolimits, 55
- nomenc1 (pakke), 258
- \nonfrenchspacing, 28
- \nonumber, 49, 51, 94, 99
- nonumberbreak (theoremstyle, nthm), 123, 125
- nonumberplain (theoremstyle, nthm), 122, 123, 125
- \noopsort, 215
- normal tekst, 32
- \normalfont, 32, 121, 123, 127

`\normalrulethickness` (*memoir*), 310
`\normalsize`, 31, 121, 237
`\not`, 51
`\notag`, 49, 99
`note` (*BibTeX* felt), 205–207
`Notepad++`, 391
`notheorem` (*beamer*), 276
`notheorems` (*beamer*), 276
`\notin` (\notin), 53
`\nouppercase`, 344
`\nouppercase` (*fancyhdr*), 344
`\nouppercaseheads` (*memoir*), 312
`\nrightarrow` (\nrightarrow), 55
`\rightarrow` (\rightarrow), 55
`ntheorem` (*pakke*), 117, 118, 120–133, 401, 414, 427
 `\newtheoremstyle`, 123
 options
 `amsmath`, 120
 `thmmarks`, 120
 `\theorembodyfont`, 121, 125–127
 `\theoremclass`, 121
 `\theoremheaderfont`, 121, 125–127
 `\theoremindent`, 121
 `\theoremnumbering`, 121
 `\theoremseparator`, 121, 125, 126, 133
 `\theoremstyle`, 121–123, 125–127
 `break`, 122, 123, 132
 `change`, 122
 `changebreak`, 122
 `empty`, 123, 126, 127
 `emptybreak`, 123
 `margin`, 122
 `marginbreak`, 122, 123
 `nonumberbreak`, 123, 125
 `nonumberplain`, 122, 123, 125
 `plain`, 122, 132
 `\theoremsymbol`, 121, 125–127
`\nu` (ν), 52, 231
`\null`, 320
`\num`, 179
`\num` (*siunitx*), 264, 268
 `number` (*BibTeX* felt), 205
`\numberline`, 346
 `numbers` (*natbib* option), 204, 211
`\numberwithin`, 100, 366
 `nummereret liste`, 38
 `numprint` (*pakke*, brug ikke), 416
`\numrange` (*siunitx*), 264
`\nuparrow`, 54
`\nwarrow` (\nwarrow), 55

O

`\O` (\O), 29, 30
`\o` (\o), 29, 30
`\obeylines`, 415
`\odot` (\odot), 53
`\OE` (\OE), 30
`\oe` (\oe), 30
`\ohm` (*siunitx* enhed), 266
`\oiint` (\oiint), 56
`\oint` (\oint), 56
`\ointclockwise` (\oint), 56
`\ointctrlockwise` (\oint), 56
 Okular, 396
`\oldsection`, 146
`\Omega` (Ω), 52
`\omega` (ω), 52
 `onecolumn` (*klasse option*), 11
`\OnehalfSpacing` (*memoir*), 327
`\onelineskip`, 316
 `oneside` (*klasse option*), 11, 326
 `onlyamsmath` (*pakke*), 331
 `opcit` (*pakke*), 197
 `openany` (*klasse option*), 11
`\opening`, 247
 `openright` (*klasse option*), 11
 operatorer
 `navngivne`, 59
 `prædefinerede`, 60
`\operatorname`, 60
 `operatorsymboler`, 53
 `opgave` (*environment*), 127
`\oplus` (\oplus), 53
 `optimering`, 92
 `andet eksempel`, 133
 `orddeling`, 37
 `\-`, 24
 `ingen`, 37
 `organization` (*BibTeX* felt), 205
 `origin` (*graphicx*), 138
`\oslash` (\oslash), 53
 `otherlanguage` (*environment*), 227, 229, 321, 322
 `otherlanguage*` (*environment*), 228
`\otimes` (\otimes), 53
`\overbrace`, 76
`\overbracket`, 76
`\overleftarrow`, 63
`\overline`, 63
 `overpic` (*pakke*), 157
`\overrightarrow`, 63
`\overset`, 55

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
Æ
Ø
Å

- P**
- \P (¶), 30
 - p (søjle specifier), 171–173, 182
 - page (graphicx), 139
 - page (tæller), 365
 - \pagebreak, 334
 - \pageref, 34, 35
 - pages (BIBTEX felt), 205, 208
 - \pagestyle, 42, 310, 313, 340–344
 - pagestyles
 - chapter (memoir), 308–310, 313
 - cleared (memoir), 308
 - empty, 42, 308, 341
 - empty (memoir), 308
 - fancy, 341, 342
 - headings, 310, 312, 341, 342
 - headings (memoir), 309, 312, 326
 - indextitlepagestyle (memoir), 309, 313
 - part (memoir), 308, 313
 - plain, 308, 341
 - plain (memoir), 308, 313
 - title (memoir), 308
 - titlingpage (memoir), 308
 - pakke
 - lave sin egen, 358
 - \makeatletter, 359
 - \makeatother, 359
 - pakker, 12
 - a4 (brug ikke), 416
 - a4wide (brug ikke), 416
 - ae (brug ikke), 417
 - aecompl (brug ikke), 417
 - afterpage, 146, 412
 - algorithm, 285
 - algorithmicx, 283–285, 425
 - algpseudocode, 283
 - amsfonts, 51
 - amsmath, 6, 12, 27, 45, 50, 51, 59, 71, 72, 74, 90, 95, 100–102, 104, 107, 120, 124, 125, 133, 178, 186, 231, 269, 282, 334, 366, 401, 412, 423, 430
 - amssymb, 6, 12, 45, 51, 59, 75, 107, 125, 172, 231, 334, 413
 - amsthm, 115, 117–122, 124, 131, 331, 334
 - animate, 277
 - anysize, 413
 - array, 169–172, 175, 176, 185, 427
 - authblk, 335
 - authoblk, 369
 - babel, 6, 11, 12, 19, 24–28, 38, 110, 119, 216, 227–231, 315, 321, 357, 403, 412
 - babelbib, 216
 - beramono, 236, 412
 - berasans, 236, 412
 - biblatex, 216, 222, 360
 - bibtopic, xiii, 217–219
 - bibunits, 217
 - bigfoot, 225, 226, 249
 - blindtext, 231
 - blkarray, 74
 - bm, 6, 12, 59, 231, 334, 412
 - bookmark, 287
 - booktabs, 174–179, 185, 189, 193, 195, 423
 - boxedminipage, 372
 - breakurl, 286
 - breqn, 68, 269
 - calc, 364, 365, 400, 412
 - caption, 150–154, 169, 188, 412, 414
 - cellspace, 179, 186
 - chapterbib, 217, 323
 - chngcntr, 101, 366
 - chngpage, 191
 - cite, 199, 204, 421
 - cleveref, 298, 299
 - color, 245, 412, 422, 425
 - colortbl, 193
 - comment, 295, 412
 - contmath, 235
 - cool, 352
 - csquotes, 25, 412
 - datatool, 194, 195, 248
 - dcolumn, 71, 177–181
 - dk-apali, 213
 - dk-bib, 210, 211, 216
 - dlfltxbcodetips, xviii, 413
 - dlfltxbmarkup, 254, 412
 - dlfltxbmarkupbookkeys, 254
 - dlfltxbmisc, 412
 - dlfltxbtocconfig, 413
 - ellipsis, 29
 - empheq, 76, 87, 106, 127, 425
 - endnotes, 226
 - enumerate, 40, 239, 240, 242
 - enumitem, 39, 41, 190, 239, 240, 242, 245, 413
 - epsfig, 136
 - epsfig (brug ikke), 416
 - esint, 56
 - eso-pic, 141, 156
 - etex, 413
 - etoolbox, xi, 359, 360, 367, 369
 - eulervm, 233, 417
 - fancyhdr, 301, 308, 341–345, 430
 - fancyref, 298, 299, 428
 - fancyvrb, 249, 250, 403
 - fix-cm, 237, 413
 - fixltx2e, 33, 413
 - fixme, xviii, 295–297, 413

