

Den nye $\text{\LaTeX}$ for forfattere
En introduktion til $\text{\LaTeX}$ og IMFUFA- $\text{\LaTeX}$

Jørgen Larsen

Juni 2005

TEKSTER fra

IMFUFA

ROSKILDE UNIVERSITETSCENTER
INSTITUT FOR STUDIET AF MATEMATIK OG FYSIK SAMT DERES
FUNKTIONER I UNDERVISNING, FORSKNING OG ANVENDELSER

IMFUFA
Roskilde Universitetscenter
Postboks 260
DK-4000 Roskilde

t	+45 46 74 22 63
f	+45 46 74 30 20
m	imfufa@ruc.dk
w	imfufa.ruc.dk

Jørgen Larsen: Den nye $\LaTeX$ for forfattere. En introduktion til $\LaTeX$ og IMFUFA- $\LaTeX$.

IMFUFA tekst nr. 445/2005

124 sider

issn 0106-6242

Erstatter IMFUFA tekst nr. 366/1999

$\LaTeX$ er et edb-værktøj der er velegnet for brugere der måske ikke besidder den store indsigt i typografiens mysterier, men alligevel ønsker at fremstille pæne, velafbalancerede tekster hvor layoutet medvirker til formidlingen af indholdet. Nærværende skrift giver en kort introduktion til $\LaTeX$ og den overbygning hertil, IMFUFA- $\LaTeX$, som er udviklet ved IMFUFA. Fremstillingen er blevet til på baggrund af forfatterens mangeårige virke som IMFUFAs $\LaTeX$ -vedligeholder og -vejleder, og den indeholder svar på mange af de utallige $\LaTeX$ -spørgsmål han har fået stillet i tidens løb, og også på nogle af de spørgsmål der burde være stillet. Værket henvender sig især til personer der har brug for at fremstille en almindelig tekst skrevet med det latinske alfabet og muligvis med en større eller mindre portion matematiske formler. Derimod står der ikke noget om hvordan man skriver kemiske formler, noder, poesi, tekster med flere forskellige alfabeter osv. osv. Der står heller ikke noget om hvordan man fremstiller billeder eller grafiske fremstillinger på $\LaTeX$ -egnede måder – det ville kræve et særskilt hæfte (for ikke at sige bog).

Den nye $\LaTeX$ for forfattere er en stærkt revideret udgave af *$\LaTeX$ for forfattere* (IMFUFA-tekst nr. 366), blandt andet er matematikafsnittet fuldstændig ændret, og den øvrige tekst er søgt bragt i overensstemmelse med vore dages udbud og funktionalitet af de forskellige dokumentklasser og pakker.

IMFUFA- $\LaTeX$ er et sæt dokumentklasser der er udviklet som supplement til Standard $\LaTeX$, oprindeligt med det formål at gøre livet lidt lettere for danskskrivende $\LaTeX$ -brugere. IMFUFA- $\LaTeX$ kan findes på nettet på <http://dirac.ruc.dk/imfufalatem/>

Indhold

1 Fortale	5
1.1 Vejledning til læseren	6
2 Enkle tekster uden matematik	9
2.1 De første skridt	9
2.2 Enkle L ^A T _E X-kommandoer	10
2.3 Specialtegn	12
2.4 Særlige bogstaver, accenter og andre diakritiske tegn	13
2.5 Punkter og underpunkter	13
2.6 Kapitler og afsnit	15
2.7 Henvisninger, kort	16
2.8 Indholdsfortegnelsen	16
2.9 Dokumentklasserne	17
3 For let øvede	21
3.1 Opstilling i skemaer og tabeller	21
3.2 Figurer og tabeller	26
3.3 Sidebemærkninger	30
3.4 Citater og forskellige sprog	31
3.5 Overspring tekst	34
3.6 Sdens udformning og opbygning	35
4 Ting i kasser	37
4.1 Kasser og rammer	37
4.2 Rotation og skalering	39
4.3 En side på siden: minipage-konstruktioner	40
4.4 Eksterne billeder; graphicx-pakken	41
4.5 Grafik med picture	44
5 Æstetiske betragtninger	49
5.1 Valg af skriftgrad	49
5.2 Valg af skriftsnit	49
5.3 Læselighed	54
5.4 Orddeling	55
5.5 Streger og mellemrum	56

6	Matematik i $\LaTeX$	59
6.1	Simple formler	59
6.2	Grundprincipperne	61
6.3	Tekst i formler	67
6.4	Parenteser og prikker	68
6.5	Matricer og arrays	70
6.6	Opstillinger og nummererede formler	72
6.7	Sætning og bevis	74
6.8	Fed matematik; særlige skrifttyper	76
6.9	Skematiske oversigter	78
7	Henvisninger på kryds og tværs	87
7.1	Henvisninger	87
7.2	Stikord og stikordsregister	89
7.3	Referencer og litteraturhenvisninger	91
7.4	Hyperlinks med hyperref-pakken	96
8	Videregående emner	99
8.1	Om at definere $\LaTeX$ -kommandoer	99
8.2	De indbyggede tekstkonstanter	101
8.3	Operatornavne i matematik	101
8.4	Længder og tællere	102
8.5	Verbatim tekst	104
8.6	»Hvis De vil vide mere ...«	105
A	Et eksempel	107
A.1	Primærfilen	107
A.2	Forsiden	108
A.3	En primærfil til Standard $\LaTeX$	109
	Referencer	111
	Stikord	113

1 Fortale

L^AT_EX er et hjælpemiddel der kan benyttes i den proces hvor en forfatters produkt, en tekst, overgår fra at befinde sig inde i forfatterens hoved til at befinde sig på papir i trykt tilstand. Inden vi præsenterer L^AT_EX, kan det derfor være nyttigt at gennemgå denne omformningsproces lidt nøjere, især som den måske foregik for halvandet hundrede år siden. Man skal tænke sig at der medvirker forskellige aktører (nogle af dem kan meget vel være den samme person):

- *Forfatteren* har to funktioner:
 - Forfatteren bestemmer først og fremmest tekstens *indhold*, og derudover også dens *opbygning*, dvs. opdelingen i sætninger, afsnit, kapitler osv. – Dette krævende kreative arbejde interesserer os ikke her.
 - Forfatteren nedskriver sin tekst som *manuskript*.
Ordet manuskript betyder direkte oversat *noget der er skrevet i hånden*, og vores forfatter for halvandet hundrede år siden skrev netop i hånden, med blyant eller med pen og blæk. Manuskriptet har ofte set forfærdeligt ud, fyldt af rettelser og tilføjelse, og med sære tegn og markeringer der skulle vise at et ord skulle være fremhævet, eller at der skulle begynde et nyt afsnit eller kapitel eller lignende.
- *Designeren* eller *tilrettelæggeren* (for halvandet hundrede år siden var det bogtrykkeren) fastsætter de overordnede retningslinjer for hvordan den trykte tekst skal se ud. Der er tale om afgørelser om blandt andet papirformatet, marginernes størrelse, skriftsnittet (skal man bruge antikva eller måske en blokskrift), skriftgraden (dvs. skriftens størrelse), overskrifternes udformning, sidenumrenes placering, udformningen af opstillinger med punkter og underpunkter, osv.
- *Sætteren* sætter forfatterens tekst efter designerens anvisninger.
Sætteren sørger for at en række typografiske (og æstetiske) retningslinjer bliver overholdt, for eksempel skal bogstaverne i et ord have den rette indbyrdes afstand, mellemrummene mellem ordene skal have den rigtige størrelse, der skal være lige venstre- og højremargin, ord skal deles rigtigt, plus adskilligt andet. Derudover gælder der en række særlige regler for matematiske formler.
Det at sætteren *sætter* teksten, skulle dengang forstås helt bogstaveligt: han tog blytyperne en ad gangen og placerede dem, *satte* dem, i en stor ramme der skulle komme til at svare til en teksts side. (Typerne opbevarede i en slags lille reol, en sættekasse, hvor hvert bogstav havde sit eget rum, de »store« bogstaver i den øverste del af kassen og de »små« bogstaver i den nederste del; deraf kommer de engelske betegnelser 'uppercase' og 'lowercase' for »store« og »små« bogstaver.)

- *Trykkeren* fremstiller papiraftryk af den sats som sætterne har lavet.
- *Korrekturlæseren* læser et prøvetryk igennem for at finde trykfejl. Forfatteren læser også prøvetrykket igennem, men det er mere for at finde indholdsfejl.

I vore dage er en del af disse funktioner overtaget af computeren. I L^AT_EX-verdenen er det sådan at *sætteren*s opgave udføres af computeren. Forfatteren og designeren er stadig mennesker, men nu kan de ikke længere gå hen og sludre med sætteren og fortælle hvad de der mærkelige kruseduller på side 102 egentlig betyder; i stedet må de betjene sig af et standardiseret sprog til typografiske markeringer; L^AT_EX er et sådant sprog. Når forfatteren arbejder på sit manuskript, som nu skrives på computer, skal han eller hun altså stadig markere hvis der skal ske noget særligt, f.eks. at et ord skal skrives med fed skrift, blot gøres det nu ikke mere med en eller anden slags understregning, men med en bestemt L^AT_EX-kommando. På samme måde giver designeren sine instruktioner til »sætteren« i form af L^AT_EX-kommandoer.

Resultatet af »sætteren«s arbejde er en fil i pdf-format eller eventuelt i det såkaldte dvi-format. Man har derfor brug for endnu et computerprogram, en *previewer*, der kan vise det færdige arbejde skærmen, og som kan printe det ud på en printer.

Desforuden skal forfatteren jo fremstille en fil med manuskriptet, og til det brug har han/hun brug for en simpel tekst-editor.

Efterhånden er der ved at være så mange programmer i spil at man også har brug for et program der kan hjælpe brugeren med at holde styr på de andre programmer. Et sådant program, hvis grad af sofistikerede er meget afhængig af hvilket operativsystem computeren benytter sig af, er typisk bygget op omkring en hensigtsmæssig tekst-editor hvor man ved hjælp af simple tastekombinationer eller fra menuer kan aktivere alle de øvrige programmer.

1.1 Vejledning til læseren

Metavejledning: Dette afsnit »Vejledning til læseren« har oplysninger både til nybegynderen og til den viderekomne, men man skal selv sortere i indholdet: nybegynderen opfordres til at gå let hen over det der forekommer uforståeligt (men så vende tilbage til det ved en senere lejlighed), og den garvede bruger kan lynhurtigt læse hen over det hun véd i forvejen. Der findes sidst i hæftet et stikordsregister som måske kan være til nogen hjælp.

Som antydnet ovenfor er det at arbejde med L^AT_EX en flertrinsproces: man skal skrive en tekstfil, programmet **pdflatex** (eller **latex**) skal behandle denne tekstfil og fremstille en pdf-fil (eller en dvi-fil), og denne skal *view*'es eller printes ud. Dette hæfte handler om *hvad* der skal stå i tekstfilen for at det færdige resultat bliver pænt.

Derimod handler hæftet ikke om *hvordan* man konkret fremstiller tekstfilen, og hvordan man konkret aktiverer **pdflatex**-programmet osv. (Der findes forskellige L^AT_EX-editorer og udviklingsmiljøer, hver med deres fordele og ulemper. Det konkrete valg må afhænge af hvilke slags krav man stiller til sin editor, og det er derfor i høj grad et personligt valg.)

De tekstfiler man skriver, bør normalt navngives sådan at de ender på **.tex**, og derfor kaldes de i daglig tale for **tex**-filer.

Typografiske konventioner

Mange steder i dette hæfte vises ting i tospaltede eksempelkasser hvor der i højre spalte står det man skal skrive i `tex`-filen, og i venstre spalte det resultat der kommer ud af det. I eksempelkasserne såvel som i den øvrige tekst gælder at den tekst som brugeren skal skrive i sin `tex`-fil, vises med **skrivemaskineskrift**. Visse ting skrives med **blokskrift** (‘sans serif’), nemlig navne på pakker (se nedenfor), dokumentklasser (omtales i afsnit 2.9 side 17) og (computer)programmer. Ved omtalen af kommandoer o.lgn. bliver navnet på det omtalte objekt ofte skrevet som sidenote i ydermarginen.

Sidenote

Supplerende og uddybende bemærkninger skrives med lille skrift. De henvender sig fortrinsvis til den viderekomne L^AT_EX-bruger.

- Hvis et afsnit indeholder noget særlig vigtigt, markeres det med et fedt udråbstegn i venstre margin.

Kommandoer og deres argumenter

I `tex`-filen skriver man almindelig tekst samt et større eller mindre antal L^AT_EX-kommandoer. En L^AT_EX-kommando kendes på at den begynder med tegnet `\` (‘backslash’). Et eksempel på en kommando er `\today`, hvis virkning er at den indsætter den aktuelle dato i teksten på det sted hvor kommandoen står. – Der er rigtig mange L^AT_EX-kommandoer, og nærværende tekst omtaler kun en mindre del af dem. Derudover kan brugeren definere sine egne kommandoer (se afsnit 8.1 side 99).

Nogle kommandoer har et eller flere *argumenter*. Et argument til en kommando er »noget« som kommandoen skal operere på; argumenter skrives i `{ }` efter kommandonavnet. – Eksempel: Kommandoen `\textit` skriver sit argument i kursiv, så f.eks. resultatet af `\textit{gås}` er at ordet *gås* skrives i kursiv.

Når man skal beskrive en kommando (f.eks. angive dens syntaks), er det ofte nyttigt at kunne referere til dens argument(er); vi skriver de formelle argumenter i spidse parenteser og i kursiv, sådan her: *⟨arg⟩*. – Eksempel: Hvis der står at der findes kommandoen `\author{⟨navn⟩}`, så er meningen at kommandoen `\author` har ét argument, og at vi bruger symbolet *⟨navn⟩* som pladsholder for dette argument (som nemlig typisk er et navn); når man benytter `\author`, skal man i stedet for *⟨navn⟩* skrive et navn; en mulighed er `\author{Hercule Poirot}`.

Generelt kan en kommando have 0 eller flere valgfrie argumenter og 0 eller flere obligatoriske argumenter; et valgfrit argument skrives i `[ ]`. Når man vil angive mere end et argument til en kommando, skal hvert argument stå i sin egen parentes (`[ ]` eller `{ }`) og argumenterne skal komme i den rigtige rækkefølge (argumentnavnene er kun til glæde for læsere af L^AT_EX-manualer, latex-programmet kender dem ikke).

Dokumentets opbygning

Et L^AT_EX-dokument er altid opbygget på en bestemt måde:

1. Først kommer *dokumentklasseerklæringen*. Dokumentklassen bestemmer det layout som teksten skal udformes efter, og den specificeres med en *dokumentklasseerklæring* af formen `\documentclass{⟨klasse⟩}` eller `\documentclass[⟨valg⟩]{⟨klasse⟩}`

`\documentclass`

hvor $\langle valg \rangle$ er en eller flere valgmuligheder ('options'), adskilt af komma.

2. *Præamblen* er de linjer der kommer efter dokumentklasseerklæringen men inden $\begin{document}$. I præamblen foregår indlæsning af *pakker*, og hvis brugeren ønsker at definere sine egne kommandoer, sker det normalt også i præamblen (læs om hvordan på side 99).

$\backslash usepackage$

En *pakke* er i L^AT_EX-sammenhæng en samling definitioner og kommandoer der ændrer eller øger L^AT_EXs funktionalitet. Pakker indlæses med $\backslash usepackage\{\langle pakke \rangle\}$ eller $\backslash usepackage[\langle valg \rangle]\{\langle pakke \rangle\}$, og det kan kun ske i præamblen.

Eksempel: hvis man ønsker at benytte pakken *bm*, skal man i præamblen skrive $\backslash usepackage\{bm\}$.

Dokumentklasserne fra IMFUFA-L^AT_EX indlæser automatisk et antal pakker der blandt andet sørger for at vælge dansk orddeling (se evt. side 32).

3. Selve teksten indledes med $\begin{document}$ og afsluttes med $\end{document}$. Den første outputproducerende kommando er ofte $\maketitle$.

4. Hvis man har at gøre med en lang tekst, kan det være upraktisk at have hele teksten stående i én stor fil. Man står sig som regel ved at dele teksten op i mindre filer, f.eks. svarende til kapitler eller afsnit, og så instruere L^AT_EX om at læse disse i den rigtige rækkefølge. Kommandoen til indlæsning af filer er $\backslash input\{\langle filnavn \rangle\}$; eksempel: $\backslash input\{kapitel2.tex\}$.

$\backslash input$

Man kan med fordel organisere arbejdet på den måde at man har en overordnet fil, den såkaldte *primærfil*, som indeholder dokumentklasseerklæringen og præamblen, og som indlæser de underordnede filer med de enkelte kapitler eller afsnit.

Undervejs i tilblivelsesprocessen kan man udkommentere en større eller mindre del af inputkommandoerne (ved at sætte kommentartegnet % foran), sådan at man kan nøjes med at behandle udvalgte filer.