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
Æ
Ø
Å

pakker fortsat...

- flafter, 146
- float, 146
- floatflt, 155
- fontenc, 6, 12, 13, 19, 24, 26, 29, 252, 413, 416
- footmisc, 225, 226
- fourier, 36, 233, 234, 318, 330
- fouriernc, 234
- framed, 193, 225, 246
- gauss, 74
- geometry, 306, 307, 324, 337–340, 364
- glossaries, 258, 259, 261, 262, 416
- glossary (brug ikke), 416
- graphicx, xii, 136, 154, 156, 306, 373, 413, 416, 422
- grffile, 140
- harvard, 199, 430
- helvet, 235
- hyperref, 43, 254, 261, 285–287, 300, 329, 354, 379, 402, 413
- ifpdf, 140, 263
- ifsym, 413
- indentfirst, 24
- inputenc, 6, 12, 13, 15, 19, 29, 231, 387, 390, 399, 413, 416
- isolatin1 (brug ikke), 416
- jurabib, 197, 208, 214–216
- keyval, 353, 413
- kpfonts, 56, 234, 413
- latexsym (brug ikke), 417
- lipsum, 230, 231, 413
- listings, 250–252, 273, 333, 413, 425, 426
- lmodern, 27, 232, 237
- longtable, 183, 185, 188, 189, 413
- lscape, 192
- luximono, 236, 252
- mailing, 247
- makeidx, 230, 253
- mathdesign, 234
- mathdots, 62
- mathpazo, 233, 417
- mathptm (brug ikke), 417
- mathptms, 233
- mathptmx, 232, 417
- mathtools, 6, 12, 54–56, 64–66, 69, 71, 72, 75, 76, 87, 94, 95, 97, 100, 102–106, 333, 334, 385, 413
- mdframed, 128–130, 194, 246, 413
- mflogo, 320, 413
- mhchem, 281, 282
- microtype, 413
- MnSymbol, 54
- multibib, xiii, 217–219
- multicol, 11, 413
- multirow, 174
- natbib, 199, 202–204, 211–213, 215, 216, 324, 405, 414, 423
- nath, 331
- needspace, 255
- nomencl, 258
- ntheorem, 117, 118, 120–133, 401, 414, 427
- numprint (brug ikke), 416
- onlyamsmath, 331
- opcit, 197
- overpic, 157
- palatino, 233, 417
- palatino (brug ikke), 417
- paralist, 239, 240, 414
- pdfpages, 141, 142, 156, 324
- pdftricks, 162
- pgfpages, 277, 278
- picture, 157
- placeins, 145, 146, 355, 414
- printlen, 364
- psfig, 136
- psfig (brug ikke), 416
- psfrag, 155, 156, 424
- pslatex (brug ikke), 417
- ragged2e, 37, 171, 414
- rotating, 191, 192
- SASdisplay, 252, 253, 273, 281
- sectsty, 223
- setspace, 238, 337
- showidx, 298
- showkeys, 297, 298
- sinutx, 266
- sistyle (brug ikke), 416
- siunit (brug ikke), 416
- siunitx, 30, 177, 179–182, 186, 195, 263–266, 268, 379, 416
- soul, 238, 239, 316, 318, 414, 424
- stmaryrd, 51, 55, 56, 65, 414
- subfig, 154, 414, 416, 423
- subfigure (brug ikke), 154, 416, 423
- svninfo, 271
- svn-multi, 269–271, 414
- Sweave, 280, 281
- t1enc (brug ikke), 416
- tabularx, 182
- textcomp, 29, 414
- thmtools, 118
- threeparttable, 152, 179, 188, 414
- threeparttablex, 189, 414
- tikz, 112, 414
- times (brug ikke), 417
- titlesec, 223, 347, 422
- tocbibind, 347
- tocloft, 224, 347
- todonotes, 295

#

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

Æ

Ø

Å

pakker fortsat...