- Forskellige styresystemer kan have forskellige konventioner for filnavne. Nogle systemer (f.eks. Windows) bruger 'backslash' (\) som skille tegn i filnavnene, andre (f.eks. Unix/Linux) bruger 'forwardslash' (/). Inde i *tex*-filen skal man imidlertid *altid* bruge 'forwardslash'. Windows-brugere kan derfor komme ud for at skulle skrive f.eks. $\backslash input\{c:/modul3/projekt/kapitel2.tex\}$.

- Til illustration og inspiration kan man på side 107 se primærfilen til nærværende dokument.

2 Enkle tekster uden matematik

Dette kapitel er en blid indøvelse i de grundlæggende L^AT_EX-tankegange i forbindelse med fremstilling af almindelige tekster uden matematiske formler. Der kommer mange eksempler, og de er for det meste skrevet i de særlige tospaltede eksempelkasser hvor der til højre står hvad man skal skrive (input-teksten), og til venstre hvad der kommer ud af det.

2.1 De første skridt

Vi begynder med at vise hvad der skal skrives for at fremstille den meget enkle tekst »Hej med dig!«:

Hej med dig!	<pre>\documentclass{imfnote} \begin{document} Hej med dig! \end{document}</pre>
--------------	---

EKS. 2.1

Input-filen til en L^AT_EX-tekst begynder altid med en dokumentklasseerklæring der i det simpleste tilfælde er af formen `\documentclass{<klasse>}`; dokumentklassen fastlægger det layout der skal benyttes til den type tryksag man ønsker at fremstille. Efter dokumentklasseerklæringen følger selve teksten, der altid skal omgives af `\begin{document}` og `\end{document}`. I IMFUFA-L^AT_EX findes blandt andet dokumentklasserne `imfnote`, `imfreport`, `imfbook` og `imfletter`; der kommer mere om dokumentklasser på side 17.

`\documentclass`

Trods sin enkelhed illustrerer eksemplet et par væsentlige ting. Der optræder særlige »ord« der begynder med en *backslash* (dvs. tegnet `\`); det er *kommandoer*, dvs. instruktioner til L^AT_EX. En backslash betyder altid at det der kommer umiddelbart efter, skal opfattes på en særlig måde, nemlig som en *kommando*, dvs. en instruktion til L^AT_EX. Allerede i første linje er der en kommando, nemlig `\documentclass`. Kommandoer kan have obligatoriske og/eller valgfrie *argumenter*; i eksemplet har kommandoen `\documentclass` argumentet `imfnote`; obligatoriske argumenter skrives i `{ }`, hvorimod valgfrie argumenter skrives i `[ ]`.

Eksemplet indeholder også parret `\begin{document} \end{document}`. L^AT_EX-dokumenter indeholder normalt et hav af sådanne `\begin{<noget>} \end{<noget>}`-konstruktioner; deres opgave er at afgrænse et område af input-teksten hvor der skal gælde *<noget>*; i særdeleshed afgrænser `\begin{document} \end{document}` det område hvor L^AT_EX-reglerne gælder. For `\begin{...} \end{...}`-konstruktioner gælder den simple regel at ethvert `\begin{...}` skal have et modsvarende `\end{...}` (og omvendt).

I de følgende eksempler vil vi ikke gengive hele input-teksten, men kun den del der i den givne sammenhæng er interessant.

Vi udvider det første eksempel lidt:

Hej med dig! Du er vist en <i>værre</i> en, hvad er du ude på?	Hej med dig! Du er vist en <code>\textit{være}</code> en, hvad er du ude på?
--	--

EKS. 2.2

`\textit` Her er brugt kommandoen `\textit{⟨tekst⟩}` der sætter sit argument i kursiv.

Man kan se at det ikke har nogen betydning for slutresultatet om der er et eller flere mellemrum mellem ordene i input-teksten. Det betyder heller ikke noget at der er et enkelt linjeskift, men se så her:

Hej med dig! Du er vist en <i>værre</i> en, hvad er du ude på?	Hej med dig! Du er vist en <code>\textit{være}</code> en, hvad er du ude på?
---	--

EKS. 2.3

`\par` En blank linje betyder *begynd et nyt afsnit*, hvilket i den resulterende tekst ses som linjeskift plus indrykning. I stedet for en blank linje kan man skrive `\par`:

Hej med dig! Du er vist en <i>værre</i> en, hvad er du ude på?	Hej med dig!\par Du er vist en <code>\textit{være}</code> en, hvad er du ude på?
---	---

EKS. 2.4

`\bpar` Hvis man i teksten vil gå over til noget lidt andet og gerne vil markere det anderledes end med det almindelige »nyt afsnit«, kan man lave et *mellemslag* med kommandoen `\bpar` (»big par«):

Hej med dig! Du er vist en <i>værre</i> en, hvad er du ude på?	Hej med dig!\bpar Du er vist en <code>\textit{være}</code> en, hvad er du ude på?
---	--

EKS. 2.5

Det er uden betydning om der er mellemrum og/eller blanke linjer før og efter `\bpar`.

Det er en udbredt misforståelse (som vist desværre bakkes op at visse udbredte tekstbehandlingssystemer) at »nyt afsnit« skal markeres ved at begynde på ny linje idet man laver en ekstra stor afstand til den nye linje (som så ikke indrykkes). Det er svært at finde eksempler på seriøse bøger, aviser eller tidsskrifter der benytter denne praksis, og det gør L^AT_EX heller ikke.

`\imfufapar` Hvis man imidlertid alligevel en sjælden gang skulle få brug for at udarbejde en tekst med denne type »nyt afsnit«, kan man benytte kommandoen `\imfufapar` som omdefinerer kommandoerne »blank linje« og `\par` til at betyde »begynd på ny linje uden indrykning men med ekstra linjeafstand«. Kommandoen `\bpar` justeres tilsvarende. – Kommandoerne `\bpar` og `\imfufapar` findes kun i IMFUFA-L^AT_EX.

2.2 Enkle L^AT_EX-kommandoer

L^AT_EX-kommandoer har næsten altid navne af formen en backslash (\) efterfulgt af et eller flere amerikanske bogstaver (A ... Z a ... z), og de kan have nul eller flere

argumenter der skrives i { }. Vi har allerede set kommandoen `\textit` der skriver sit argument med kursiv. Et andet eksempel er kommandoen `\enquote{<tekst>}` der sætter anførselstegn om `<tekst>`. Eksempel:

`\enquote`

EKS. 2.6

En »køn« historie! Hun sagde: »En
'køn' historie!«

En `\enquote{køn}` historie!
Hun sagde: `\enquote{En
\enquote{køn} historie!}`

Som det ses sørger `\enquote` for at markere citater anderledes når de står inden i et andet citat; hvis man bruger `\enquote*`, får man altid den inderste udgave. Udseendet af anførselstegnene afhænger af det valgte sprog, se side 33f hvor der står mere om kommandoer til citater.

Kommandoen `\enquote` er til rådighed i IMFUFA-L^AT_EX-dokumentklasserne fordi de automatisk indlæser pakken `csquotes` som definerer den, se Lehman (2005)). Hvis man benytter andre dokumentklasser, må man selv sørge for at inkludere pakken.

Der er to officielle måder at skrive anførselstegn på. Den ene er den der ses i nærværende tekst og som automatisk vælges af IMFUFA-L^AT_EX-klasserne. Den anden måde fås ved at indsætte følgende kommando i præmablen: `\DeclareQuoteAlias[quotes]{danish}{danish}`. Hvis man i stedet ønsker anførselstegn efter den engelske model, skal man skrive `\DeclareQuoteAlias{english}{danish}`.

Endnu et eksempel på en kommando med ét argument er `\footnote{<tekst>}` der skriver teksten som en fodnote:

`\footnote`

EKS. 2.7

Nogle forfattere er glade for fodnoter,¹ men mange fodnoter gør teksten svær at læse.

¹ En god ting kan også overdrives.

Nogle forfattere er glade for fodnoter, `\footnote{En god ting kan også overdrives.}` men mange fodnoter gør teksten svær at læse.

Som eksempel på en kommando med to argumenter bruger vi en kommando der fremstiller pæne brøker (eng.: nice fractions), nemlig `\nicefrac{<tæller>}{<nævner>}` :

`\nicefrac`

EKS. 2.8

$\frac{1}{2}$ kg saftige ribs og $\frac{1}{4}$ kg fuldmodne hindbær vrides. Heraf fås $3\frac{1}{2}$ dl saft.

`\nicefrac{1}{2}` kg saftige ribs og `\nicefrac{1}{4}` kg fuldmodne hindbær vrides. Heraf fås `3\nicefrac{1}{2}` dl saft.

Man kan sagtens have kommandoer med nul argumenter, for eksempel bliver L^AT_EX-logoet fremstillet af kommandoen `\LaTeX`:

EKS. 2.9

L^AT_EXer nemt.

`\LaTeX` er nemt.

Dette eksempel viser at L^AT_EX måske alligevel ikke er helt så nemt, for hvorfor er der nu ikke noget mellemrum mellem »L^AT_EX« og »er«? Det kommer sig af at når L^AT_EX-fortolkeren støder på tegnet `\`, skifter den fra en »læs ord«-tilstand til en »læs kommando«-tilstand, og i »læs kommando«-tilstanden er mellemrumstegnets betydning den at det kan bruges til at afslutte et kommandonavn, men derudover giver det ikke noget synligt resultat. Og hvad gør man så? Ja, her vises tre forskellige løsningsmuligheder; til lejligheden er mellemrumstegnet i input-teksten vist som `\`:

$\LaTeX$ er nemt.	<code>{\LaTeX}_er_nemt.</code>
$\LaTeX$ er nemt.	<code>\LaTeX\{\}_er_nemt.</code>
$\LaTeX$ er nemt.	<code>\LaTeX_er_nemt.</code>

EKS. 2.10

I den første løsning er der sat $\TeX$ -parenteser (`{ }`) omkring `\LaTeX` for at angive at denne kommando skal behandles for sig selv; i den anden løsning bruges det tomme tegn `{ }` til at markere afslutningen på kommandonavnet; i den tredje løsning benyttes kommandoen »backslash-mellemrum« (`\_`) der giver et almindeligt ord-mellemrum.

2.3 Specialtegn

Visse tegn, nemlig `\ % { } $ " & # _ ^ ~`, har en helt speciel betydning:

`\` en *backslash* bruges til at markere $\LaTeX$ -kommandoer.

Teksttegnet `\` frembringes med kommandoen `\textbackslash`.

`%` *procenttegn* er et kommentar-tegn. Tegnet selv og hele resten af linjen ignoreres.

Teksttegnet `%` frembringes med kommandoen `\%`.

`{ }` *krøllede parenteser* bruges om argumenter til kommandoer, og de bruges også som parenteser i forhold til $\LaTeX$ -kommandosproget, dvs. de kan afgrænse rækkevidden af definitioner.

Teksttegnene `{ }` frembringes med kommandoerne `\{ }` og `\}` eller alternativt med `\textbraceleft` og `\textbraceright`.

`$` *dollartegn* skifter frem og tilbage mellem $\LaTeX$ s tekst- og matematik-tilstand.

Teksttegnet `$` frembringes med kommandoen `\$`.

`"` *dobbelt anførselstegn* er et specialtegn i *babel*-pakken og må *ikke* bruges som anførselstegn; dels er det grimt, dels virker det ikke altid som man skulle forvente (se f.eks. eks. 5.17 på side 56).

`&` *og-tegn* bruges som tabulatortegn i tabeller og arrays.

Teksttegnet `&` frembringes med kommandoen `\&`.

`#` *nummertegn* bruges i forbindelse med definition af kommandoer.

Teksttegnet `#` frembringes med kommandoen `\#`.

`_` *understreg-tegn* bruges i matematiksammenhæng til »fodtegn« (som i x_n).

Teksttegnet `_` frembringes med kommandoen `\_`.

`^` *circumflex-tegn* bruges i matematiksammenhæng til »toptegn« (som i x^n).

Teksttegnet `^` frembringes med kommandoen `\textasciicircum`. Hvis man vil sætte en circumflex-accent over et bogstav, gøres det med kommandoen `\^`, se tabel 2.2.

`~` *tilde* bruges til at angive et fast mellemrum, dvs. et ordmellemrum hvor der ikke må forekomme linjeskift (se eks. 5.20 side 57).

Teksttegnet `~` frembringes med kommandoen `\textasciitilde`. Hvis man vil sætte en tilde over et bogstav, gøres det med kommandoen `\~`, se tabel 2.2.

Tabel 2.1 Særlige bogstaver og deres L^AT_EX-kommandoer.

æ	<code>\ae</code>	Æ	<code>\AE</code>
ø	<code>\o</code>	Ø	<code>\O</code>
å	<code>\aa</code>	Å	<code>\AA</code>
ß	<code>\ss</code>	SS	<code>\SS</code>
œ	<code>\oe</code>	Œ	<code>\OE</code>
ł	<code>\l</code>	Ł	<code>\L</code>
ı	<code>\i</code>	İ	<code>\j</code>
ŋ	<code>\ng</code>	Ŋ	<code>\NG</code>
ð	<code>\dh</code>	Ð	<code>\DH</code>
þ	<code>\th</code>	Þ	<code>\TH</code>

Tabel 2.2 Accenter og andre diakritiske tegn, anvendt på bogstavet u.

ù	<code>\'u</code>	gravis	ú	<code>\'u</code>	akut
û	<code>\^u</code>	circumfleks	ũ	<code>\~u</code>	tilde
ü	<code>\"u</code>	Umlaut	ů	<code>\H u</code>	
ů	<code>\u u</code>	brevis	ǔ	<code>\v u</code>	háček
ù	<code>\. u</code>		ȳ	<code>\d u</code>	
ȳ	<code>\c u</code>	cedille	ȳ	<code>\k u</code>	ogonek
ū	<code>\= u</code>	macron	ḁ	<code>\b u</code>	
ũ	<code>\r u</code>	bolle			

2.4 Særlige bogstaver, accenter og andre diakritiske tegn

T_EX og L^AT_EX blev oprindeligt udviklet i USA, og (blandt andet) derfor anerkendte de første udgaver kun de to gange 26 bogstaver ABCD...XYZ og abcd...xyz som rigtige bogstaver. Sidenhen er L^AT_EX i høj grad også blevet et europæisk projekt, og udmærkede bogstaver som æ ø å ř é ä ß ð þ er kommet ind i varmen.

Hvis man benytter en fornuftig L^AT_EX-installation, kan man som hovedregel gå ud fra at de bogstaver man kan skrive på tastaturet, også vil optræde på den forventede måde i den resulterende tekst. Imidlertid er det jo ikke alle tastaturer der har f.eks. en ð-tast, men så er det muligt at angive bogstaverne ved hjælp af deres L^AT_EX-kommandoer, se tabel 2.1.

Derudover findes en lang række specialtegn, såsom ‰ (`\promille`), € (`\euro`), og °C (`\celsius`).

Der er L^AT_EX-kommandoer til alskens accenter og andre diakritiske tegn; disse vises i tabel 2.2 anvendt på bogstavet u, men de kan anvendes på ethvert bogstav.

2.5 Punkter og underpunkter

En væsentlig grund til at skrive en tekst er at man har en forventning om at der er nogen der vil læse den. Derfor bør man opfatte skriveprocessen som en formidlingsproces, og i den sammenhæng er L^AT_EX et effektivt formidlingsredskab.

Det sker at tekst-dele består af udviklede forklaringer eller ræsonnementer der med fordel kan stilles op i en række punkter hvoraf nogle måske har underpunkter. I sådanne situationer kan man anvende en eller flere af L^AT_EX's tre opremsnings-konstruktioner. Fælles for dem er at de enkelte punkters begyndelse markeres med kommandoen `\item`; der må ikke stå tekst foran det første `\item`.

`itemize` `itemize` producerer en række punkter. Eksempel:

En opremsning: • Et punkt. • Et punkt til. • Endnu et punkt. Færdig!	En opremsning: <pre>\begin{itemize} \item Et punkt. \item Et punkt til. \item Endnu et punkt. \end{itemize}</pre> Færdig!
---	--

EKS. 2.11

Man kan vælge det tegn der skal markere de enkelte punkter ved at skrive det i kantede parenteser efter `\begin{itemize}`, f.eks. `\begin{itemize}[*]`. Man kan også omdefinere det generelt, se eks. 8.8 på side 101.

`enumerate` `enumerate` producerer en række nummererede punkter. Eksempel:

Nummererede punkter: 1. Et punkt. 2. Et punkt til. 3. Endnu et punkt. Færdig!	Nummererede punkter: <pre>\begin{enumerate} \item Et punkt. \item Et punkt til. \item Endnu et punkt. \end{enumerate}</pre> Færdig!
--	--

EKS. 2.12

Bemærk at L^AT_EX selv sørger for nummereringen.