- truncate, 345
- typearea, 306
- url, 9, 42, 195, 207, 286, 401, 414, 421
- ushort, 65
- varioref, 11, 142, 286, 298, 299, 403, 414
- varwidth, 372
- verbatim, 250
- verbatimcopy, 414
- wrapfig, 155
- xcolor, 130, 193, 245, 246, 270
- xifthen, 367
- xkeyval, 359, 422
- xparse, 130, 353, 357, 359, 414
- xr, 299, 300
- xr-hyper, 300
- xspace, 355, 414
- xy, 90, 107–112, 428
- Palatino, 233
- palatino (pakke), 233, 417
- palatino (pakke, brug ikke), 417
- \paperheight (længde), 156, 365
- \paperwidth (længde), 156, 365
- \par, 316, 320, 411
- \paragraph, 23, 36, 39, 240
- paralist (pakke), 239, 240, 414
- \parallel (||), 53
- \parbox, 148, 246, 364, 371, 372, 400
- \parsep (længde), 132
- \part, 23, 36, 276, 287, 304, 306, 308, 313, 314, 346
- part (pagestyle, memoir), 308, 313
- \partial, 53, 71
- \partname (sprogspecifikt navn), 230
- \pascal (siunitx enhed), 266
- \path, 9, 42, 43, 401
- \pause (beamer), 275
- pdf
 - dvipdfmx, 137
 - flette filer sammen, 142
 - pdflatex, 136
 - pdf (grafik format), 136, 161
 - pdf2ps (program), 160
 - pdfcrop (program), 156, 159, 161
 - pdffonts (program), 396
 - pdfinfo (program), 396
 - pdflatex (program), 7, 19, 108, 136, 137, 139–141, 159, 163, 164, 286, 287, 324, 379, 388
 - pdfpages (pakke), 141, 142, 156, 324
 - flette filer sammen, 142
 - pdftex (program), 136, 137
 - pdftk (program), 397
 - pdftops (program), 160, 161
 - pdftricks (pakke), 162
 - PDF-Xchange Viewer, 396
 - \percent (siunitx enhed), 266
 - \perp (\perp), 53, 54
 - \peta (siunitx prefix), 267
 - \pgfpagelayout (beamer), 278
 - pgfpages (pakke), 277, 278
 - \pgfpagesdeclarelayout, 278
 - \pgfuselayout, 277
 - \phantom, 99, 104
 - \phantomsection, 286
 - phdthesis (BIBTEX type), 205
 - \Phi (Φ), 52
 - \phi (ϕ), 52
 - \Pi (Π), 52
 - \pi (π), 47, 52
 - \pico (siunitx prefix), 267
 - picture (environment), 156, 270, 365
 - picture (pakke), 157
 - pil
 - skrive over eller under, 54
 - pile
 - skrive over, 54
 - placeins (pakke), 145, 146, 355, 414
 - plain (BIBTEX stil), 200, 208
 - plain (pagestyle), 308, 341
 - plain (pagestyle, memoir), 308, 313
 - plain (theoremstyle, nthm), 122, 132
 - plainnat (BIBTEX stil), 211
 - \pm (\pm), 53
 - pmatrix (environment), 71, 74, 400
 - \pmod, 70
 - png
 - dvipdfmx, 137
 - pdflatex, 136
 - png (grafik format), 136
 - \pod, 70
 - PostScript (grafik format), 137
 - powerdot (klasse), 272
 - ppower4 (program), 278
 - \Pr (mat. operator), 60
 - preamble, 10
 - preamblen, 25
 - \prec (\prec), 53
 - \preceq (\preceq), 53
 - preview (program), 393
 - \prime (\prime), 53, 54, 71
 - printe B5 på A4-papir, 339
 - \printindex, 254, 294
 - printlen (pakke), 364
 - problem
 - EPS til PDF via ps2pdf, 159
 - JPG til EPS
 - fil bliver meget stor, 160
 - problem (environment), 133
 - \Procedure, 284, 285
 - proceedings (BIBTEX type), 205

- \backslash prod (\prod), 56
 - programmer
 - biber, 222
 - bibtex, 200, 204, 205, 219, 222, 256, 323, 379
 - bibtex8, 205, 216
 - convert, 160, 161, 398
 - dvipdfm, 430
 - dvipdfmx, 136, 137, 306
 - dvipng, 235
 - dvips, 6, 8, 135–137, 139, 155, 159, 161–164, 286
 - ebb, 137
 - eps2eps, 157
 - epstool, 158, 159
 - epstopdf, 158–160
 - evince, 6
 - Excel, 167
 - excel2latex, 194
 - getnonfreefonts, 252, 383
 - getnonfreefonts-sys, 383
 - ghostscript, 160
 - Gnumeric, 194
 - gnuplot, 163
 - gsview, 6
 - htlatex, 409
 - hunspell, 386, 387
 - Imagemagick, 398
 - jabref, 217, 220, 221
 - jpeg2eps, 398
 - jpeg2ps, 160, 398
 - latex, 8, 19, 137, 139, 140, 155, 163, 200, 219, 256, 286
 - latexmk, 256, 257, 379, 385
 - makebst, 213, 216, 423
 - makeglossaries, 259, 262, 385
 - makeindex, ix, 255–258, 379, 415
 - mkind, 255–257
 - mpost, 162
 - pdf2ps, 160
 - pdfcrop, 156, 159, 161
 - pdffonts, 396
 - pdfinfo, 396
 - pdflatex, 7, 19, 108, 136, 137, 139–141, 159, 163, 164, 286, 287, 324, 379, 388
 - pdftex, 136, 137
 - pdftk, 397
 - pdftops, 160, 161
 - ppower4, 278
 - preview, 393
 - ps2pdf, 6, 8, 137, 159, 160, 286
 - R, 158, 280
 - sam2p, 161
 - texcount, 271
 - texdoc, 17, 379
 - texindy, ix, 255, 257
 - tlmgr, 379, 380, 383
 - xdvi, 391
 - xindy, 254–256
 - xpdf, 160
 - yap, 391
 - proof (environment), 115, 119, 124–126, 275, 276
 - anden overskrift, 119
 - proof (environment, amsthm), 119
 - \backslash proofname, 119
 - proofof (environment), 126
 - \backslash propto (\propto), 53
 - prosper (klasse), 272
 - \backslash protect, 347
 - \backslash providecommand, 143, 354
 - \backslash ProvidesPackage, 358
 - ps2pdf (program), 6, 8, 137, 159, 160, 286
 - pseudokode, 283
 - \backslash psfig, 416
 - psfig (pakke), 136
 - psfig (pakke, brug ikke), 416
 - psfrag (pakke), 155, 156, 424
 - \backslash Psi (Ψ), 52
 - \backslash psi (ψ), 52
 - pslatex (pakke, brug ikke), 417
 - psmallmatrix (environment), 72
 - publisher (BIBTEX felt), 205, 206, 208
 - punktopstilling, 38
 - \backslash put, 156, 157
- Q**
- \backslash qed, 120
 - \backslash qedhere, 119, 331
 - \backslash qqquad, 46, 58, 74, 96, 97, 139, 147, 154, 175, 330
 - \backslash quad, 22, 46, 47, 58, 84, 92, 96, 121, 156, 171, 183, 186
 - quotation (environment), 38, 357
 - quote (environment), 38, 357
- R**
- \backslash r (\r), 29, 30
 - R (program), 158, 280
 - r (søjle specifier), 168, 173, 175, 176, 183
 - \backslash radian (siunitx enhed), 266
 - ragged2e (pakke), 37, 171, 414
 - \backslash raggedbottomsectiontrue (memoir), 314
 - \backslash RaggedLeft, 38, 171, 229
 - \backslash raggedleft, 37, 171, 229, 414
 - \backslash RaggedRight, 38, 171, 229
 - \backslash raggedright, 37, 171, 174, 182, 229, 414
 - \backslash raisebox, 373
 - \backslash raisetag, 101
 - \backslash rangle (hegn), 65, 103
 - \backslash rceil (hegn), 65