`description` `description` producerer en række navngivne punkter. De enkelte punkter indledes med `\item[⟨tekst⟩]`. Eksempel:

Husmoderens tip: Gæs spises Mortens aften, ihukommende Skt. Martin. Ænder spises til jul, fordi en gås er for stor. Høns spises i asparges. God appetit.	Husmoderens tip: <pre>\begin{description} \item[Gæs] spises Mortens aften, ihukommende Skt. Martin. \item[Ænder] spises til jul, fordi en gås er for stor. \item[Høns] spises i asparges. \end{description}</pre> God appetit.
---	---

EKS. 2.13

Som det ses bliver `⟨tekst⟩`-argumentet til `\item` sat med en fed skrift. Dette kan ændres med kommandoen `\descriptionlabelfont[⟨fontklæring⟩]` (se side 51 med hensyn til hvad argumentet `⟨fontklæring⟩` kan være). Eksempel: `\descriptionlabelfont{\itshape}`. Hvis kommandoen står i præamblen, gælder ændringen globalt. Kommandoen `\descriptionlabelfont` findes kun i IMFUFA-L^AT_EX.

`\descriptionlabelfont`

Tabel 2.3 De forskellige ‘sectioning commands’ og deres plads i afsnits-hierarkiet.

kommando	niveau
<code>\part</code>	-1
<code>\chapter</code>	0
<code>\section</code>	1
<code>\subsection</code>	2
<code>\subsubsection</code>	3
<code>\paragraph</code>	4
<code>\subparagraph</code>	5

I de tre eksempler er hvert punkt (item) af pladshensyn kun på én linje; hvis et punkt strækker sig over flere linjer, bliver disse automatisk rykket ind så det er tydeligt at der er tale om en opstilling i punkter. Det kan der findes mange eksempler på i dette hæfte.

Et punkt kan indeholde vilkårlig tekst (og matematiske formler), specielt kan det indeholde en af de tre nævnte opremsningskonstruktioner; på den måde får man underpunkter osv. $\LaTeX$ sørger selv for at underpunkter markeres og nummereres på en hensigtsmæssig måde.

2.6 Kapitler og afsnit

En tekst er ofte inddelt i afsnit med overskrifter af forskellige grader, f.eks. *kapitler* der kan være inddelt i *hovedafsnit* der igen kan være inddelt i *mellemrubrikker* (underafsnit). I større afhandlinger kan kapitlerne være samlet i *dele*. I $\LaTeX$ markeres sligt ved hjælp af kommandoerne `\part`, `\chapter`, `\section`, `\subsection` osv. (jf. tabel 2.3) der har det pågældende afsnits titel som argument. Eksempler:

```
\chapter{Enkle tekster uden matematik}
\section{Kapitler og afsnit}
```

`\chapter`
`\section` osv.

Alle afsnitskommandoerne har samme syntaks, så vi nøjes med at anføre syntaksen for `\section`. Den er `\section[<alt-overskrift>]{<overskrift>}`, hvor den valgfrie alternative overskrift *<alt-overskrift>* bruges i stedet for *<overskrift>* i indholdsfortegnelsen og i sidens »hoved«.

$\LaTeX$ sørger selv for nummereringen af kapitler, afsnit osv. Den valgte dokumentklasse bestemmer hvordan kapitler, afsnit osv. markeres, eksempelvis overskrifternes udseende, om kapitler altid skal begynde på en højreside, og hvor mange niveauer i afsnitshierarkiet der skal nummereres.

Med kommandoen `\setsecnumdepth{<navn>}` kan man selv bestemme hvor dybt der skal nummereres, f.eks. vil `\setsecnumdepth{chapter}` bevirke at kun kapitler (og dele) nummereres. Bemærk dog at dokumentklassen sætter en grænse for hvor langt ned der kan nummereres; denne grænse kan man ændre ved i præamblen at benytte kommandoen `\maxsecnumdepth{<navn>}`, f.eks. `\maxsecnumdepth{subsection}`.

`\setsecnumdepth`

`\maxsecnumdepth`

Kommandoerne `\setsecnumdepth` og `\maxsecnumdepth` findes kun i *memoir*-klassen og dermed også i *IMFUF*A- $\LaTeX$ -klasserne. Hvis man benytter *Standard* $\LaTeX$, ændrer man nummereringsgrænsen med `\setcounter{secnumdepth}{<niveau>}`, f.eks. `\setcounter{secnumdepth}{2}`.

Afsnitskommandoerne har alle en »stjernet« variant, f.eks. `\section*{overskrift}`. Når man benytter en sådan, bliver det pågældende afsnit aldrig nummereret og kommer aldrig med i indholdsfortegnelsen.

Måske skal teksten have et appendiks eller to til sidst. I så fald benyttes kommandoen `\appendix` én gang; alle efterfølgende `\chapter`-kommandoer fremstiller da appendikser.

2.7 Henvisninger, kort

I saglitteratur har forfatteren ofte brug for (og læseren glæde af) henvisninger fra den fortløbende tekst til figurer, tabeller, afsnit sider, formler, fodnoter osv. Med $\text{\LaTeX}$ er det grundlæggende meget enkelt at lave sådanne henvisninger: på det sted man gerne vil henvise til, placerer man et `\label{mærke}` (ved brug af kommandoen `\label{mærke}`), derefter kan man henvise til det med kommandoen `\ref{mærke}` eller en af varianterne heraf.

Følg anvisningen i eks. 2.15

Følg anvisningen i eks. `\ref{ex:label}`

Eks. 2.14

Læs om henvisninger på side 87.

`\label{ex:label}`
Læs om henvisninger på
`\pref{sec:refoglabel}`.

Eks. 2.15

Læs nærmere på side 87.

2.8 Indholdsfortegnelsen

`\tableofcontents*` Indholdsfortegnelsen fås ved at skrive kommandoen `\tableofcontents*` på det sted i teksten hvor indholdsfortegnelsen ønskes anbragt, og det er jo almindeligvis et sted i begyndelsen.

- Hvis man gerne vil have indholdsfortegnelsen til at begynde på en højreside (som det er almindelig kutyme), skal man skrive `\cleartorecto` lige før `\tableofcontents*`. – *Recto* betyder på højre side af et blad eller håndskrift; tilsvarende betyder *verso* på venstre side.

I Standard $\text{\LaTeX}$ fremstilles indholdsfortegnelsen med `\tableofcontents`. Denne kommando findes også i *memoir* (og dermed i *IMFUFA- $\text{\LaTeX}$* -klasserne), men her fremstiller den en indholdsfortegnelse der indeholder sig selv.

Første gang $\text{\LaTeX}$ sætter teksten, bliver indholdsfortegnelsen tom, anden gang har den det rigtige indhold men ofte forkerte sidetal (fordi hvis den selv fylder mere end en eller to sider, så ændres nummereringen af hele den følgende tekst), fra og med tredje gang er den rigtig, medmindre man selv ændrer i teksten.

`\settocdepth` Kommandoen `\settocdepth{<navn>}` kan bruges til at bestemme hvor mange niveauer der skal med i indholdsfortegnelsen, eksempelvis vil `\settocdepth{chapter}` bevirke at kun kapitler (og dele) kommer med. Dokumentklassen sætter en grænse for hvor mange afsnitsniveauer der kan komme med i indholdsfortegnelsen. Det kan man ændre med kommandoen `\maxtocdepth{<navn>}`, f.eks. `\maxtocdepth{subsubsection}`;

dette skal gøres inden `\tableofcontents*`.

Kommandoerne `\settocdepth` og `\maxtocdepth` findes kun i `memoir`-klassen og dermed også i `IMFUFA-LATEX`-klasserne. Hvis man benytter Standard `LATEX`, ændrer man nummereringsgrænsen med `\setcounter{tocdepth}{\langle niveau \rangle}`, f.eks. `\setcounter{tocdepth}{2}`.

2.9 Dokumentklasserne

Dokumentklassen bestemmer det *layout* som teksten skal fremstilles efter. Standard `LATEX` er udviklet til brug for amerikanere og indeholder blandt andet dokumentklasserne `book`, `report`, `article` og `letter`. `IMFUFA-LATEX` er beregnet for danskere og engelsk-skrivende europæere og indeholder blandt andet dokumentklasserne `imfbook`, `imfreport`, `imfart`, `imfnote` og `imfletter`; disse klasser er overbygninger på `memoir`-klassen (Wilson, 2004). Dokumentklassernes navne antyder hvad de er beregnet til; nærværende publikation er fremstillet med dokumentklassen `imfreport`.

Valgmuligheder (‘options’)

Dokumentklassen specificeres med `\documentclass[\langle valg \rangle]{\langle klasse \rangle}` hvor `\langle klasse \rangle` er f.eks. `imfreport`, og `\langle valg \rangle` er en liste af valgmuligheder. Idéen er at klassen definerer layoutet, og valgmulighederne bruges til eventuelle småjusteringer. Her er nogle af mulighederne (se nærmere i Wilson (2004, kapitel 5) og Larsen (2005)):

- Normalskriftens størrelse vælges ved at angive et af valgene `9pt`, `10pt`, `11pt`, `12pt`, `14pt`, `17pt`. Standard er `10pt` (læs evt. om enheden `pt` på side 102).
Mulighederne `9pt`, `14pt` og `17pt` findes ikke i Standard `LATEX`, kun i `memoir` og `IMFUFA-LATEX`.
9pt 10pt 11pt
12pt 14pt 17pt
- Som standard vælges papirformatet A4, svarende til valgmuligheden `a4paper`. (`IMFUFA-LATEX` har dette format som default, ellers er defaultformatet det engelske ‘letter’ format). Der er en del andre mulige formater, bl.a. A5 (`a5paper`) og B5 (`b5paper`). En helt tredje mulighed er e-book formatet (`ebook`).
Man kan desuden vælge `landscape`.
a4paper

ebook
landscape
- Nogle klasser er indrettet på at slutresultatet skal hæftes sammen i ryggen, og at der skal være tekst på både højre- og venstresiderne. Som følge heraf bliver siderne med hhv. ulige og lige sidetal udformet så de kan bruges som hhv. højre- og venstresider i det færdige opslag. Man kan med valgmuligheden `oneside` fortælle at der ikke skal være forskel på lige og ulige sider (hvorimod `twoside` insisterer på at der skal være forskel).
oneside
twoside
- Almindeligvis begynder kapitler i »tosidede« tekster altid på en højreside. Man kan ændre dokumentklassens valg med en af valgmulighederne `openright` (standard), `openany` og `openleft`.
openright
openany
openleft
- Som nævnt ovenfor bliver linjeafstanden (»skydningen«) øget en smule når man bruger `palatino`-muligheden. Hvis man ikke ønsker Palatino-skriften, men alligevel gerne vil have den ekstra linjeafstand, kan man angive valgmuligheden `pplleading`. (Resultatet bliver da som i denne publikation.)
pplleading
- Mange af funktionaliteterne i `LATEX` bør afhænge af hvilket sprog teksten skrives på.
(sprog)

I en dansksproget publikation er det eksempelvis fornuftigt at betegne indholdsfortegnelsen »Indhold«, hvorimod det i en engelsksproget publikation ville være bedre med »Contents«. Hvis man i IMFUFA- $\text{\LaTeX}$ vil skrive på andre sprog end dansk, skal man angive alle (de engelske navne på) de sprog man har tænkt sig at bruge, plus `danish`, som valgmuligheder til dokumentklassen, og dokumentets hovedsprog skal være det sidste. Herefter går resten af sig selv, takket være `babel`-pakken som IMFUFA- $\text{\LaTeX}$ automatisk indlæser.

I Standard $\text{\LaTeX}$ skal man på samme måde skrive alle ønskede sprog, hovedsproget til sidst, men her behøver man ikke at skrive `danish`, medmindre man ønsker sproget dansk. Desuden skal man selv indlæse `babel`.

- | | |
|-----------------------|--|
| <code>leqno</code> | • Nummererede matematiske formler får almindeligvis skrevet nummeret yderst til højre på linjen. Hvis man i stedet vil have nummeret stående yderst til venstre, skal man angive valgmuligheden <code>leqno</code> . |
| <code>fleqn</code> | Udhævede matematiske formler bliver normalt centreret horisontalt. Hvis man angiver <code>fleqn</code> som valgmulighed, bliver de i stedet rykket ud mod venstre (med lige venstre »formelmarginal«). |
| <code>amsmath</code> | <ul style="list-style-type: none"> • Hvis man vil udfærdige dokumenter som indeholder matematiske formler, er det næsten altid en god idé at indlæse <code>amsmath</code>-pakken; i IMFUFA-$\text{\LaTeX}$ kan man med fordel gøre det ved at angive <code>amsmath</code> som valgmulighed. <p>Grunden til at det anbefales at indlæse <code>amsmath</code> på denne måde, er at IMFUFA-$\text{\LaTeX}$-klasserne altid indlæser <code>amsthm</code>-pakken, men hvis man vil bruge både <code>amsthm</code> og <code>amsmath</code>, skal <code>amsmath</code> indlæses først.</p> <ul style="list-style-type: none"> • Som standard bliver $\text{\LaTeX}$-tekster sat med Computer Modern skriften (som er den der benyttes her). Der findes imidlertid adskillige andre muligheder, både med kommercielle og med 'freeware'-skrifter. Her nævnes tre 'freeware'-muligheder: |
| <code>times</code> | – Valgmuligheden <code>times</code> bevirker at teksten sættes med Times (se skriftprøven i figur 2.1(a)). |
| <code>palatino</code> | – Valgmuligheden <code>palatino</code> bevirker at teksten sættes med Palatino (se skriftprøven i figur 2.1(b)); desuden bliver linjeafstanden øget en smule (fordi Palatino er lidt højere end Computer Modern). |
| <code>ccfonts</code> | – Valgmuligheden <code>ccfonts</code> bevirker at teksten sættes med Concrete (ikke vist). |

I alle tre tilfælde bliver der automatisk valgt en matematikfont der passer (nogenlunde) til tekstfonten.

De tre valg `times`, `palatino` og `ccfonts` er kun til rådighed i IMFUFA- $\text{\LaTeX}$ -klasserne.

Forsider og titelblade

<code>\maketitle</code>	I de fleste tilfælde ønsker man at det færdige dokument har en forside eller et titelblad eller en side 1 med et særligt udseende; også dette sørger dokumentklassen for hvis man beder om det. Kommandoen <code>\maketitle</code> bevirker at der fremstilles en forside eller en første side. Kommandoen skal stå som noget af det første efter <code>\begin{document}</code> .
-------------------------	---

For at $\text{\LaTeX}$ kan vide hvad forsiden skal indeholde, er man nødt til først at levere forskellige oplysninger, normalt i præamblen. Nogle oplysninger er obligatoriske (f.eks. titel), andre er valgfrie idet de på forhånd har fået tildelt en standardværdi (f.eks. har

Aufklärung ist der Ausgang des Menschen aus seiner selbst verschuldeten Unmündigkeit. Unmündigkeit ist das Unvermögen, sich seines Verstandes ohne Leitung eines anderen zu bedienen. Selbstverschuldet ist diese Unmündigkeit, wenn die Ursache derselben nicht am Mangel des Verstandes, sondern der Entschlieung und des Mutes liegt, sich seiner ohne Leitung eines anderen zu bedienen. Sapere aude! Habe Mut dich deines eigenen Verstandes zu bedienen! ist also der Wahlspruch der Aufklärung.

Kant (1784)

(a) Times

Aufklärung ist der Ausgang des Menschen aus seiner selbst verschuldeten Unmündigkeit. Unmündigkeit ist das Unvermögen, sich seines Verstandes ohne Leitung eines anderen zu bedienen. Selbstverschuldet ist diese Unmündigkeit, wenn die Ursache derselben nicht am Mangel des Verstandes, sondern der Entschlieung und des Mutes liegt, sich seiner ohne Leitung eines anderen zu bedienen. Sapere aude! Habe Mut dich deines eigenen Verstandes zu bedienen! ist also der Wahlspruch der Aufklärung.

Kant (1784)

(b) Palatino

Figur 2.1 Den samme tekst i to forskellige skriftsnit, nemlig Times og Palatino. Det er tydeligt at valget af skriftsnit har betydning for det indtryk læseren får af teksten. – Der er i øvrigt tale om Kants definition af begrebet oplysning; en dansk oversættelse af citatet kan f.eks. findes i Bredsdorff (2003, side 11).

datoen standardværdien `\today`). De forskellige dokumentklasser har forskellige sæt obligatoriske og valgfrie oplysninger.

Dokumentklasserne `imfbook`, `imfreport` og `imfart` forventer en `\title`, en `\forfatter` og en `\dato`; værdier tildeles med `\title{\title}`, `\author{\forfatter}` og `\date{\dato}`. På side 107 kan man se hvordan det er foregået for nærværende teksts vedkommende.