\Re (℞), 53, 60
 \reDefineSequence, 361
 \ref, 34, 35, 40, 50, 100, 133, 154, 298
 ref (enumitem), 242
 referencer
 vis kun refererede formler, 104
 \reflectbox, 373
 \refname (sprogspecifikt navn), 230
 \refstepcounter, 365, 366
 relationssymboler, 53
 \relax, 183
 remark (environment), 118
 \renewcommand, 33, 51, 86, 144, 146, 174, 189, 224, 229, 335, 341, 343, 344, 354
 \renewenvironment, *Se* \newenvironment
 \Repeat, 285
 report (klasse), 11, 12, 42, 230, 305, 341
 reqno (klasse option), 11
 \Require, 284
 \resizebox, 373
 resume* (enumitem), 243
 revtex4 (klasse), 12
 \rfloor (hegn), 65
 \rgroup (hegn), 65
 \rho (ρ), 52
 \right, 49, 66–68, 70, 71, 73, 77, 103, 332, 410
 \Rightarrow (⇒), 55, 67, 96
 \rightarrow (→), 55
 \rightharpoondown (↘), 55
 \rightharpoonup (↗), 55
 \rightleftarrows (⇔), 55
 \rightleftharpoons (⇌), 55
 \rightmark, 309–312, 326, 343, 344
 \rlap, 111, 112
 \rmfamily, 32
 \rmodels, 54
 \Roman, 366
 \roman, 366
 \Roman* (enumitem), 242
 \roman* (enumitem), 242
 \rotatebox, 135, 373, 374
 rotating (pakke), 191, 192
 round (natbib option), 203
 \rowops (gauss), 74
 \rrbracket (hegn), 65
 \rule, 125
 \rVert (hegn), 65, 66
 \rvert (hegn), 65, 66, 103

S

\S (§), 30
 \s (siunitx enhed), 267
 S (søjle specifier), 173, 179, 186
 sam2p (program), 161

sans serif, 32
 SASdisplay (pakke), 252, 253, 273, 281
 \savebox, 374
 \sbox, 374, 375
 \sc (brug ikke), 32
 scale (graphicx), 138
 \scalebox, 135, 373
 school (BIBTEX felt), 205
 Scientific Workplace, 392, 392, 393
 scope, 356
 scrartcl (klasse), 12, 305
 scrbook (klasse), 12, 305
 \scriptscriptstyle, 57
 \scriptsize, 31
 \scriptstyle, 112, 125
 scrlettr (klasse), 305
 scrlettr2 (klasse), 12
 sclltr2 (klasse), 247
 sclltr2 (klasse), 12, 22, 305
 scrreprt (klasse), 12, 305
 \scshape, 32, 33
 \searrow (↙), 55
 \sec (mat. operator), 60
 secnumdepth (tæller), 314, 348
 \second (siunitx enhed), 266, 267
 \section, 23, 24, 34, 36, 39, 100, 146, 208, 223, 225, 247, 276, 306, 311, 313, 314, 321, 330, 332, 343–346, 353, 355, 369, 370, 400, 404
 section (chapter style), 326
 section (tæller), 116
 \section*, 24, 36
 \sectionmark, 311, 343, 344
 sectsty (pakke), 223
 \see, 254
 \seename (sprogspecifikt navn), 230
 \selectfont, 237
 \selectlanguage, 227, 228, 322
 semicolon (natbib option), 203
 series (BIBTEX felt), 205, 208
 \setbeamertemplate, 277
 \setcounter, 72, 346, 365
 \setdescription (enumitem), 244
 \setenumerate (enumitem), 244
 \setFloatBlockFor (memoir), 146
 \setitemize (enumitem), 244
 \setlength, 132, 306, 363, 364, 370, 412, 429
 \setlist (enumitem), 244
 \setlrmargins (memoir), 308
 \setlrmarginsandblock (memoir), 308
 \setminus (\), 9, 53
 \setpnumwidth (memoir), 315
 \setR, 351
 \setrmarg (memoir), 315
 \setseheadstyle (memoir), 314

 A
 B
 C
 D
 E
 F
 G
 H
 I
 J
 K
 L
 M
 N
 O
 P
 Q
 R
 S
 T
 U
 V
 W
 X
 Y
 Z
 Æ
 Ø
 Å

`\setsecnumdepth` (*memoir*), 314, 315
`setspace` (*pakke*), 238, 337
`\setsubseheadstyle` (*memoir*), 314
`\setsubsubseheadstyle` (*memoir*), 314
`\settocdepth` (*memoir*), 314, 315, 346
`\settodepth`, 363
`\settoheight`, 363
`\settotalheightof`, 364
`\settowidth`, 363
`\settypeblocksizes` (*memoir*), 308
`\setulmargins` (*memoir*), 308
`\setulmarginsandblock` (*memoir*), 308
`\sffamily`, 32, 121, 235, 312
`shadecolor`, 246
`shaded` (*environment*), 193, 225, 246
`\shorthandoff`, 110, 111
`\shorthandon`, 110, 111
`\shortintertext`, 94, 98, 333
`shorttitle` (*BibTeX felt*), 208
`\shoveleft`, 95
`\shoveright`, 95
`showidx` (*pakke*), 298
`showkeys` (*pakke*), 297, 298
`\SI` (*siunitx*), 264, 265, 268
`\si` (*siunitx*), 264, 265, 268
`sidetal`
 fjerne, 341
`sideways` (*environment*), 191, 192
`sidewaysfigure` (*environment*), 191
`sidewaystable` (*environment*), 191
`\siemens` (*siunitx enhed*), 266
`\sievert` (*siunitx enhed*), 266
`\Sigma` (Σ), 52
`\sigma` (σ), 52
`\sim`, 9, 53, 352
`\simeq` (\simeq), 53, 57
`\sin`, 48, 59, 60
`\SingleSpacing` (*memoir*), 327
`\sinh` (*mat. operator*), 60
`sinutx` (*pakke*), 266
`\SIrange`, 264
`\sisetup`, 182
`sistyle` (*pakke, brug ikke*), 416
`siunit` (*pakke, brug ikke*), 416
`siunitx` (*pakke*), 30, 177, 179–182, 186, 195, 263–266, 268, 379, 416
 enhed
 `\ampere`, 266
 `\angstrom`, 266
 `\arcminute`, 266
 `\arcsecond`, 266
 `\atomicmassunit`, 266
 `\bar`, 266
 `\barn`, 266
 `\becquerel`, 266
 `\bel`, 266

`\candela`, 266
`\celsius`, 266
`\centimetre`, 267
`\coulomb`, 266
`\day`, 266
`\decibel`, 266
`\degree`, 266
`\degreeCelsius`, 266
`\electronvolt`, 266
`\farad`, 266
`\gigahertz`, 267
`\gray`, 266
`\hectare`, 266
`\henry`, 266
`\hertz`, 266, 267
`\hour`, 266
`\joule`, 266
`\katal`, 266
`\kelvin`, 266
`\kg`, 266
`\kiloampere`, 267
`\kiloelectronvolt`, 267
`\kilogram`, 266, 267
`\kilohertz`, 267
`\kilojoule`, 267
`\kilometre`, 267
`\kilovolt`, 267
`\liter`, 266
`\litre`, 266
`\lumen`, 266
`\lux`, 266
`\megaelectronvolt`, 267
`\megahertz`, 267
`\meter`, 266
`\metre`, 266
`\MHz`, 266
`\micrometre`, 267
`\microsecond`, 267
`\milliampere`, 267
`\millielectronvolt`, 267
`\milligram`, 267
`\millihertz`, 267
`\millilitre`, 267
`\millimetre`, 267
`\millisecond`, 267
`\millivolt`, 267
`\minute`, 266
`\mmHg`, 266
`\mole`, 266
`\nanometre`, 267
`\neper`, 266
`\newton`, 266
`\ohm`, 266
`\pascal`, 266
`\percent`, 266
`\radian`, 266

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
Æ
Ø
Å

<i>siunitx</i> (pakke) fortsat...		
\second, 266, 267	\peta, 267	
\siemens, 266	\pico, 267	#
\sievert, 266	\tera, 267	A
\steradian, 266	\yocto, 267	B
\tesla, 266	\yotta, 267	C
\tonne, 266	\zepto, 267	D
\volt, 266	\zetta, 267	E
\watt, 266	\SI, 264, 265, 268	F
\weber, 266	\si, 264, 265, 268	G
forkortet enhed	skabeloner,	H
\cm, 267	Skim, 395	I
\GHz, 267	slantet, 32	J
\Hz, 267	\slshape, 32	K
\kA, 267	slutnoter, 35	L
\keV, 267	\small, 31, 238, 312, 320, 363	M
\kg, 267	small caps, 32	N
\kHz, 267	smallmatrix (environment), 70–72	O
\kJ, 267	\smash, 57, 64, 352	P
\km, 267	\smashoperator, 102	Q
\kV, 267	\so, 239	R
\mA, 267	\sodef, 239	S
\MeV, 267	sort (natbib option), 204	T
\meV, 267	sort&compress (natbib option), 204	U
\mg, 267	soul (pakke), 238, 239, 316, 318, 414, 424	V
\MHz, 267	problem med UTF8, 239	W
\mHz, 267	spacing, 46	X
\mL, 267	_, 46	Y
\mm, 267	em, 46	Z
\ms, 267	\enskip, 46	Æ
\mV, 267	math unit, 46	Ø
\nm, 267	medspace (\:), 46	Å
\s, 267	mu, <i>Se</i> math unit	
\um, 267	\negmedspace, 46	
\us, 267	\negthickspace, 46	
\num, 264, 268	\quad, 46	
\numrange, 264	\quad, 46	
potens	thickspace (\;), 46	
\cubed, 267	thinspace (\,), 46	
\cubic, 267	\spadesuit (♠), 53	
\square, 267	\Span, 354	
\squared, 267	\span, 61, 354	
prefix	\specialrule (booktabs), 175	
\atto, 267	Specialtegn, 29	
\centi, 267	\sphericalangle (<), 53	
\deca, 267	split (environment), 95, 97, 99, 101, 400	
\deci, 267	\splitdffrac (mathtools), 69	
\exa, 267	\splitfrac (mathtools), 69	
\femto, 267	sprogspecifikt navn	
\giga, 267	\abstractname, 230	
\hecto, 267	\alsoname, 230	
\kilo, 267	\appendixname, 230	
\mega, 267	\bibname, 230	
\micro, 267	\chaptername, 230	
\milli, 267	\contentsname, 230	
\nano, 267	\figurename, 230	
	\indexname, 230	

sprogspecifikt navn fortsat...

- \listfigurename, 230
- \listtablename, 230
- \partname, 230
- \refname, 230
- \seename, 230
- \tablename, 230
- \sqcap (\sqcap), 53
- \sqcup (\sqcup), 53
- \sqiint (\sqiint), 56
- \sqint (\sqint), 56
- \sqrt, 50, 70
- \sqsubset (\sqsubset), 53
- \sqsubseteq (\sqsubseteq), 53
- \sqsupset (\sqsupset), 53
- \sqsupseteq (\sqsupseteq), 53
- \square (\square), 53, 121
 - square (natbib option), 203
- \square (siunitx prefix), 267
- \squared (siunitx prefix), 267
- \searrow (\searrow), 55
- \swarrow (\swarrow), 55
- \stackrel, 55, 57, 64, 352
 - standardnavne i L^AT_EX, 230
 - standardnavne i L^AT_EX, 229
- \star (\star), 53
- \starbreak, 330
- \starredbullet, 330
- start (enumitem), 243
- \State, 284
- \Statex, 284
- \stepcounter, 365
- \steradian (siunitx enhed), 266
- stikordsregister, 253
 - !, 254
 - @, 254
 - |, 254
- stmaryrd (pakke), 51, 55, 56, 65, 414
- store symboler
 - \bigbox (\bigbox), 56
 - \bigcap (\bigcap), 56
 - \bigcup (\bigcup), 56
 - \biginterleave (\biginterleave), 56
 - \bigoplus (\bigoplus), 56
 - \bigotimes (\bigotimes), 56
 - \bigparallel (\bigparallel), 56
 - \bigsqcap (\bigsqcap), 56
 - \bigsqcup (\bigsqcup), 56
 - \bigvee (\bigvee), 56
 - \bigwedge (\bigwedge), 56
 - \coprod (\coprod), 56
 - \dotsint (\dotsint), 56
 - \iiiint (\iiiint), 56
 - \iiint (\iiint), 56
 - \iint (\iint), 56