Dokumentklassen `imfnote` (der er velegnet til kortere notater) skal bruge en `\title`, en `\forfatter` (default er `\today`) og et `\sted` (default er `IMFUFA`); værdierne tildeles med kommandoerne `\title{\title}`, `\author{\forfatter}` og `\location{\sted}`.

Når man angiver værdier for titel, forfatter osv., kan man bruge `\\` for at markere *linjeskift*; der må ikke være tomme linjer, men der må gerne være en linje der kun indeholder et blanktegn (der skrives som `\_` (»backslash-mellemrum«)).

Et eventuelt *abstract* skrives omgivet af `\begin{abstract}` `\end{abstract}`.

Hvis titlen skal stå på et særligt titelblad (og det ikke sker automatisk), kan man putte `\maketitle` ind mellem `\begin{titlingpage}` og `\end{titlingpage}`.

Læs nærmere om forsider og titelblade i Wilson (2004, kapitel 7).

`\title`
`\author`
`\date`

`\location`
`\\`

`abstract`

`titlingpage`

Breve

Dokumentklassen `imfletter` definerer en **letter**-konstruktion til breve efter danske standarder. Brevets første side får et særligt brevpapir-udseende, med oplysninger om afsender, modtager og dato anbragt passende steder. De følgende sider bliver sat med en særlig sidestil `tplain`; hvis brevet er så kort at det kan stå på en side, vil teksten blive centreret vertikalt på siden. I figur 2.2 ses en skabelon til breve med `imfletter`.

`letter`

Som det fremgår af skabelonen, er der mulighed for at indsætte en lang række data om afsenderen; de ikke-relevante punkter udelades eller sættes til den tomme streng {}. Modtagerens navn (og adresse) skal stå som det obligatoriske argument til `letter`-konstruktionen. Eventuelle linjeskift angives med `\\`. Brevet *skal* begynde med `\opening{<tekst>}` og afsluttes med `\closing[<pos>]{<tekst>}`; det valgfrie argument `<pos>` angiver underskriftens horisontale placering (mulige værdier af `<pos>` er `c`, `l` og `r`).

`\opening`
`\closing`

Et `imfletter`-dokument kan indlæse en eller flere `ils`-filer ('IMFUFA letter style' filer) med kommandoen `\letterstyle{<ils-fil>}` (der skal stå inden `\begin{letter}`). I `ils`-filen kan man f.eks. skrive kommandoer der identificerer afsenderen (altså `\name`, `\address` osv.) således at brugeren kan nøjes med én enkelt `\letterstyle`-kommando. (Se evt. Larsen (2005).)

```
\documentclass{imfletter}

%% alle disse punkter er valgfrie:
\name{<navn>}
\address{<adresse>}
\signature{<underskrift>}
\date{<dato>} % default er \today
\telephone{<telefon>}
\fax{<fax>}
\email{<e-mail>}
\homepage{<hjemmeside>}
\location{<lokale>}
\journal{<afsenderens journalnummer>}
\yourreference{<modtagerens journalnummer>}
%% her slutter de valgfrie punkter.

\begin{document}
\begin{letter}{<modtagers navn og adresse>}
  \opening{<tekst>} % f.eks. Kære Karoline
    <brevteksten>
  \closing[<pos>]{<tekst>} % f.eks. Med venlig
hilsen
\end{letter}
\end{document}
```

Figur 2.2 En brev-skabelon.

3 For let øvede

I dette kapitel forklares hvordan man kan fremstille tabeller og skemaer, hvordan man kan indkapsle en portion tekst i en såkaldt *minipage*, og hvordan man håndterer citater og tekstdele på andre sprog. Endvidere præsenteres de såkaldte ‘floats’, dvs. tekstobjekter som »flyder omkring« således at deres placering ikke er helt forudsigelig; der er tale om figur-objekter, tabel-objekter og marginalnoter.

3.1 Opstilling i skemaer og tabeller

Undertiden har man brug for at stille ting op i skematiske oversigter. Skemaer kan fremstilles ved hjælp af `tabular`-konstruktioner. Det vil vi nu demonstrere ved hjælp af en række eksempler. Her er først en tabel over planeternes middelfastand fra Solen (angivet i astronomiske enheder):

`tabular`

Merkur	0.387	<code>\begin{tabular}{lr}</code>
Venus	0.723	Merkur & 0.387 \\
Jorden	1.000	Venus & 0.723 \\
Mars	1.524	Jorden & 1.000 \\
Jupiter	5.203	Mars & 1.524 \\
Saturn	9.555	Jupiter & 5.203 \\
Uranus	19.218	Saturn & 9.555 \\
Neptun	30.110	Uranus & 19.218 \\
Pluto	39.545	Neptun & 30.110 \\
		Pluto & 39.545
		<code>\end{tabular}</code>

Eks. 3.1

Hele `tabular`-konstruktionen omgives af `\begin{tabular}` og `\end{tabular}`. Umiddelbart efter `\begin{tabular}` skal man specificere hvor mange søjler der skal være, og om de skal venstrejusteres, centreres eller højrejusteres; det gøres ved at man for hver ønsket søjle skriver ét af bogstaverne `l` (left), `c` (centre) og `r` (right). I eksemplet ønskes der altså to søjler hvoraf den første skal venstrejusteres og den anden højrejusteres. Skemaets indhold skrives række for række; tegnet `&` er tabulatortegn, altså skille tegn mellem de enkelte felter, og kommandoen `\\` adskiller de enkelte rækker. – I eksemplet har brugeren skrevet input-teksten pænt op med alle `&`’erne under hinanden; dette er udelukkende af æstetiske grunde (og fordi når man skriver tingene pænt og systematisk op, er det lettere at opdage eventuelle fejl), $\text{\LaTeX}$ er fuldstændig ligeglad.

`l c r`

`&`

`\\`

Man kan få lodrette linjer mellem søjlerne ved at sætte en `|` ind de ønskede steder i søjle-specifikationen, og man kan få vandrette linjer med kommandoen `\hline`; bemærk

`|`

`\hline`

at der skal være `\\` efter den linje der skal understreges. Her er et temmelig grimt eksempel:

Merkur	0.387	<code>\begin{tabular}{ l r }</code>
Venus	0.723	<code>\hline</code>
Jorden	1.000	<code>Merkur & 0.387 \\</code>
Mars	1.524	<code>Venus & 0.723 \\</code>
Jupiter	5.203	<code>Jorden & 1.000 \\</code>
Saturn	9.555	<code>\hline</code>
Uranus	19.218	<code>Mars & 1.524 \\</code>
Neptun	30.110	<code>Jupiter & 5.203 \\</code>
Pluto	39.545	<code>Saturn & 9.555 \\</code>
		<code>Uranus & 19.218 \\</code>
		<code>Neptun & 30.110 \\</code>
		<code>Pluto & 39.545 \\</code>
		<code>\hline\hline</code>
		<code>\end{tabular}</code>

Eks. 3.2

Det anses normalt for dårlig smag at have *lodrette* linjer i tabeller, og vandrette dobbelt-linjer er heller ikke for kønt. Her er en bedre løsning:

Merkur	0.387	<code>\begin{tabular}{lr}</code>
Venus	0.723	<code>\toprule</code>
Jorden	1.000	<code>Merkur & 0.387 \\</code>
Mars	1.524	<code>Venus & 0.723 \\</code>
Jupiter	5.203	<code>Jorden & 1.000 \\</code>
Saturn	9.555	<code>\cmidrule(lr){1-2}</code>
Uranus	19.218	<code>Mars & 1.524 \\</code>
Neptun	30.110	<code>Jupiter & 5.203 \\</code>
Pluto	39.545	<code>Saturn & 9.555 \\</code>
		<code>Uranus & 19.218 \\</code>
		<code>Neptun & 30.110 \\</code>
		<code>Pluto & 39.545 \\</code>
		<code>\bottomrule</code>
		<code>\end{tabular}</code>

Eks. 3.3

`\toprule` I eks. 3.3 er den øverste vandrette linje fremstillet med `\toprule` og den nederste vandrette linje med `\bottomrule`; disse er lidt tykkere end midterlinjen. Den enkleste måde at lave midterlinjen på er med kommandoen `\midrule` der vil give en linje af samme længde som top- og bundlinjen. For at få en linje der er 'trimmet' lidt i enderne, har vi imidlertid brugt kommandoen `\cmidrule`. Det simpleste kald af `\cmidrule` er af formen `\cmidrule{<a-b>}` som resulterer i en linje fra og med søjle nr. *a* til og med søjle nr. *b*. Det mere avancerede kald der er benyttet her, er af formen `\cmidrule(<trim>){<a-b>}` hvor *<trim>* er *l* eller *r* eller *lr* (*l* = left, *r* = right).

Kommandoerne `\toprule`, `\bottomrule`, `\midrule` og `\cmidrule` er til rådighed i IMFUFA- $\text{\LaTeX}$ via `memoir`-klassen (Wilson, 2004, afsnit 12.5). Hvis man benytter andre klasser, kan man indlæse pakken `booktabs` som også definerer disse kommandoer.

Nu udbygger vi tabellen med en søjle der viser planeternes masser (angivet med Jordmassen som enhed). Samtidig skal de to talsøjler have overskrifter. Her er et forslag hvor de tre søjler er hhv. venstrejusteret (l), højrejusteret (r) og centreret (c):

Eks. 3.4

	afstand	masse	
Merkur	0.387	0.055	
Venus	0.723	0.815	
Jorden	1.000	1.000	
Mars	1.524	0.107	
Jupiter	5.203	317.83	
Saturn	9.555	95.159	
Uranus	19.218	14.500	
Neptun	30.110	17.204	
Pluto	39.545	0.0025	

```

\begin{tabular}{lrc}
& afstand & masse \\
\toprule
Merkur & 0.387 & 0.055 \\
Venus & 0.723 & 0.815 \\
Jorden & 1.000 & 1.000 \\
Mars & 1.524 & 0.107 \\
Jupiter & 5.203 & 317.83 \\
Saturn & 9.555 & 95.159 \\
Uranus & 19.218 & 14.500 \\
Neptun & 30.110 & 17.204 \\
Pluto & 39.545 & 0.0025 \\
\bottomrule
\end{tabular}

```

De forskellige planetmasser har forskellige antal cifre både efter og før kommaet, og hvordan får man så kommaerne til at stå under hinanden? Her er en løsning:

Eks. 3.5

	afstand	masse	
Merkur	0.387	0.055	
Venus	0.723	0.815	
Jorden	1.000	1.000	
Mars	1.524	0.107	
Jupiter	5.203	317.83	
Saturn	9.555	95.159	
Uranus	19.218	14.500	
Neptun	30.110	17.204	
Pluto	39.545	0.0025	

```

\begin{tabular}{lrD..{3.4}}
& afstand&\multicolumn{1}{c}{masse}\\
\toprule
Merkur & 0.387 & 0.055 \\
Venus & 0.723 & 0.815 \\
Jorden & 1.000 & 1.000 \\
Mars & 1.524 & 0.107 \\
Jupiter & 5.203 & 317.83 \\
Saturn & 9.555 & 95.159 \\
Uranus & 19.218 & 14.500 \\
Neptun & 30.110 & 17.204 \\
Pluto & 39.545 & 0.0025 \\
\bottomrule
\end{tabular}

```

I søjlespecifikationen er c'et der centrerede tredje søjle, nu erstattet af D..*3.4*. Her er D en »kommando« med syntaksen D{*inputsep*}{*outputsep*}{*før.efter*}. Argumentet *inputsep* er det tegn der bruges som decimal»komma« i tex-filen, *outputsep* er det tegn der skal bruges som decimal»komma« i det færdige resultat, og *før.efter* angiver det maksimale antal cifre før og efter »kommaet«; i eksemplet er *inputsep* og *outputsep* begge tegnet punktum (.), og der forventes højst tre cifre foran kommaet og fire cifre efter. Den tredje søjles overskrift bliver nu fremstillet med \multicolumn{1}{c}{masse}. Hvis man ikke havde gjort det, ville resultatet blive at teksten »masse« blev placeret sådan at et underforstået punktum efter den ville flugte med de øvrige punktummer.

D

`\multicolumn` Kommandoen `\multicolumn{⟨antal⟩}{⟨pos⟩}{⟨indhold⟩}` indsætter `⟨indhold⟩` i tabellen i en »celle« som spænder over `⟨antal⟩` søjler; argumentet `⟨pos⟩` er `l`, `c` eller `r` og angiver om `⟨indhold⟩` skal venstrejusteres, centreres eller højrejusteres. Eksempel:

A	B	C	D	<code>\begin{tabular}{cccc}</code>
A	BC		D	<code>A & B & C & D \\</code>
A	BC		D	<code>A & \multicolumn{2}{l}{BC} & D \\</code>
A	BC		D	<code>A & \multicolumn{2}{c}{BC} & D \\</code>
A	BC		D	<code>A & \multicolumn{2}{r}{BC} & D</code>
				<code>\end{tabular}</code>

Eks. 3.6

I eks. 3.6 er der fire søjler der alle har samme specifikation (`c`); i stedet for `cccc` kunne man skrive `*{4}{c}`. Generelt kan man skrive `*{⟨antal⟩}{⟨spec⟩}` for at få `⟨antal⟩` kopier af specifikationen `⟨spec⟩`.

Normalt indretter `tabular`-konstruktionen søjlernes bredde efter bredden af det der står i dem. Men undertiden vil man måske selv have et ord at skulle have sagt i den forbindelse. Med `p{⟨bredde⟩}`, `m{⟨bredde⟩}` eller `b{⟨bredde⟩}` fås en søjle af bredde `⟨bredde⟩`:

Linjen	Jeg er på øen i åen.	<code>\begin{tabular}{lm{2cm}}</code>
		<code>Linjen & Jeg er på øen i åen.</code>
		<code>\end{tabular}</code>

Eks. 3.7

Teksten i en `p`-, `m`- eller `b`-søjle formatteres i linjer af den angivne bredde, og hhv. første linje, midten eller nederste linje kommer til at flugte med den aktuelle tabel-linje.

Undertiden skal alle felterne i en bestemt søjle behandles på en særlig måde, eksempelvis skrives med fed skrift eller sættes i matematik-mode. Det klares med `>{⟨foran⟩}` og `>{⟨efter⟩}` der skrives umiddelbart før hhv. efter søjlespecifikationen, og hvor virkningen er at hvert enkelt felt i den pågældende søjle bliver formatteret som om der stod `⟨foran⟩` lige før og `⟨efter⟩` lige efter den L^AT_EX-kode der hører til feltet. I eks. 3.8 bliver der således skrevet `\bfseries` i starten af hvert enkelt felt i anden søjle, dvs. hele anden søjle sættes med fed skrift, og i eks. 3.9 sættes `$` foran og efter indholdet af hvert felt i anden søjle, dvs. hele anden søjle er i matematik-mode:

<code>r1s1</code>	<code>r1s2</code>	<code>r1s3</code>	<code>\begin{tabular}{c >{\bfseries}c c}</code>
<code>r2s1</code>	<code>r2s2</code>	<code>r2s3</code>	<code>r1s1 & r1s2 & r1s3 \\</code>
<code>r3s1</code>	<code>r3s2</code>	<code>r3s3</code>	<code>r2s1 & r2s2 & r2s3 \\</code>
			<code>r3s1 & r3s2 & r3s3</code>
			<code>\end{tabular}</code>

Eks. 3.8

addition	$x + y$	<code>\begin{tabular}{l >{\$}c<{\$}}</code>
subtraktion	$x - y$	<code>addition & x+y \\</code>
multiplikation	$x \cdot y$	<code>subtraktion & x-y \\</code>
		<code>multiplikation & x \cdot y</code>
		<code>\end{tabular}</code>

Eks. 3.9

Skemaets placering

Én ting er at få det rigtige indhold i skemaet, noget andet er at få placeret skemaet rigtigt i forhold til den omgivende tekst. $\LaTeX$ behandler et skema som et tekstobjekt der indgår i den almindelige tegnstrøm:

EKS. 3.10

$\LaTeX$ betragter skemaet	A	B	som et	$\{\LaTeX\}$ betragter skemaet
almindeligt tekstobjekt.	C	D		$\begin{tabular}{cc}$
				A & B \\ C & D
				$\end{tabular}$
				som et almindeligt tekstobjekt.

Som det ses bliver skemaet centreret vertikalt i forhold til teksten, men det kan man let ændre ved at angive plads-specifikatoren `b` (bund) eller `t` (top):

EKS. 3.11

$\LaTeX$ betragter skemaet	A	B	som et	$\{\LaTeX\}$ betragter skemaet
almindeligt tekstobjekt.	C	D		$\begin{tabular}[t]{cc}$
				A & B \\ C & D
				$\end{tabular}$
				som et almindeligt tekstobjekt.

Hvis man ønsker skemaet centreret på en linje for sig selv, kan det gøres på forskellige måder. Én mulighed er at bruge `\hfill` der laver horisontal udfyldning (i eksemplet bruges `\mbox{}` til at producere et nødvendigt tekstobjekt der ingenting fylder og ikke kan ses, men som dog er »noget«):

 $\hfill$

EKS. 3.12

Her bliver skemaet centreret:	Her bliver skemaet centreret:
A B	$\newline\mbox{}\hfill$
C D	$\begin{tabular}{cc}$
Det kom altså til at se sådan ud.	A & B \\ C & D
	$\end{tabular}$
	$\hfill\mbox{}\newline$
	Det kom altså til at se sådan ud.

En anden mulighed er at benytte `\centering` der bevirker at teksten fra og med det aktuelle afsnit centrerer; rækkevidden afgrænses med `{ }`.

 $\centering$

EKS. 3.13

Her bliver skemaet centreret:	Her bliver skemaet centreret:
A B	$\par$
C D	$\{\centering$
Det kom altså til at se sådan ud.	$\begin{tabular}{cc}$
	A & B \\ C & D
	$\end{tabular}\par$
	$\noindent$
	Det kom altså til at se sådan ud.

- Et »afsnit« er det der står mellem én blank linje (eller `\par`) og en anden.

Hvis man vil have skemaet til at stå fremhævet og centreret, kan man putte det ind i `center` center-omgivelser:

Her bliver skemaet fremhævet og centreret:	Her bliver skemaet fremhævet og centreret:
<pre> A B C D </pre>	<pre> \begin{center} \begin{tabular}{cc} A & B \\ C & D \end{tabular} \end{center} </pre>
Det kom altså til at se sådan ud.	Det kom altså til at se sådan ud.

EKS. 3.14

3.2 Figurer og tabeller

Almindeligvis bytter $\text{\LaTeX}$ ikke om på rækkefølgen af ord og formler osv., alt kommer i samme rækkefølge som forfatteren skriver det. Figurer og tabeller er imidlertid en undtagelse; de bliver af pladsmæssige og æstetiske grunde opfattet som selvstændige tekstenheder der blot skal placeres et eller andet passende sted i nærheden af den sammenhæng hvori de indgår. Tekstobjekter som figurer og tabeller kaldes i $\text{\LaTeX}$ -jargonen for ‘floats’ fordi deres placering er lidt flydende. $\text{\LaTeX}$ har et sæt regler for hvordan floats skal placeres (helst øverst på en side, eventuelt nederst; kun en vis brøkdel af siden må optages af floats, medmindre det er en side der kun bruges til floats, osv.). Det skal understreges at pointen med floats netop er at deres placering *ikke* er fuldstændig fast, og at de altså *ikke* er tænkt til at skulle stå et helt bestemt sted i teksten.

figure Figurer hhv. tabeller fremstilles efter grundopskriften

table	<pre> \begin{figure}[\langle pos \rangle] \langle indhold \rangle \caption{\langle tekst \rangle} \label{\langle mærke \rangle} \end{figure} </pre>	hhv.	<pre> \begin{table}[\langle pos \rangle] \caption{\langle tekst \rangle} \label{\langle mærke \rangle} \langle indhold \rangle \end{table} </pre>
--------------	---	------	---

Argumentet $\langle pos \rangle$ angiver figurens/tabellens ønskede position på siden (øverst eller nederst); det er normalt fornuftigt at bruge værdien **tb!** (hvor **t** og **b** angiver at figuren/tabellen må placeres øverst eller nederst på siden, og udråbstegnet betyder at $\text{\LaTeX}$ gerne må administrere sine regler lidt lempeligt). En anden mulighed for $\langle pos \rangle$ er **p** (som page (side)); så bliver figuren/tabellen placeret på en side der kun bruges til ‘floats’.

- $\text{\LaTeX}$ kan somme tider have vanskeligheder med at placere floats, især hvis de fylder meget i højden. Det går undertiden lettere med placeringen hvis man angiver **tb!** som $\langle pos \rangle$. Det anbefales også at sørge for at der er en blank linje umiddelbart før $\text{\LaTeX}$ `\begin{figure}/\begin{table}` og en blank linje umiddelbart efter $\text{\LaTeX}$ `\end{figure}/\end{table}`.

En liden demonstrationsfigur

Figur 3.1 Figurtekst.

- Hvis $\LaTeX$ ikke kan finde ud af at placere en figur eller tabel, så bliver den gemt til sidst (eller til der kommer en `\clearpage`-kommando); og eftersom der aldrig byttes om på den indbyrdes rækkefølge af f.eks. figurer, så betyder det at hvis én figur gemmes til sidst, så gemmes også alle de følgende figurer til sidst.

Som `\indhold` i en `figure` eller `table` kan man skrive vilkårlig $\LaTeX$ -tekst, f.eks.

- almindelig tekst, der med fordel kan skrives i en `minipage`, se side 40;
- en `\includegraphics`-kommando der indlæser et billede, se side 41;
- en eller flere skematiske opstillinger, se side 21.

Figurteksten/tabelteksten fremstilles med `\caption[<kort-tekst>]{<tekst>}`. Indholdet af *<tekst>* bliver skrevet i figuren/tabellen, og værdien af *<kort-tekst>* vil optræde i en eventuel »Liste over figurer« eller »Liste over tabeller« (som man fremstiller med `\listoffigures` hhv. `\listoftables`); hvis man ikke skriver nogen *<kort-tekst>*, bruges *<tekst>* i disse lister.

`\caption`

- I `\caption`-kommandoen må *<kort-tekst>* ikke være for lang(!), i særdeleshed må den ikke indeholde blanke linjer. Det betyder at hvis man har tænkt sig at *<tekst>* skal være meget lang eller indeholde blanke linjer (eller `\par`), så *skal* man angive en anden værdi for *<kort-tekst>*. Dette gælder også selv om man ikke har planer om at lave lister over figurer eller tabeller.
- Bemærk at en eventuel `\label`-kommando bør stå *efter* `\caption`-kommandoen (og inde i `figure`/`table`-konstruktionen).

Eksempel: De linjer der er vist i eks. 3.15, frembringer figur 3.1 som $\LaTeX$ vil placere et sted her i nærheden (desværre kan der ikke komme figurer og tabeller ind i eksempelkasserne):

```
\begin{figure}[tb!]
  \centering
  \Huge\bfseries En liden demonstrationsfigur
  \caption{Figurtekst.}
  \label{fig:Figur1}
\end{figure}
```

Eks. 3.15

Der er mange muligheder for at ændre figurtekstens udseende, se f.eks. Wilson (2004, kapitel 11).

Dobbeltfigurer

Hvis man har meget små figurer/tabeller, kan det være formålstjenligt at placere dem ved siden af hinanden som det ses i figur 3.2 og 3.3 der er fremstillet som vist i eks. 3.16. Man bruger `\begin{figure}` `\end{figure}` til at afgrænse det der skal udgøre

Figur A

Figur 3.2 Dette er den første figur.

Figur B

Figur 3.3 Dette er den anden figur.

Figur C1

(a) Første delfigur.

Figur C2

(b) Anden delfigur.

Figur 3.4 Denne figur har to delfigurer

den samlede ‘float’, og man fremstiller delfigurerne i hver deres `minipage` hvor man angiver figurindhold og figurtekst (i eksemplet er der ikke gjort noget for at centrere figurindholdet i forhold til den pågældende `minipage`). Man kan placere `minipage`’rne i forhold til hinanden som man ønsker, her bliver de centreret horisontalt ved hjælp af `\hfill`-kommandoerne.

```
\begin{figure}[tb!]
\hfill
\begin{minipage}{0.3\textwidth}
\centering\Huge\bfseries Figur A
\caption{Dette er den første figur.}
\label{fig:FigurA}
\end{minipage}
\hfill
\begin{minipage}{0.3\textwidth}
\centering\Huge\bfseries Figur B
\caption{Dette er den anden figur.}
\label{fig:FigurB}
\end{minipage}
\hfill\
\end{figure}
```

Eks. 3.16

Eksemplet er pænt (måske for pænt) i den forstand at de to delfigurers indhold har samme højde og deres figurtekst samme antal linjer, således at `minipage`’rne får samme højde. Hvis dette ikke havde været tilfældet, skulle man måske angive hvordan `minipagen` skulle justeres (se side 41).

`\subcaption` Man kan også have egentlige del-figurer der får deres figurtekst med kommandoen `\subcaption{<figurtekst>}`; figur 3.4 er fremstillet sådan (hvor man bemærker at `\label`-kommandoen står *inde i* `\subcaption`):

```

\begin{figure}[tb!]
\hfill
\begin{minipage}{0.3\textwidth}
\centering\Huge\bfseries Figur C1
\subcaption{Første delfigur.\label{fig:figC1}}
\end{minipage}
\hfill
\begin{minipage}{0.3\textwidth}
\centering\Huge\bfseries Figur C2
\subcaption{Anden delfigur.\label{fig:figC2}}
\end{minipage}
\hfill\
\caption{Denne figur har to delfigurer}
\label{fig:FigurC}
\end{figure}

```

EKS. 3.17

Man kan nu henvise til både figur og delfigurer:

`\subcaptionref`

Delfigur (a) i figur 3.4 refereres som figur 3.4(a).	Delfigur <code>\subcaptionref{fig:figC1}</code> i <code>\fref{fig:FigurC}</code> refereres som <code>\fref{fig:figC1}</code> .
--	--

EKS. 3.18

Andre muligheder er `\subbottom[⟨til lof⟩][⟨figurtekst⟩]{⟨indhold⟩}` (jf. figur 3.5):

`\subbottom`

```

\begin{figure}[tb!]
\hfill
\subbottom[Første delfigur.\label{fig:figD1}]{\Huge\bfseries Figur D1}
\hfill
\subbottom[Anden delfigur.\label{fig:figD2}]{\Huge\bfseries Figur D2}
\hfill\
\caption{Denne figur har to delfigurer}
\label{fig:FigurD}
\end{figure}

```

EKS. 3.19

eller `\subtop[⟨til lof⟩][⟨figurtekst⟩]{⟨indhold⟩}` (jf. figur 3.6):

`\subtop`

```

\begin{figure}[tb!]
\hfill
\subtop[Første delfigur.\label{fig:figE1}]{\Huge\bfseries Figur E1}
\hfill
\subtop[Anden delfigur.\label{fig:figE2}]{\Huge\bfseries Figur E2}
\hfill\
\caption{Denne figur har to delfigurer}
\label{fig:FigurE}
\end{figure}

```

EKS. 3.20

Figur D1

(a) Første delfigur.

Figur D2

(b) Anden delfigur.

Figur 3.5 Denne figur har to delfigurer

(a) Første delfigur.

Figur E1

(b) Anden delfigur.

Figur E2

Figur 3.6 Denne figur har to delfigurer

Det valgfrie argument $\langle \textit{til lof} \rangle$ er den tekst der kommer i en eventuel Liste over figurer.

Man kan naturligvis henvise til såvel delfigurerne som hele figuren:

Delfigur (a) i figur 3.6 refereres som figur 3.6(a).

Delfigur `\subcaptionref{fig:figE1}` i `\fref{fig:FigurE}` refereres som `\fref{fig:figE1}`.

EKS. 3.21

De omtalte `\sub...`-kommandoer findes i memoir (læs om dem og meget mere i Wilson (2004, kapitel 11)) og derigennem i IMFUFA- $\text{\LaTeX}$.

Bemærk at man er nødt til at skrive `\newsubfloat{figure}` i præamblen for at fortælle at man ønsker at figurer skal kunne have delfigurer.

Indrammede floats

`ibox` IMFUFA- $\text{\LaTeX}$ stiller float-typen `ibox` til rådighed; boks 3.1 viser hvordan den ser ud. – Syntaksen for `ibox` er den samme som for `figure` og `table`.

Musica est exercitium arithmeticae occultum nescientis se numerare animi.

Musik er en hemmelig aritmetisk øvelse for sjælen som ikke ved at den tæller.

Boks 3.1 Leibniz' bud på hvad musik blandt andet er.

3.3 Sidebemærkninger

`\sidebar`

Det kan se smart ud at skrive små bemærkninger ude i marginen, og undertiden kan læseren måske endda have glæde af det. Kommandoen `\sidebar{\langle tekst \rangle}` skriver tekst i ydermarginen, begyndende et sted »nær toppen«. Øverst næste side ses et eksempel (fra Emma Gad: Takt og Tone).

`\sidebarfont`
`\sidebarwidth`
`\sidebarhsep`

Man kan regulere udseendet af `\sidebar`'en. Kommandoen `\sidebarfont` indeholder den fontspecifikation der benyttes til teksten, længden `\sidebarwidth` angiver bredden af sidebar'en, og `\sidebarhsep` angiver afstanden mellem hovedteksten og sidebar'en (se også (Wilson, 2004, afsnit 13.3)). I nærværende tekst bruges

EKS. 3.22

```
\renewcommand*{\sidebarfont}{\footnotesize}
\setlength{\sidebarwidth}{0.6\foremargin}
\setlength{\sidebarhsep}{0.2\foremargin}
```

Kommandoen `\sidebar` er ikke 100% driftssikker, f.eks. virker den ikke hvis den skrives lige efter en `\section`-kommando. Hvis man har både `\sidebar`'er og `\marginpar`'er på samme side, afhænger resultatet af om `\sidebar`-kommandoen står før eller efter `\marginpar`-kommandoen. (Det er årsagen til at den sædvanlige praksis med at skrive kommandonavne i marginen er sat ud af kraft på denne side.)

Hvis man vil skrive margintekst ud for en bestemt linje (så vidt det er muligt), kan man benytte `\marginpar` eller `\sidepar`. Kommandoen `\marginpar[⟨venstre⟩]{⟨højre⟩}` placerer tekst i den yderste margin; hvis man har angivet både `⟨venstre⟩` og `⟨højre⟩`, så benyttes `⟨venstre⟩` når noten står til venstre for teksten, og `⟨højre⟩` benyttes når noten står til højre for teksten. Kommandoen `\sidepar[⟨venstre⟩]{⟨højre⟩}` placerer tekst i den inderste margin; man kan dog flytte den til ydermarginen med kommandoen `\reversesideparfalse`.

Eksempel: I nærværende tekst er eksempelkassernes numre anbragt (i indermarginen) på en måde der svarer til `\sidepar[\raggedleft ⟨nr⟩]{\raggedright ⟨nr⟩}`, og kommandonavne mv. er anbragt (i ydermarginen) på lignende måde med `\marginpar`.

- $\text{\LaTeX}$ kan undertiden have lidt svært ved at finde ud af om en marginalnote står på en lige eller ulige side, og så kan det der skulle stå i ydermarginen, blive placeret i indermarginen og omvendt. Det kan afhjælpes med kommandoen `\strictpagechecktrue`.

`\sidebar`'er og `\sidepar`'er findes kun i `memoir` (og dermed også i `IMFUFA- $\text{\LaTeX}$`).

Driv ikke et Barns Eksamenslæsning ud over det Rimelige; Overanstrengelse lige før en Eksamen er af det Onde, og den Smule, der kan indhentes i et Par Dage, forslaar ikke meget overfor lang Tids Forsømmelse.

3.4 Citater og forskellige sprog

Man kan have brug for at indsætte særligt fremhævede tekstdele, eksempelvis citater, og de skal måske være på et andet sprog end hovedteksten. Dette afsnit fortæller om nogle måder at gøre det på.

Standard $\text{\LaTeX}$ indeholder `quote`- og `quotation`-konstruktioner. De næste to eksempler viser hvordan de fungerer (men se også side 33f om kommandoen `\blockquote` og dens slægtninge):

EKS. 3.23

Hovedteksten.	Hovedteksten.
Højre- og venstremargin rykkes ind i forhold til den omgivende tekst.	<code>\begin{quote}</code> Højre- og venstremargin rykkes ind i forhold til den omgivende tekst.
Bemærk hvordan »nyt afsnit« vises.	Bemærk hvordan <code>\enquote{nyt afsnit}</code> vises.
Hovedteksten fortsat.	<code>\end{quote}</code> Hovedteksten fortsat.

Hovedteksten.	Hovedteksten.
Højre- og venstremargin rykkes ind i forhold til den omgivende tekst.	<code>\begin{quotation}</code> Højre- og venstremargin rykkes ind i forhold til den omgivende tekst.
Bemærk hvordan »nyt afsnit« vises.	Bemærk hvordan <code>\enquote{nyt afsnit}</code> vises.
Hovedteksten fortsat.	<code>\end{quotation}</code> Hovedteksten fortsat.

EKS. 3.24

Om flersprogede tekster

En del af L^AT_EXs adfærd skal afhænge af hvilket sprog der skrives på. Det gælder for eksempel benævnelsen af kapitler, figurer og tabeller, den måde datoen skrives på og reglerne for orddeling. For at gøre det mere besværligt er det jo ikke altid at hele dokumentet skrives på samme sprog, nogle forfattere ynder at smide om sig med citater på adskillige sprog. Med babel-pakken (Braams, 2005) er der fastlagt en standard for hvordan sprogvælger og -skift i en given tekst skal håndteres. Set fra brugerens synspunkt er det ganske enkelt.