- \int (\int), 55, 56
- \oiint (\oiint), 56
- \oint (\oint), 56
- \ointclockwise (\oint), 56
- \ointctrlockwise (\oint), 56
- \prod (\prod), 56
- \sqiint (\sqiint), 56
- \sqint (\sqint), 56
- \sum (\sum), 45, 55, 56, 76, 102
- streger, 26
- stregtegning
 - problemer med, 161
- \stretch, 321, 370, 371
- style (enumitem), 244
- subappendices (env, memoir), 224
- subequations (environment), 76, 101, 133
- subfig (pakke), 154, 414, 416, 423
 - \ContinuedFloat, 155
 - \subfloat, 154
 - \subref, 154
 - \subref*, 154
- subfigure (pakke, brug ikke), 154, 416, 423
- \subfloat, 154
- \subfloat (subfig), 154
- \subparagraph, 23
- \subref (subfig), 154
- \subref* (subfig), 154
- subscript, 68
- \subsection, 23, 36, 276, 312
- \subsection*, 24
- \subsectionmark, 343
- \subset (\subset), 53
- \subseteq (\subseteq), 53
- \subseteqq (\subseteqq), 53
- \subsetneq (\subsetneq), 53
- \subsetneqq (\subsetneqq), 53
- \substack, 76, 77
- \subsubsection, 23, 306, 312, 314, 330, 343, 345, 347, 348
- \subtitle (beamer), 274
- \succ (\succ), 53
- \sum (\sum), 45, 55, 56, 76, 102
 - Sumatra PDF, 394, 394, 395
- \sup (mat. operator), 60
 - super (natbib option), 204, 211
 - superscript, 68
- \suppressfloats, 146
- \supset (\supset), 53
- \supseteq (\supseteq), 53
- \supseteqq (\supseteqq), 53
- \supsetneq (\supsetneq), 53
- \supsetneqq (\supsetneqq), 53
 - svg (grafik format), 161
- \svnfiledate, 270
- \svnfileday, 270

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
Æ
Ø
Å

- \svnfilehour, 270
 - \svnfileminure, 270
 - \svnfilemonth, 270
 - \svnfilerrev, 270
 - \svnfileyear, 270
 - \svnidlong, 270
 - svninfo (pakke), 271
 - svn-multi (pakke), 269–271, 414
 - \swap (gauss), 74
 - \swarrow (↙), 55
 - Sweave, 280
 - Sweave (pakke), 280, 281
 - symboler
 - diverse, 53
 - inklusioner, 53
 - pile, 55
 - store, 56
 - symboloversigt
 - lave en, 172
 - sætninger
 - angive ekstra overskift, 117
 - nummereret efter afsnit eller kapitel, 116
 - nummereret samlet, 117
 - selv bestemme overskift, 127
 - søjle specificier
 - A, 173
 - C, 179
 - c, 168, 171–174, 176, 179, 182, 186
 - D, 178, 179
 - d, 178
 - l, 168, 172, 173, 175, 176, 188, 191, 192
 - p, 171–173, 182
 - r, 168, 173, 175, 176, 183
 - S, 173, 179, 186
 - X, 182, 183
- T**
- \t (bind efter), 30
 - tlenc (pakke, brug ikke), 416
 - \tabcolsep (længde), 186
 - tabel
 - beskyttede overskriftcelle, 173
 - celler
 - spænde over flere søjler, Se \multicolumn
 - linier i, 174
 - preamble, 169
 - søjle
 - A, 173
 - b (array), 170
 - C, 179
 - c, 168, 169, 171–174, 176, 179, 182, 186
 - D, 178, 179
 - d, 178
 - l, 168, 169, 172, 173, 175, 176, 188, 191, 192
 - m (array), 170
 - p, 169, 171–173, 182
 - r, 168, 169, 173, 175, 176, 183
 - S, 173, 179, 186
 - X, 182, 183
 - vandrette linier med booktabs, 174
 - tabeller
 - celler
 - spænde over flere rækker, 174
 - tabelpreamble, 168
 - genvej til, 172
 - table (environment), 142, 152, 168, 183, 186, 188, 192, 285, 400
 - table*
 - floats i to søjler, 143
 - \tablename (sprogspecifikt navn), 230
 - TableNotes (environment), 189
 - tablenotes (environment), 188, 189
 - \tableofcontents, 36, 294, 311, 344, 345
 - \tableofcontents*, 315
 - tabular (environment), 74, 168, 169, 171–176, 178, 179, 182, 183, 186, 188, 191, 371
 - tabularx (environment), 182, 183, 186, 187
 - tabularx (pakke), 182
 - \tag, 99, 100
 - \tag*, 99, 100
 - \tan (mat. operator), 60
 - \tanh (mat. operator), 60
 - \tau (τ), 52
 - \tbinom, 70
 - TDS-struktur, 379
 - tekst
 - uden L^AT_EX fortolkning, 249
 - \@tempdima (længde), 358, 365
 - \@tempdimb (længde), 365
 - \@tempdimc (længde), 365
 - \tera (siunitx prefix), 267
 - \tesla (siunitx enhed), 266
 - teste efter PDF-output, 140
 - \TeX, 4
 - texcount (program), 271
 - texdoc (program), 17, 379
 - texindy (program), ix, 255, 257
 - T_EX Live
 - installation på Linux, 381
 - Installation på Mac, 377
 - installation på Windows, 378
 - opdatering Linux, 383
 - opdatering Mac, 378
 - opdatering Windows, 379
 - TeXmacs, 391
 - TeXMaker, 388

- TeXMaker (editor), 381, 388, 389
 TeXmaker (editor), 392
 texmaker (editor), 380
 TeXnicCenter, 389
 TeXnicCenter (editor), 14, 387–389
 texniccenter (editor), 384, 385
 \texorpdfstring, 287, 354, 402
 TeXShop, 389, 389, 390
 TeXShop (editor), 14, 389, 390
 texshop (editor), 378
 TeXstudio, 389
 TeXstudio (editor), 389
 \text, 57–59, 67, 73, 75, 76, 92, 178, 258, 268, 399
 \textasciicircum, 9
 \textasciitilde, 9
 \textbackslash, 9
 \textbf, 32, 119
 \textcelsius (°C, textcomp), 30
 \textcolor, 246
 textcomp (pakke), 29, 414
 \copyright (©), 30, 320
 \textcelsius (°C), 30
 \textdagger (†), 30
 \textdegree (°), 30
 \texteuro (€), 30
 \textmu (μ), 30
 \textparagraph (¶), 30
 \textperthousand (‰), 30
 \textregistered (®), 30
 \textsection (§), 30
 \textsterling (£), 30
 \texttrademark (™), 30
 \textdagger, 188
 \textdagger (†, textcomp), 30
 \textdegree (°, textcomp), 30
 \textdollar, 9
 \texteuro (€, textcomp), 30
 \textfloat (længde), 144
 \textfraction, 144
 \textheight (længde), 138, 365, 415
 \textit, 32, 33, 50, 58, 179
 \textmd, 32
 \textmu (μ, textcomp), 30
 \textnormal, 32, 119, 127
 \textparagraph (¶, textcomp), 30
 \textperthousand (‰, textcomp), 30
 \textregistered (®, textcomp), 30
 \textrm, 32, 268
 \textsc, 32
 \textsection (§, textcomp), 30
 \textsf, 32, 235
 \textsl, 32
 \textsterling (£, textcomp), 30
 textstyle, 45, 75
 \texttrademark (™, textcomp), 30
 \texttt, 32, 250
 \textunderscore, 9
 \textup, 32, 59, 268, 269, 399, 416
 \textvisiblespace (␣), xviii
 \textwidth (længde), 138, 148, 153, 182, 310, 365, 415
 TeXworks, 386–388
 TeXWorks (editor), 14
 TeXworks (editor), 386–389
 texworks (editor), 378, 380
 \tfrac, 66, 69
 \TH (Þ), 30
 \th (þ), 30
 \thanks, 42, 225, 226
 \the, 364
 thebibliography (environment), 10, 197–201, 311
 \thechapter, 329
 \theendnotes, 226
 theorem (environment), 276
 \theorembodyfont (ntheorem), 121, 125–127
 \theoremclass (ntheorem), 121
 \theoremheaderfont (ntheorem), 121, 125–127
 \theoremindent (ntheorem), 121
 \theoremnumbering (ntheorem), 121
 Alph, 121
 alph, 121
 arabic, 121
 Greek, 121
 greek, 121
 Roman, 121
 roman, 121
 \theoremseparator (ntheorem), 121, 125, 126, 133
 \theoremstyle, 118, 132
 \theoremstyle (amsthm), 118
 definition, 118
 plain, 118
 remark, 118
 \theoremstyle (ntheorem), 121–123, 125–127
 break, 122, 123, 131, 132
 change, 122
 changebreak, 122
 empty, 123, 126, 127
 emptybreak, 123
 margin, 122
 marginbreak, 122, 123
 nonumberbreak, 123, 125
 nonumberplain, 122, 123, 125
 plain, 122, 132
 \theoremsymbol (ntheorem), 121, 125–127
 \thepage, 270, 310, 312, 343, 344
 \thesection, 366
 \Theta (Θ), 52

\theta (θ), 48, 52
 thickspace ($\backslash;$), 46
 \thinspace, 86
 thinspace ($\backslash,$), 46
 \thispagestyle, 42, 159, 309, 320, 321, 327, 340, 341
 thm (environment), 116–118, 121, 123, 125, 132
 thm (tæller), 117, 118, 121, 123, 132
 thmbreak (environment), 132
 thmtools (pakke), 118
 ThreePartTable (environment), 189
 threeparttable (environment), 188, 189
 threeparttable (pakke), 152, 179, 188, 414
 threeparttablex (pakke), 189, 414
 tie, 9
 \tightlist, 242, 354
 \tightlist (memoir), 239, 354
 \tightlists (memoir), 239
 tikz (pakke), 112, 414
 \tilde, 63
 Tilføj folder til \TeX Live , 379
 Times, 232
 \times (\times), 53, 86, 178
 times (pakke, brug ikke), 417
 \tiny, 31
 \title, 41, 42, 274
 title (Bib \TeX felt), 205, 206, 208
 title (pagestyle, memoir), 308
 \titlepage (beamer), 274
 \titleref, 35
 \titleref (memoir), 330
 titlesec (pakke), 223, 347, 422
 titlingpage (env, memoir), 308, 320
 titlingpage (pagestyle, memoir), 308
 tæller
 bottomnumber, 144
 chapter, 100, 116, 365
 enumi, 243
 enumii, 243
 equation, 100
 footnote, 225
 lastsheet (memoir), 327
 MaxMatrixCols, 72
 page, 365
 secnumdepth, 314, 348
 section, 116
 thm, 117, 118, 121, 123, 132
 tocdepth, 314, 345, 348
 topnumber, 144
 totalnumber, 144
 tlmgr (program), 379, 380, 383
 \tnote, 188, 189
 \tnotex, 189
 \to (\rightarrow), 55, 93
 tocibind (pakke), 347

tocdepth (tæller), 314, 345, 348
 tocloft (pakke), 224, 347
 \tocmark (memoir), 311
 \today, 41, 42
 todonotes (pakke), 295
 \togglefalse (etoolbox), 368
 \toggletrue (etoolbox), 368
 \tonne (siunitx enhed), 266
 \top (\top), 53
 \topfraction, 144, 145
 topnumber (tæller), 144
 \toprule (booktabs), 175, 176, 178, 179, 182, 183, 195
 totalheight (graphicx), 138, 400
 \totalheight (længde), 373
 \totalheightof, 364
 totalnumber (tæller), 144
 \TPTminimum, 189
 \tr, 60
 trans (beamer), 278
 \triangle (Δ), 53
 \triangledown (∇), 53
 trim (graphicx), 139, 363
 truncate (pakke), 345
 \ttfamily, 32
 tuborg
 horisontal, 76
 tvunget sideskift
 indholdsfortegnelse, 347
 twocolumn (klasse option), 11
 \twoheadleftarrow (\leftarrow), 55
 \twoheadrightarrow (\rightarrow), 55
 twoside (klasse option), 11, 324, 326, 339, 340
 TXTT, 236
 type (Bib \TeX felt), 205
 typearea (pakke), 306
 typesetting engine, 4

U
 \u (accent breve), 30
 uafhængighedssymbol (\perp), 54
 u.bb., 92
 \um (siunitx enhed), 267
 \uncover (beamer), 275
 \underbrace, 76
 \underbracket, 76
 underfull
 \hbox, 404
 \vbox, 404
 \underleftarrow, 63
 \underleftrightharrow, 63
 \underline, 63, 65
 undernummerering, 101
 \underrightarrow, 63
 unicode, 13

 A
 B
 C
 D
 E
 F
 G
 H
 I
 J
 K
 L
 M
 N
 O
 P
 Q
 R
 S
T
U
 V
 W
 X
 Y
 Z
 Æ
 Ø
 Å

\unitlength, 156, 320
 \unitlength (længde), 365
 unsrt (BIB_TE_X stil), 201, 209
 unsrnat (BIB_TE_X stil), 212
 \Until, 285
 \Uparrow (↑), 55, 66
 \uparrow (↑), 54, 55, 66
 \Updownarrow (↕), 55, 66
 \updownarrow (hegn), 66
 \upshape, 32
 \Upsilon (Υ), 52, 231
 \upsilon (υ), 52
 \url, 9, 42, 43, 195, 206, 207, 286, 413, 414
 url (BIB_TE_X felt), 205, 207
 url (pakke), 9, 42, 195, 207, 286, 401, 414, 421
 \us (siunitx enhed), 267
 \usebox, 374
 \usepackage, 6, 11–13, 19, 25, 27, 33, 36, 40, 118–120, 125–127, 132, 133, 142, 193, 228, 337, 342–344, 359, 388
 \useshortands, 27, 228
 ushort (pakke), 65
 utf8 (encoding), 13–15, 390
 Utopia, 233