1. I dokumentklasseerklæringen angiver man navnene på de ønskede sprog som valgmuligheder, f.eks.

```
\documentclass[danish,english,ngerman]{imfbook}
```

(Som det ses skal man bruge de engelske navne.) Det sprog der kommer sidst i listen, bliver valgt til »gældende sprog«; i det viste eksempel har forfatteren tænkt sig at benytte sprogene dansk, engelsk og tysk (med ny retskrivning), og tysk er det sprog der er gældende ved dokumentets begyndelse.

- Hvis man ikke specificerer noget sprog, vælger IMFUFA-L^AT_EX-klasserne automatisk sproget dansk; det har af tekniske grunde sin pris: Hvis man angiver nogle sprog som valgmuligheder til en IMFUFA-L^AT_EX-klasse, så *skal* et af dem være **danish**.

Hvis man benytter Standard L^AT_EX, skal man selv sørge for at indlæse pakken babel.

`\selectlanguage`

2. Undervejs i dokumentet skifter man sprog med `\selectlanguage{<prog>}` der blandt andet ændrer de indbyggede tekstkonstanter, datoformatet og orddelingsreglerne.

I dag er det den 3. juli 2006.	I dag er det den <code>\today</code> .
Today is July 3, 2006.	<code>\selectlanguage{english}</code> Today is <code>\today</code> .

`otherlanguage`

EKS. 3.25

3. En anden mulighed er at sætte den tekst der er skrevet i sproget `<prog>` ind mellem `\begin{otherlanguage}{<prog>}` og `\end{otherlanguage}`. Derved får man – ligesom med `\selectlanguage` – ændret både orddelingsreglerne og de

indbyggede tekstkonstanter mm. Hvis man kun ønsker at ændre orddelingsreglerne, skal man bruge `otherlanguage*` i stedet for `otherlanguage`.

`otherlanguage*`

EKS. 3.26

I dag er det den 3. juli 2006.	I dag er det den \today.
Today is 3. juli 2006.	\begin{otherlanguage*}{english}
	Today is \today.
	\end{otherlanguage*}

4. Til korte tekststumper er det formentlig som oftest lettere at benytte kommandoen `\foreignlanguage{<sprog>}{<tekst>}` der har samme virkning som hvis man putter `<tekst>` ind i en `otherlanguage*`-konstruktion.

`\foreignlanguage`

EKS. 3.27

Vi så stykket `\foreignlanguage{english}{\textit{The Merchant of Venice}}`.

Man kan læse mere om `babel`-pakken og dens kommandoer til skift mellem sprog i Braams (2005).

Bedre citater med `csquotes`

Pakken `csquotes` (der automatisk indlæses af `IMFUFUA-LATEX`) indeholder en række fleksible redskaber til markering af citater. I det følgende omtales nogle af disse.

Vi har tidligere (side 11) mødt kommandoen `\enquote` der sætter sit argument i anførselstegn; dette sker faktisk under hensyntagen til de normer for udseende og placering af anførselstegn som gælder for det »gældende sprog«.

`\enquote`

Kommandoen `\blockquote[<kilde>]{<tekst>}` skriver `<tekst>` omgivet af anførselstegn; hvis `<kilde>` er angivet, bliver den skrevet efter citatet:

`\blockquote`

EKS. 3.28

»Hvad det især gælder om er at holde Forfatterens Idé klar.« (Karen Blixen)	\blockquote[Karen Blixen]{Hvad det især gælder om er at holde Forfatterens Idé klar.}
---	---

Det raffinerede ved `\blockquote` er at korte citater bliver sat som vi lige har set, men lange citater sættes på denne måde (hvor citatet nu også gerne skulle være korrekt (efter Blixen (1960)):

Hvad det, mine Børn, især gælder om i en Marionetkomedie er at holde Forfatterens Idé klar. Det er en Hemmelighed, som jeg dog vil fortælle Dig, at dette er den sande Lykke, som Folk leder om paa andre Steder. Ja, det er det velsignede ved at spille med i en Marionetkomedie, da jeg nu langt om længe er kommet ind i en, vil jeg heller aldrig ud igen. Men o I mine Medspillende, hold Forfatterens Idé klar, ja driv den ud i dens yderste Konsekvens.

(Karen Blixen i »Sandhedens Hævn«)

Man kan konfigurere den måde som `\blockquote` og dens slægtninge opfører sig på, se Lehman (2005).

Til citater på fremmede sprog kan benyttes `\foreignquote{<sprog>}{<tekst>}` eller `\hyphenquote{<sprog>}{<tekst>}`:

`\foreignquote`
`\hyphenquote`

„Nur eine Waffe taugt“	<code>\foreignquote{ngerman}{Nur eine Waffe taugt}</code>
»Nur eine Waffe taugt«	<code>\hyphenquote{ngerman}{Nur eine Waffe taugt}</code>

Eks. 3.29

Kommandoene `\foreignquote{⟨sprog⟩}{⟨tekst⟩}` svarer til at `\enquote{⟨tekst⟩}` bliver udført efter de regler der gælder for sproget `⟨sprog⟩`. Omvendt afstedkommer kommandoene `\hyphenquote{⟨sprog⟩}{⟨tekst⟩}` at `⟨tekst⟩` bliver udformet efter de regler der gælder for sproget `⟨sprog⟩`, og derefter bliver der sat anførselstegn efter de regler der gælder for hovedteksten. Varianterne `\foreignquote*`, `\hyphenquote*` (og den tidligere omtalte `\enquote*`) benytter anførselstegn hørende til det inderste niveau.

`\foreignquote*`
`\hyphenquote*`
`\foreignblockquote`

Hvis man ønsker en ‘blockquote’ på et fremmed sprog, kan man benytte kommandoene `\foreignblockquote{⟨sprog⟩}[⟨kilde⟩]{⟨tekst⟩}` der svarer til en kombination af `\foreignlanguage` og `\blockquote`, f.eks.

<code>\foreignblockquote{ngerman}[Nietzsche]{Im Grunde gab es nur Einen Christen, und der starb am Kreuz.}</code>

Eks. 3.30

`\hyphenblockquote`

Alternativt kan man benytte kommandoene `\hyphenblockquote`; forskellen mellem `\foreignblockquote` og `\hyphenblockquote` er den samme som mellem `\foreignquote` og `\hyphenquote`.

Her er et eksempel på resultatet af en `\foreignblockquote` anvendt på et langt citat (der beskriver Dr Watsons forbløffelse over Sherlock Holmes’ selektive viden); anførselstegnene er naturligvis fremstillet med `\enquote`, og som det ses er det den rigtige (engelske) model:

My surprise reached a climax, however, when I found incidentally that he was ignorant of the Copernican Theory and of the composition of the Solar System. That any civilized human being in this nineteenth century should not be aware that the earth travelled round the sun appeared to be to me such an extraordinary fact that I could hardly realize it.

“You appear to be astonished,” he said, smiling at my expression of surprise.
 “Now that I do know it I shall do my best to forget it.”

(A Study In Scarlett, Chapter 2)

3.5 Overspring tekst

Hvordan beder man L^AT_EX om at springe et vist tekstafsnit over? Hvis man bruger en hensigtsmæssig teksteditor, kan man lynhurtigt markere en blok og »udkommentere« den (dvs. sætte kommentartegn (%) i begyndelsen af hver linje), og man kan lynhurtigt fjerne det igen.

`\newcomment`

Undertiden kan der være brug for andre metoder. Man kan definere en »comment-konstruktion« med kommandoene `\newcomment{⟨navn⟩}`; herefter vil alt hvad der står mellem `\begin{⟨navn⟩}` og `\end{⟨navn⟩}` blive ignoreret. Hvis man for eksempel skriver `\newcomment{kommentar}`, så vil alt mellem `\begin{kommentar}` og den tilhørende `\end{kommentar}` blive ignoreret.

Eks. 3.31

```

Her er noget tekst. Her er noget mere tekst.
\begin{kommentar}%
    Denne del af teksten
    springes over.
\end{kommentar}%
Her er noget mere tekst.

```

Man kan slå `comment`-konstruktionen `<navn>` fra med `\commentsoff{<navn>}`. Eksempel: Hvis man (efter `\newcomment{komentar}`) skriver `\commentsoff{komentar}`, ophæves virkningen af `komentar`. Man kan godt have flere »`comment`-konstruktioner«.

`\commentsoff`

- Kommandoerne `\newcomment` og `\commentsoff` kan kun bruges i præamblen.

De omtalte »`comment`-konstruktioner« er mulige i `memoir` og `IMFUFA-LATEX`.

3.6 Sidens udformning og opbygning

`LATEX` sørger automatisk for en passende udformning af siden, herunder for størrelsen af marginer, nummerering af sider og placering af sidetal. I visse dokumentklasser har siderne desuden en såkaldt *levende kolumnetitel* (således som det ses øverst på siderne i dette hæfte). Der er særlige regler for udformningen af den første side i et kapitel og den første side i det samlede dokument.

Brugeren kan vælge mellem forskellige standardudformninger af den normale side med kommandoen `\pagestyle{<stil>}` der ændrer fra og med indeværende side, og `\thispagestyle{<thispagestyle>}` der kun ændrer indeværende side. I dokumentklasserne `imfreport` og `imfbook` kan `<stil>` være `lheadings` eller `lplain`; i `imfnote` kan den være `nheadings` eller `nplain`; i `imfletter` er den `tplain` (dog ikke på første side). Desuden kan man altid bruge `empty`, `plain` og `headings`; herudover definerer `memoir`-klassen et antal sidestile (Wilson, 2004, kapitel 14).

`\pagestyle``\thispagestyle`

Med nogle stile får siderne en levende kolumnetitel med information om kapitel og afsnit. Det foregår (af sig selv) på den måde at `LATEX` henter information fra `\chapter-` og `\section-`kommandoerne og benytter den til kolumnetitlerne. Man kan gribe ind i automatikken ved at angive et ekstra argument til `\chapter-` eller `\section-`kommandoen: `\chapter[<kort titel>]{<titel>}` bevirker at `<kort titel>` benyttes i kolumnetitlerne (og i indholdsfortegnelsen); der gælder tilsvarende for `\section`.

Sideskift

`LATEX` betjener sig af en del indbyggede regler for hvornår teksten må og hvornår den skal begynde på en ny side, men det er ikke altid at resultatet bliver tilfredsstillende. Så må brugeren gribe ind. Der er kommandoer der på forskellig måde fremprovokerer eller forhindrer et sideskift:

- Kommandoen `\newpage` skifter råt og brutalt til ny side. Kommandoen `\clearpage` sørger for at eventuelle ophobede figurer og tabeller mv. bliver skrevet ud, og derefter begyndes på ny side. Kommandoen `\cleardoublepage` virker ligesom `\clearpage`, blot begyndes der på en ny højreside.

`\newpage``\clearpage``\cleardoublepage`

<code>\cleartorecto</code> <code>\cleartoverso</code>	I memoir og dermed i IMFUFA-$\text{\LaTeX}$ findes flere kommandoer i samme kategori (jf. Wilson, 2004, afsnit 17.12). Her skal kun nævnes <code>\cleartorecto</code> der skifter til næste højreside, og <code>\cleartoverso</code> der skifter til næste venstreside; sidestilen for en eventuel »blank« side bliver den til dokumentklassen hørende sidestil til »blanke« sider (<code>cleared</code>).
<code>\enlargethispage</code>	I nødsfald kan man forstørre (eller formindske) en side med en given længde ved hjælp af kommandoen <code>\enlargethispage{<længde>}</code>. Som <code><længde></code> skal man helst bruge et multiplum af <code>\baselineskip</code>. Hvis man for eksempel skriver <code>\enlargethispage{2\baselineskip}</code>, så bliver siden gjort to linjer længere (størrelsen <code>\baselineskip</code> er afstanden mellem underkanten af på hinanden følgende tekstlinjer).
<code>\noprelistbreak</code>	Kommandoen <code>\noprelistbreak</code> kan anbringes umiddelbart før en <code>itemize</code>- eller <code>enumerate</code>- eller <code>description</code>-konstruktion for at forhindre et sideskift på det pågældende sted, <code>\noprelistbreak</code> er defineret i memoir (og dermed i IMFUFA-$\text{\LaTeX}$).

Undertiden kan man få brug for sider uden tekst, men som i øvrigt har samme udformning som de øvrige sider (f.eks. har sidetal). Til det formål har IMFUFA- $\text{\LaTeX}$ kommandoen `\emptypages{<antal>}` der indsætter `<antal>` tomme sider.

`\emptypages`

Linjeskift

Normalt klarer $\text{\LaTeX}$ selv at skifte linje på de rigtige steder, men en sjælden gang må man gribe ind, for eksempel hvis linjen indeholder lange udelelige ord. Følgende kommandoer kan være nyttige at kende:

<code>\par</code>	Kommandoen »blank linje« (eller <code>\par</code>) betyder <i>begynd på nyt afsnit</i>; alt efter omstændighederne viser dette sig enten ved at der begyndes på ny linje som indrykkes, eller ved at der begyndes på ny linje som ikke indrykkes, men som har ekstra afstand til den foregående tekst.
<code>\bpar</code>	Kommandoen <code>\bpar</code> betyder <i>begynd på et nyt afsnit, men lav lidt ekstra afstand</i> (se eks. 2.5 på side 10). Denne kommando findes kun i IMFUFA-$\text{\LaTeX}$.
<code>\\</code>	Kommandoen <code>\\</code> (eller <code>\newline</code>) betyder <i>begynd på ny linje</i>.
<code>\indent</code> <code>\noindent</code>	- Kommandoen <code>\indent</code> laver <i>indrykning</i>. Kommandoen <code>\noindent</code> ophæver en eventuel indrykning. - For at forhindre at $\text{\LaTeX}$ deler et bestemt ord, eller for at forhindre meget store ordmellelrum, kan man putte den »besværlige« tekst ind i en <code>\mbox</code>.

Læs om orddeling på side 55.

4 Ting i kasser

Man kan operere med tekstobjekter hvis indhold er en større eller mindre mængde tekst. I det følgende omtales blandt andet

- `\makebox` og `\mbox` der typisk anvendes når man ønsker at nogle få ord el.lgn. opfattes som ét enkelt tekstobjekt (evt. af en given længde),
- `\framebox` og `\fbox` som er indrammede udgaver af `\makebox` og `\mbox`,
- `\raisebox` der kan hæve eller sænke noget tekst i forhold til grundlinjen,
- `\scalebox`, `\reflectbox` og `\resizebox` der skalerer tekstobjekter,
- `minipage`-konstruktionen der skaber et tekstobjekt hvor noget tekst er sat op som om det var på en selvstændig side af given længde

Derudover findes kommandoen `\boxed` der kan anvendes til indramning af formler, se eks. 6.5 side 61, og float-typen `ibox`, se side 30.

4.1 Kasser og rammer

Simple bokse – `\fbox` og `\framebox`, `\mbox` og `\makebox`

Man kan indramme ting med kommandoen `\fbox{<tekst>}`:

`\fbox`

EKS. 4.1

Her er indrammet tekst, og her er indrammet tekst.

Her er `\fbox{indrammet tekst}`,
og her er indrammet
`\fbox{\phantom{usynlig}}` tekst.

Kommandoen `\mbox{<tekst>}`: virker på samme måde, blot kommer der ingen synlig indramning:

`\mbox`

EKS. 4.2

Her er indrammet tekst, og her er indrammet tekst.

Her er `\mbox{indrammet tekst}`,
og her er indrammet
`\mbox{\phantom{usynlig}}` tekst.

Det ses i øvrigt at kommandoen `\phantom` fremstiller *usynlig* tekst.

`\phantom`

Størrelsen af en `\fbox` eller `\mbox` bestemmes af det der skal indrammes; der findes endvidere de mere generelle kommandoer `\framebox[<bredde>][<pos>]{<tekst>}` og `\makebox[<bredde>][<pos>]{<tekst>}` hvor brugeren selv kan bestemme bredden samt indholdets placering:

`\framebox`

`\makebox`

En 3 cm lang kasse.	En <code>\makebox[3cm]{3 cm lang} kasse.</code>
En 3 cm lang kasse.	En <code>\makebox[3cm][c]{3 cm lang} kasse.</code>
En 3 cm lang kasse.	En <code>\makebox[3cm][s]{3 cm lang} kasse.</code>
En 3 cm lang kasse.	En <code>\makebox[3cm][l]{3 cm lang} kasse.</code>
En 3 cm lang kasse.	En <code>\makebox[3cm][r]{3 cm lang} kasse.</code>
En 3 cm lang kasse.	En <code>\framebox[3cm][c]{3 cm lang} kasse.</code>

Eks. 4.3

Som det ses bestemmer argumentet $\langle pos \rangle$ positionen i kassen; mulige værdier er `c l r s`. Argumentet $\langle bredde \rangle$ bestemmer kassens bredde; her kan man bruge almindelige længder (se side 102), men også de særlige størrelser `\width` (bredden af $\langle tekst \rangle$), `\height` (hvor højt $\langle tekst \rangle$ når op over linjen), `\depth` (hvor langt $\langle tekst \rangle$ når ned under linjen) og `\totalheight` (`\height + \depth`).

`\width`
`\height`
`\depth`
`\totalheight`

50% større	<code>\framebox[1.5\width]{50\% større}</code>
--	--

Eks. 4.4

Hvis man angiver bredden af en `\makebox` eller `\framebox` til 0pt, så bliver det taget helt alvorligt:

Se her, man kan skrive til venstre for tekstens venstrekant.	<code>\makebox[0pt][r]{\ding{43}\,}</code> Se her, man kan skrive til venstre for tekstens venstrekant.
--	---

Eks. 4.5

Vertikale forskydninger – `\raisebox`

`\raisebox` Kommandoen `\raisebox{\l\o\ft}{\langle tekst \rangle}` kan hæve eller sænke en kasse et givet stykke i forhold til linjen; argumentet $\langle l\o\ft \rangle$ er en længde, jf. side 102:

Her er den helt almindelige tekstlinje. Nu hæver vi, og nu sænker vi i forhold til den normale linje. Se hvad der sker.	Her er den helt almindelige tekstlinje. Nu <code>\raisebox{2ex}{\hæver}</code> vi, og nu <code>\raisebox{-2ex}{\sænker}</code> vi i forhold til den normale linje. Se hvad der sker.
---	--

Eks. 4.6

L^AT_EX justerer afstanden sådan at det hævede (sænkede) ikke kolliderer med de nærmeste linjer. Hvis man ikke ønsker dette, kan man angive et eller to valgfrie argumenter der specificerer den ønskede højde og eventuelt også dybde:

Her er den helt almindelige tekstlinje. Nu hæver vi, og nu sænker vi i forhold til den normale linje. Se hvad der sker.	Her er den helt almindelige tekstlinje. Nu <code>\raisebox{2ex}[0ex]{\hæver}</code> vi, og nu <code>\raisebox{-2ex}[0ex]{\sænker}</code> vi i forhold til den normale linje. Se hvad der sker.
---	--

Eks. 4.7

Sorte kasser – `\rule`

Kommandoen `\rule[⟨løft⟩]{⟨bredde⟩}{⟨højde⟩}` fremstiller et udfyldt rektangel med en given bredde og højde; rektanglet kan hæves/sænkes: `\rule`

Eks. 4.8

En bjælke:  .	En bjælke: <code>\rule{5mm}{1mm}.</code>
En løftet bjælke:  .	En løftet bjælke: <code>\rule[-2mm]{5mm}{1mm}.</code>

Som bredde eller højde kan man godt angive 0, eller rettere 0 af en eller anden længdeenhed. Så får man et tekstobjekt som i sig selv er usynligt, men som godt kan have en synlig virkning, for eksempel kan man få en kasse til at blive større:

Eks. 4.9

 	<code>\fbox{X} \fbox{X\rule{0ex}{3ex}}</code>
---	---

4.2 Rotation og skalering

Alle tekstobjekter kan roteres og/eller skales ved hjælp af nedenfor omtalte kommandoer, der er definerede i `graphicx`-pakken.

Kommandoen `\scalebox{⟨h-skala⟩}[⟨v-skala⟩]{⟨tekstobjekt⟩}` skalerer horisontalt og vertikalt: `\scalebox`

Eks. 4.10

Køge Køge	<code>Køge \scalebox{1.5}{Køge}</code>
Køge Køge	<code>Køge \scalebox{1.5}[1]{Køge}</code>

Negative værdier af skaleringsparametrene giver spejlinger. Hvis man ikke ønsker at skalere, »kun« at spejlvende, er `\reflectbox{⟨tekstobjekt⟩}` også en mulighed: `\reflectbox`

Eks. 4.11

Køge 	<code>Køge \scalebox{-1.5}[1]{Køge}</code>
Køge 	<code>Køge \reflectbox{Køge}</code>

Kommandoen `\resizebox*{⟨bredde⟩}{⟨højde⟩}{⟨tekstobjekt⟩}` skalerer til en bestemt størrelse; det gør `\resizebox{⟨bredde⟩}{⟨højde⟩}{⟨tekstobjekt⟩}` også, og dog ikke: `\resizebox*`
`\resizebox`

Eks. 4.12

	<code>\resizebox*{1cm}{1cm}{Køge}</code>
	<code>\resizebox{1cm}{1cm}{Køge}</code>

I den stjernede udgave bliver den samlede højde lig med den angivne $\langle højde \rangle$ (i eksemplet bliver afstanden fra underkanten af g til overkanten af K lig 1 cm), i den ustjernede udgave bliver afstanden fra basislinjen til overkanten af teksten lig med $\langle højde \rangle$ (i eksemplet bliver højden af K lig 1 cm).

Som $\langle \text{bredde} \rangle$ og $\langle \text{højde} \rangle$ kan man bruge alle slags længder, men man kan også bruge $\backslash\text{depth}$, $\backslash\text{height}$, $\backslash\text{totalheight}$ og $\backslash\text{width}$, således at man eksemplevis kan skalere til en given bredde uden at ændre højden. Endelig kan man som den ene af $\langle \text{bredde} \rangle$ og $\langle \text{højde} \rangle$ skrive $!$; resultatet bliver da at skaleringen bibeholder forholdet mellem tekstobjektets bredde og højde.

Køge Køge Køge

Køge

$\backslash\text{resizebox}*\{1\text{cm}\}\{\backslash\text{totalheight}\}\{\text{Køge}\}$

$\backslash\text{resizebox}*\{1\text{cm}\}\{!\}\{\text{Køge}\}$

EKS. 4.13

$\backslash\text{rotatebox}$

Kommandoen $\backslash\text{rotatebox}[\langle \text{kevvalist} \rangle]\{\langle \text{vinkel} \rangle\}\{\langle \text{tekstobjekt} \rangle\}$ roterer $\langle \text{tekstobjekt} \rangle$ en given vinkel (der regnes i grader). I $\langle \text{kevvalist} \rangle$ kan man bl.a. skrive en origin= -specifikation på samme måde som i forbindelse med $\backslash\text{includegraphics}$, se side 42.

Køge Køge

Køge $\backslash\text{rotatebox}[\text{origin=c}]\{10\}\{\text{Køge}\}$

EKS. 4.14

Læs mere om de ovenfor omtalte kommandoer i Mittelbach m.fl. (2004, afsnit 10.3).

4.3 En side på siden: minipage-konstruktioner

Man kan undertiden have brug for at operere med tekstobjekter hvis indhold er sat op som om det var en (lille) selvstændig side, men i øvrigt efter samme regler som der gælder for den store tekstside. I sådanne situationer kan en `minipage` være nyttig.

Her er et citat fra Robert Musils roman *Manden uden egenskaber*, bind 1:

Egentlig var det kun matematikerne selv og deres disciple, naturforskerne, der levede i uvidenhed om disse farer, de fornemmede lige så lidt af alt dette i deres sjæl som cykelryttere, der flittigt tramped løs og ikke lægger mærke til andet i verden end den forankorende baghjul. Om Ulrich kunne man derimod med sikkerhed sige det ene, at han elskede matematikken på grund af de mennesker, der ikke kunne udstå den. Han var mindre videnskabeligt end menneskeligt forelsket i videnskaben. Han så, at den i alle spørgsmål, hvor den betragter sig som kompetent, tænker anderledes end almindelige mennesker. Hvis man i stedet for videnskabelige anskuelser satte livsanskuelse, i stedet for hypotese forsøg og i stedet for sandhed handling, var der ingen betydelig naturforsker eller matematiker, hvis livsværk ikke langt overgik historiens største bedrifter.

Her er et citat fra Robert Musils roman $\backslash\text{textit}\{\text{Manden uden egenskaber}\}$, bind 1:

```
 $\backslash\text{begin}\{\text{center}\}$ 
 $\backslash\text{begin}\{\text{minipage}\}\{0.87\backslash\text{textwidth}\}$ 
 $\backslash\text{scriptsize}$ 
Egentlig var det kun
matematikerne selv og deres
disciple, naturforskerne, der
levede i uvidenhed om disse farer
...
hvis livsværk ikke langt overgik
historiens største bedrifter.
 $\backslash\text{end}\{\text{minipage}\}$ 
 $\backslash\text{end}\{\text{center}\}$ 
```

EKS. 4.15

`minipage`

Syntaksen er $\backslash\text{begin}\{\text{minipage}\}[\langle \text{pos} \rangle]\{\langle \text{bredde} \rangle\}\langle \text{indhold} \rangle\backslash\text{end}\{\text{minipage}\}$, og resultatet er at $\langle \text{indhold} \rangle$ bliver sat på en mini-side af den angivne bredde; højden bestemmes af hvor meget indholdet fylder. Den $\langle \text{bredde} \rangle$ der skal angives, skal være en længde (jf. side 102). I $\langle \text{indhold} \rangle$ kan man skrive næsten alle slags L^AT_EX-kommandoer og

-konstruktioner, dog ikke dem der frembringer ‘floats’ (såsom figurer, tabeller og marginalnoter). Det valgfri argument *(pos)* bestemmer positionen af **minipage**-tekstobjektet i forhold til den omgivende tekst: mulige værdier er **c** (centreret), **t** (top) og **b** (bund); default er **c**.

IMFUFUA- $\text{\LaTeX}$ har en **fmpage**-konstruktion (‘framed minipage’) der fremstiller en minipage med en ramme om; syntaksen er som for **minipage**. **fmpage**

Fodnoter markeres anderledes i en minipage.^a

^a Med bogstaver.

```
\begin{fmpage}{0.8\textwidth}
  Fodnoter markeres anderledes i en
  minipage.\footnote{Med bogstaver.}
\end{fmpage}
```

Eks. 4.16

4.4 Eksterne billeder; **graphicx**-pakken

$\text{\LaTeX}$ er udviklet til behandling af tekst og matematiske formler, ikke billeder. Med kommandoen **\includegraphics** (defineret i pakken **graphicx**) er det dog meget enkelt at indsætte »eksterne« billeder. Her ses hvordan man kan indsætte et billede af den italienske matematiker Maria Gaëtana Agnesi; billedet findes i filen **agnesi.jpg**.¹



M. Agnesi, 1718-99

```
\centering
\includegraphics[width=2cm]{agnesi}\\
M. Agnesi, 1718-99
```

Eks. 4.17

Efter eks. 4.17 at dømme ser det jo ud til at være nemt nok at inkludere billeder, men der kan alligevel opstå forskellige små-vanskeligheder, f.eks. kan man ikke benytte vilkårlige billedformater, og de mulige formater afhænger af hvordan man kompilerer sin tex-fil. Her er en oversigt der dækker de almindeligste situationer:

metode	kompilering/outputformat	mulige billedformater
1	tex $\xrightarrow{\text{pdf\textit{latex}}}$ pdf	png, pdf, jpg, mps
2	tex $\xrightarrow{\text{latex}}$ dvi $\xrightarrow{\text{dvips}}$ ps	eps, ps

Der findes programmer der kan konvertere fra et billedformat til et andet, her skal blot nævnes **epstopdf** der omdanner en eps-fil til en pdf-fil.

Nærværende tekst er fremstillet efter »model 1« der tillader inklusion af jpeg-billeder, og derfor kunne billedet af Maria Agnesi uden videre inkluderes. Som det fremgår af eks.

¹ hentet fra <http://www-history.mcs.st-andrews.ac.uk/history/Mathematicians/Agnesi.html>.

4.17, er det ikke nødvendigt at medtage filens »efternavn« `jpg` og skrive `agnesi.jpg`; hvis man ikke angiver et efternavn, så vil `\includegraphics`-funktionen forsøge sig med et af de mulige efternavne.

Valgmuligheder i `graphicx`-pakken

Man indlæser `graphicx`-pakken med kommandoen `\usepackage[⟨valg⟩]{graphicx}`. Her er `⟨valg⟩` ofte den tomme liste, altså »ingenting«, men der er andre muligheder:

- `draft` bevirker at ingen billeder indsættes, i stedet kommer der et indrammet blankt felt af samme størrelse som det billede der skulle have været der. Billedfilens navn bliver skrevet i det indrammede felt.
- `final` er det modsatte af `draft`.
- `hiderotate` bevirker at roterede billeder ikke søges vist.
- `hidescale` bevirker at skalerede billeder ikke søges vist.

Kommandoen `\includegraphics`

`\includegraphics`

Kommandoen til inklusion af billedfiler er `\includegraphics[⟨keyvallist⟩]{⟨filnavn⟩}`. Her er `⟨filnavn⟩` navnet på den billedfil der ønskes inkluderet, og `⟨keyvallist⟩` er en liste hvis elementer (der adskilles af komma) er af formen `key=value` eller `key`; vi nævner enkelte eksempler på `key=`:

`scale=⟨faktor⟩` skalerer billedet med faktoren `⟨faktor⟩`, se eks. 4.19.

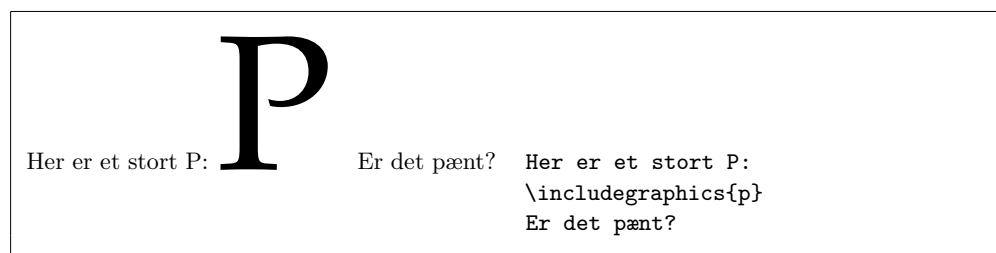
`width=⟨længde⟩` skalerer billedet så det får bredde `⟨længde⟩`, se eks. 4.20.

`height=⟨længde⟩` skalerer billedet så det får højde `⟨længde⟩`.

`angle=⟨antalgrader⟩` roterer billedet vinklen `⟨antalgrader⟩`, se eks. 4.21.

`origin=⟨omdrejningspunkt⟩` Fikspunktet for rotationen er normalt nederste venstre hjørne, men det kan ændres med `origin=⟨omdrejningspunkt⟩`. hvor `⟨omdrejningspunkt⟩` kan være et eller to af `c l r t b B` (for hhv. center, left, right, top, bottom, Baseline), se eks. 4.22 og 4.23.

Til demonstration af grafikhåndtering benytter vi et billede der viser det store P i Palatino-skriften; billedet findes i filerne `p.pdf` (til brug ifm. `pdflatex`) og `p.eps` (til brug ifm. `latex`).



Eks. 4.18

Figuren placeres så dens nederste kant står på linje med den almindelige tekst. Man kan skalere figuren op eller ned:

EKS. 4.19

Skaleret til halv størrelse:		Skaleret til halv størrelse: <code>\includegraphics[scale=0.5]{p}</code>
------------------------------	---	---

og man kan give figuren en bestemt størrelse:

EKS. 4.20

Med en given størrelse:		Med en given størrelse: <code>\includegraphics[width=1cm]{p}</code>
-------------------------	---	--

Som det ses, regulerer `width` figurens udstrækning i begge dimensioner. Man kan sætte `width` til en vilkårlig af L^AT_EX kendt længde, se side 102.

Man kan rotere figuren et ønsket antal grader:

EKS. 4.21

Skalér til halv størrelse og rotér 90 grader i negativ omløbsretning:		Skalér til halv størrelse og rotér 90 grader i negativ omløbsretning: <code>\includegraphics[scale=0.5,angle=-90]{p}</code>
---	---	--

Fikspunktet for rotationen er normalt nederste venstre hjørne. Det kan ændres med `origin=<omdrejningspunkt>`:

EKS. 4.22

Rotation om centrum:		Rotation om centrum: <code>\includegraphics[scale=0.5,origin=c,angle=-90]{p}</code> .
----------------------	---	--

EKS. 4.23

Rotation om centrum på basislinjen:		Rotation om centrum på basislinjen: <code>\includegraphics[scale=0.5,origin=cB,angle=-90]{p}</code> .
-------------------------------------	---	--

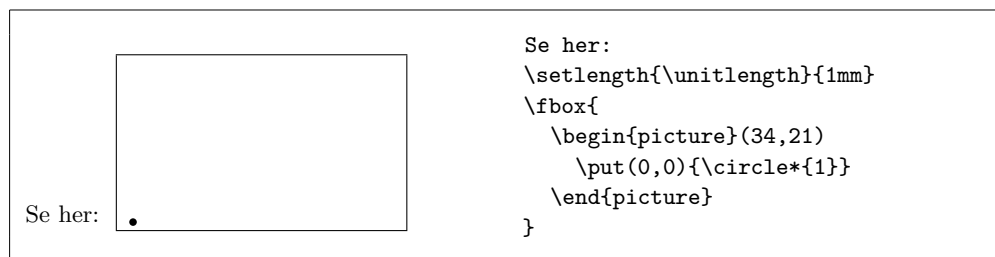
- I listen over manipulationer (rotationer, størrelsesændringer o.lgn.) der skal foretages på grafikfilen, spiller *rækkefølgen* en rolle (fordi operationerne udføres i den rækkefølge de er skrevet), eksempelvis giver `width=3cm,rotate=45` ikke det samme som `rotate=45,width=3cm`.
- Resultatet af `\includegraphics{...}` er et helt almindeligt L^AT_EX-objekt der kan placeres hvor man ønsker det, f.eks. mellem `\begin{center}` `\end{center}` (hvis det skal centreres), som argument til `\fbox` (hvis det skal indrammes), eller i en `figure`- eller `table`-konstruktion.