V

\v (accent háček), 30
 valgfri argumenter, 10
 \value, 366
 \varepsilon, 51, 52
 varioref (pakke), 11, 142, 286, 298, 299, 403, 414
 \pageref, 298
 \ref, 298
 \refrange, 298
 \varkappa (κ), 52
 \varnothing (∅), 53
 \varphi (φ), 52
 \varpi (ω), 52
 \varrho (ρ), 52
 \varsigma (ς), 52
 \vartheta (θ), 52
 varwidth (environment), 171, 372
 varwidth (pakke), 372
 \vbox, 132
 \vcenter, 111
 \Vdash (⊨), 53
 \vdash (⊢), 53
 \vdots (⋮), 62
 \vec, 63
 \vee (∨), 53
 \veebar (⊕), 53
 \verb, 225, 249, 403
 \verb*, 249

verbatim (environment), 249–252, 273
 verbatim (pakke), 250
 verbatim tekst, 249
 verbatim* (environment), 249
 verbatimcopy (pakke), 414
 \verbatiminput, 250
 \Vert (hegn), 65, 66
 \vert (hegn), 65, 66
 \vfill, 320, 369, 371
 viewport (graphicx), 139
 Vmatrix (environment), 71
 vmatrix (environment), 71
 \volt (siunitx enhed), 266
 volume (BIB_TE_X felt), 205, 208
 \pageref (varioref), 298
 \phantom, 99
 \href, 403
 \ref (varioref), 298
 \refrange (varioref), 298
 Vsmallmatrix (environment), 72
 vsmallmatrix (environment), 72
 \vspace, 369, 370
 \vspace*, 147, 321, 369

W

\watt (siunitx enhed), 266
 \wd, 375
 \weber (siunitx enhed), 266
 \wedge (∧), 53
 \While, 284, 285
 \whileboolexpr (etoolbox), 368
 \widefbox, 106
 \widehat, 63
 \widetilde, 63
 \widowpenalty, 334
 width (graphicx), 137–139, 147, 148, 156
 \width (længde), 373
 \widthof, 364
 WinEdt, 391
 wrapfig (pakke), 155

X

X (søjle specifier), 182, 183
 xcolor (pakke), 130, 193, 245, 246, 270
 Xdvi, 397
 Xdvi (fremviser), 397
 xdvi (program), 391
 Xe_LA_TE_X, 5
 \Xi (Ξ), 52
 \xi (ξ), 52, 231
 xifthen (pakke), 367
 xindy (program), 254–256
 xkeyval (pakke), 359, 422
 \xleftarrow, 54
 xparse (pakke), 130, 353, 357, 359, 414
 Xpdf, 396

 A
 B
 C
 D
 E
 F
 G
 H
 I
 J
 K
 L
 M
 N
 O
 P
 Q
 R
 S
 T
 U
 V
 W
 X
 Y
 Z
 Æ
 Ø
 Å

xpdf (program), 160
 xr (pakke), 299, 300
 xr-hyper (pakke), 300
 \xrightarrow, 54, 55, 106
 \xspace, 355, 414
 xspace (pakke), 355, 414
 xy (environment), 111
 xy (pakke), 90, 107–112, 428
 \xybox, 111
 \xymatrix (Xy-pic), 90, 107–112
 Xy-pic
 \ar, 90, 107–109
 labels, 109
 pilens udformning, 109
 placering, 108
 \CompileMatrices, 110
 flytte placering af pil, 109
 justere placering af labels, 110
 krummende pile, 109
 labels til pile, 109
 pilehoveder i anden font, 110
 prækompilering, 110
 \xymatrix, 90, 107

Y

Yap, 397
 yap (program), 391
 year (BIB_TE_X felt), 205, 206, 208
 \yocto (siunitx prefix), 267
 \yotta (siunitx prefix), 267

Z

\zepto (siunitx prefix), 267
 \zeta (ζ), 52, 231
 \zetaeta (siunitx prefix), 267
 zorn (environment), 118

Æ

æ

i matematik, 59

ændringer

hvor gælder de?, 356

Ø

ø

i matematik, 59

Å

å

i matematik, 59

\AA (Å), 29

\aa (å), 29

#

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

Æ

Ø

Å

RETTELSER

Rettelser

Dødelige: fixme example	xviii
Note: dette bør opdateres pga. TW default til utf8, og andre editorer som ikke kan utf8	12
Note: fejl: manglende luft	37
Dødelige: De følgende lister af symboler skal gennemgås for at se om der er flere relevante symboler som skal med	51
Note: skrives bedre, når der bliver tid	61
Note: husk lige at tjekke alle referencer til disse	102
Note: skal testes	160
Dødelige: tjek lige dette	185
Dødelige: skal skrives om via threeparttablex	189
Note: husk at få dette med	196
Note: det vil være naturligt at få blandet babelbib ind i det også	207
Note: mathpazo har desuden nogle options man måske kunne bruge til noget	233
Note: bør skrives om, begrundelsen er ikke helt korrekt	237
Note: det kan være en ide at nævne noget vedr. effekter med layers, fik en mail om det for lang tid siden	274
Note: dette skal gennemgås og drøftes med granfeldt og blæsild	278
Note: Her er sikkert mange andre ting vi kan tilføje, ideer?	291
Note: Der bør skrives noget vedr. cleveref, fancyref og venner	299
Note: tjek lige hvor det er man anbefales at placere en dedikation.	303
Note: denne er jeg ikke helt sikker på er et krav	303
Note: side henvisninger til memman skal tjekkes igennem	305
Dødelige: B5 beskrivelse hvor?	307
Note: skal nok placeres et andet sted	321
Dødelige: tilføj mere her, bla. med forsider, samt addtotoc, samt den ikke nummere-rede opgave	324
Note: ovenstående skal testes	325
Dødelige: skal skrives	326
Note: det vil sikkert også være en ide at nævne \sloppy	333
Note: det ville være smart her at have en funktionalitet hvor man kunne vise en højre og en venstre side samtidigt	342
Note: huh???	399
Note: så vidt jeg ved loader vi faktisk slet ikke caption pakken direkte, men gennem subfig	412