Der er som nævnt muligt at inkludere en række forskellige slags billedfiler, eventuelt efter passende konverteringer. Hvordan man fremstiller de konkrete billeder, såsom geometriske figurer, diagrammer, grafer og noder, vil ikke blive behandlet i nærværende fremstilling, men se f.eks. Goossens m.fl. (1997). I matematik- og fysiksammenhænge er METAPOST ofte et velegnet redskab til fremstilling af grafer o.lgn. Læs om METAPOST i Goossens m.fl. (1997, afsnit 3.2 og 3.3) eller (mere ajourført) på TUGs METAPOST-side <http://www.tug.org/metapost.html>.

4.5 Grafik med picture

Her er en kort omtale af standard- $\text{\LaTeX}$'s `picture`-konstruktion der kan bruges til simple tegninger. Bemærk at den syntaks der gælder i en `picture`-konstruktion, ikke stemmer helt overens med den almindelige $\text{\LaTeX}$ -syntaks, så der er gode fejlmuligheder.

Vi vil nu lave et billed-objekt som er 34 mm bredt og 21 mm højt, og hvor der er placeret en sort klat i det punkt som har koordinaterne $(0,0)$ i det lokale koordinatsystem; der er sat en ramme uden om billed-objektet (ved hjælp af `\fbox`) så man kan se hvor meget plads $\text{\LaTeX}$ har afsat til det:



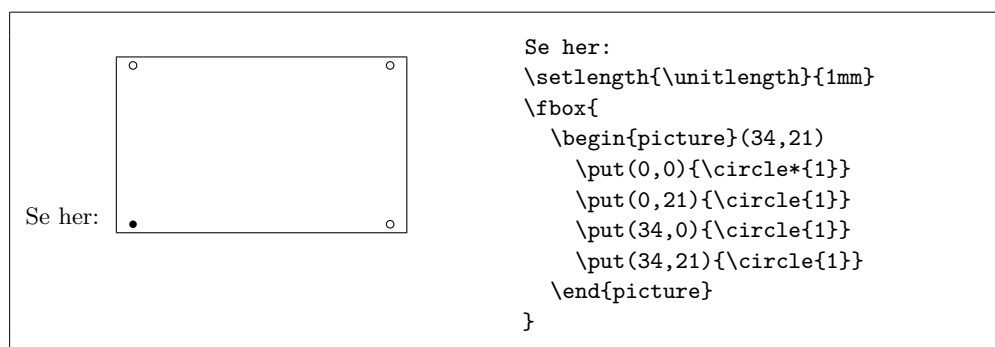
Eks. 4.24

 $\backslash\text{unitlength}$

Der sker følgende: Før man begynder på billedet, fastsættes den `\unitlength` der skal benyttes; her er den sat til 1 mm, men man kan sætte den til en hvilken som helst længde (standardværdien er 1 pt). Derefter oprettes et billed-objekt hvis bredde og højde er hhv. 34 og 21 længdeenheder. Endelig anbringes en udfyldt cirkel med radius 1 i punktet $(0,0)$. Bemærk brugen af *runde* parenteser.

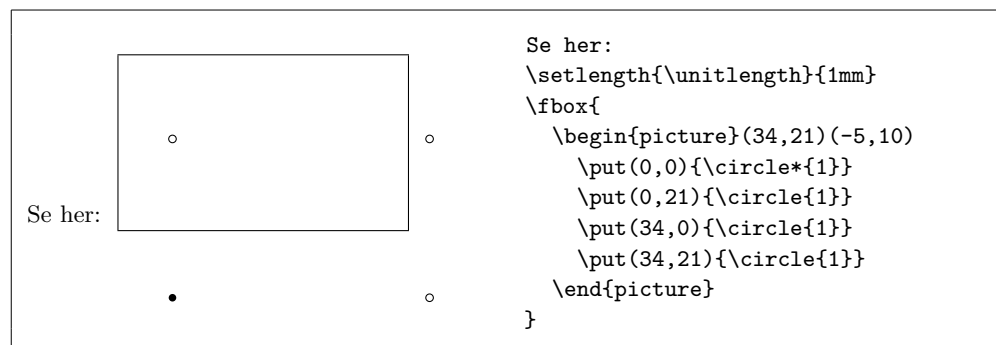
 $\backslash\text{put}$

Kommandoen `\put` placerer et del-objekt på et givet sted i det lokale koordinatsystem. Lad os placere ikke-udfyldte cirkler i de tre andre hjørner:



Eks. 4.25

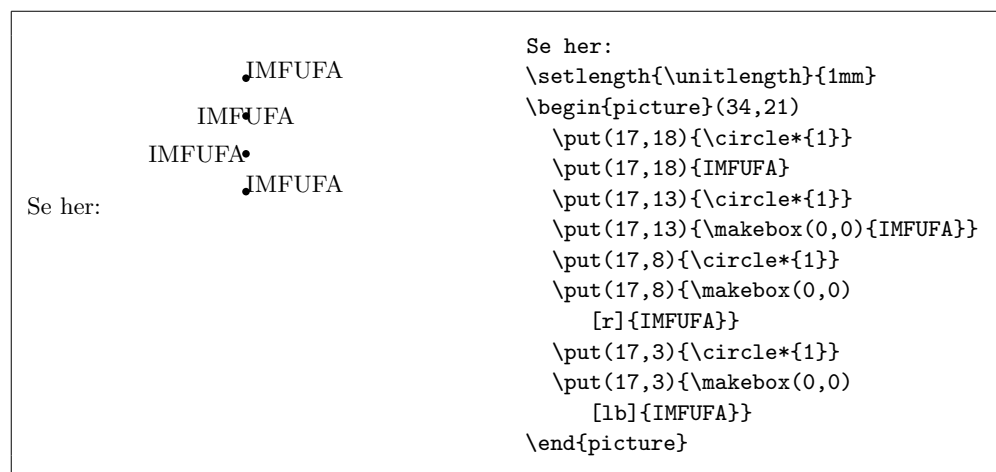
Efter `\begin{picture}` skal der stå et talpar, men der kan godt stå to:



EKS. 4.26

Det andet talpar forskyder billedets indhold; talparret (x, y) bevirker en forskydning på $(-x, -y)$. Bemærk at det netop kun er billedets *indhold* der forskydes, selve billedobjektet ændrer ikke placering eller størrelse, hvilket ses af indramningerne. – I de følgende eksempler undlader vi indramningen.

Næste eksempel viser nogle måder til placering af tekst i en `picture`-konstruktion. De små udfyldte cirkler viser de punkter som `\put`-kommandoerne refererer til.



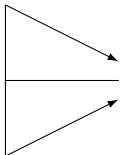
EKS. 4.27

Det talpar der står efter `\makebox`, er størrelsen af den fremstillede box. Boxen placeres med nederste venstre hjørne i det punkt som angives i `\put`, medmindre der er angivet en plads-specifikator i `[ ]`.

Foruden `\makebox` findes også kommandoerne `\framebox` og `\dashbox` der virker på tilsvarende måde, blot omgiver de deres indhold med en fuldt optrukket hhv. stiplede ramme.

Man kan tegne linjestykker og vektorer (linjestykker med en pil i den ene ende) som har en rational hældningskoefficient p/q eller som er lodrette. Syntaksen er `\line(p,q){\langle længde \rangle}` eller `\vector(p,q){\langle længde \rangle}` hvor $\langle længde \rangle$ er længden af linjens projektion på y -aksen hvis $p = 0$ og linjen dermed er lodret, eller længden af projektionen på x -aksen. I alle tilfælde skal $\langle længde \rangle$ være større eller lig 0. Linjestykket afsættes ud fra det punkt der specificeres i `\put`.

Se her:



Se her:

```

\setlength{\unitlength}{1mm}
\begin{picture}(34,21)
  \put(10, 0){\vector(2,1){15}}
  \put(10,10){\line(1,0){15}}
  \put(10,20){\vector(2,-1){15}}
  \put(10, 0){\line(0,1){20}}
\end{picture}

```

Eks. 4.28

På grund af interne begrænsninger i L^AT_EX kan man kun benytte temmelig små værdier af p og q . Hvis man inkluderer pakken `pict2e`, ophæves disse begrænsninger, og man får i det hele taget flere tilpasningsmuligheder, se Gäßlein og Niepraschk (2004).

Se her:



Se her:

```

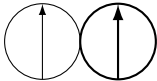
\setlength{\unitlength}{1mm}
\begin{picture}(34,21)
  \put(10, 0){\circle*{2}}
  \put(20,17.5){\circle{2}}
  \put(10,0){\line(20,35){10}}
\end{picture}

```

Eks. 4.29

Linjer kan gøres tykkere med kommandoen `\thicklines`; kommandoen `\thinlines` giver normal tykkelse.

Se her:



Se her:

```

\setlength{\unitlength}{1mm}
\begin{picture}(34,21)
  \put(10,10){\circle{10}}
  \put(10, 5){\vector(0,1){10}}
  \thicklines
  \put(20,10){\circle{10}}
  \put(20, 5){\vector(0,1){10}}
\end{picture}

```

Eks. 4.30

Foruden cirkler kan man tegne ovaler:

Se her:



Se her:

```

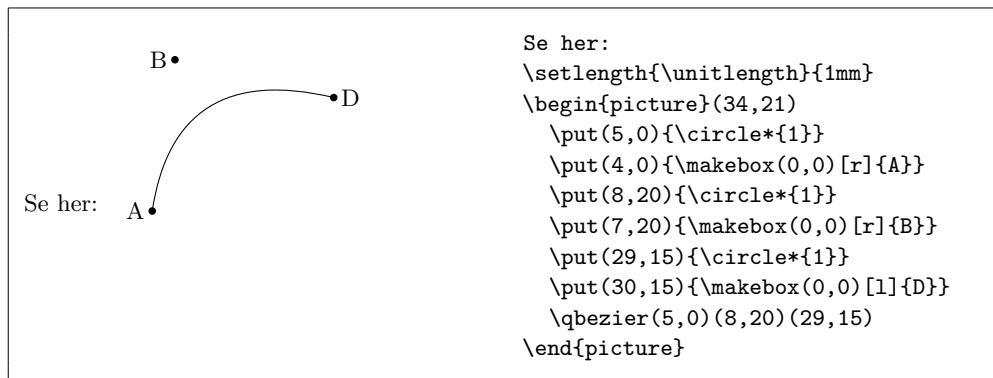
\setlength{\unitlength}{1mm}
\begin{picture}(34,21)
  \put(15,5){\oval(10,4)}
  \put(15,15){\oval(10,4)[tr]}
\end{picture}

```

Eks. 4.31

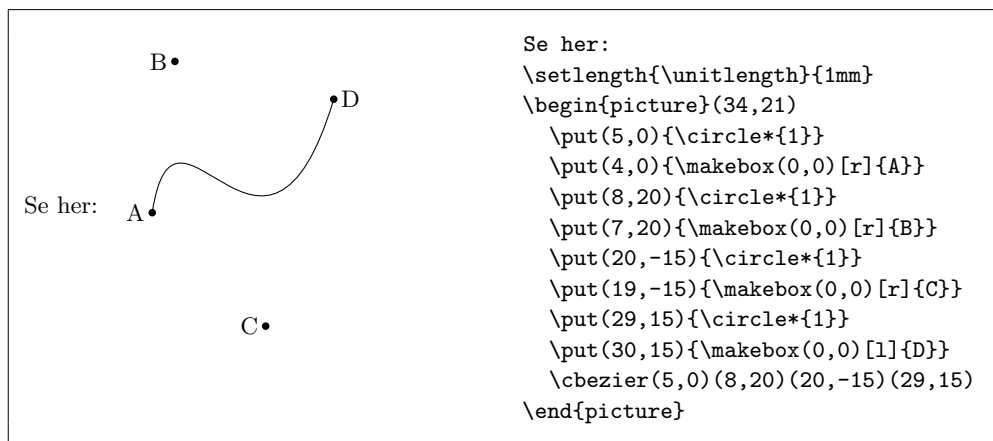
Ligesom der kun er et begrænset udvalg af hældningskoefficienter for linjer tegnet med `\line` og `\vector`, er der også kun et begrænset antal diametre for cirkler og ovaler, medmindre man benytter `pict2e`-pakken.

Kommandoen `\qbezier` tegner en kvadratisk Bézier-kurve; her forbindes punkterne A og D med en Bézier-kurve med B som det tredje Bézier-punkt:



Eks. 4.32

Kommandoen `\cbezier` tegner en kubisk Bézier-kurve. Her forbindes A og D med en Bézier-kurve med B og C som de to Bézier-punkter:



Eks. 4.33

5 Æstetiske betragtninger

I dette kapitel fortælles lidt om hvordan brugeren kan ændre forskellige af de parametre der bestemmer hvilken størrelse og art skrift der benyttes, og der er nogle betragtninger over hvordan man kan opnå et pænt skriftbillede i det færdige resultat.

5.1 Valg af skriftgrad

L^AT_EX vælger selv en skriftgrad (størrelse) der passer til omstændighederne, for eksempel sættes fodnoter med en mindre skrift end grundteksten. Grundteksten sættes normalt i 10 punkt (eller rettere 10 pt, jf. side 102). I dokumentklasseerklæringen kan man vælge en anden grad, se side 17.

Inde i teksten kan man ændre den *relative* skriftgrad med kommandoerne `\miniscule`, `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge`, `\Huge` og `\HUGE`. Kommandoernes rækkevidde afgrænses med `{ ... }` eller med `\begin{noget} ... \end{noget}`. Eksempel:

EKS. 5.1

Et lille bitte nøk.

Et `{\tiny lille bitte}` nøk.

- Linjeafstanden afhænger af den valgte skriftgrad, og L^AT_EX sætter et helt afsnit ad gangen med den linjeafstand der er gældende på det sted hvor afsnittet slutter. Derfor skal man som regel sætte et »afsnit slut«-mærke *inden* man afslutter den forstørrede eller formindskede tekst; »afsnit slut«-mærket er som altid en blank linje eller kommandoen `\par` (eller `\bpar`). Se eks. 5.2 og 5.3 på næste side.

5.2 Valg af skriftsnit

Tekstens brødskrift, dvs. den skrift som den altovervejende del af dokumentet sættes med, bør være en neutral og letlæselig skrift, normalt fra den skriftfamilie der kaldes antikva (engelsk: Roman). L^AT_EX benytter automatisk den antikva-skrift der ses som brødskrift i nærværende hæfte, og som hedder European Computer Modern Roman (eller mere præcist er det den Type 1-udgave heraf der hedder Latin Modern).

- *For at fremhæve enkelte tekstdele kan man tage særlige virkemidler i brug, som for eksempel at vælge en kursiv-skrift. Kursiv-skriftens bogstaver og tegn adskiller sig væsentligt fra antikva-skriftens, se f.eks. på bogstaver som a f g og æ.*
- *En anden mulighed er en hældende antikva-skrift.*

Her vises hvordan man <i>ikke</i> skal gøre. Afsnittet med lille skrift bliver sat med normalskriftens linjeafstand.	<code>{\tiny Her vises hvordan man \textit{ikke} skal gøre. Afsnittet med lille skrift bliver sat med normalskriftens linjeafstand.}</code>
Her er normal skriftstørrelse.	Her er normal skriftstørrelse.

Eks. 5.2

Her afsluttes afsnittet med lille skrift med et »nyt afsnit«-mærke, så derfor bliver afsnittet sat med den rigtige linjeafstand.	<code>{\tiny Her afsluttes afsnittet med lille skrift med et \enquote{nyt afsnit}-mærke, så derfor bliver afsnittet sat med den rigtige linjeafstand.\par}</code>
Her er normal skriftstørrelse.	Her er normal skriftstørrelse.

Eks. 5.3

- MAN KAN OGSÅ BENYTTE KAPITÆLSKRIFT; DEN ER KARAKTERISTISK VED AT DE SMÅ BOGSTAVER ER MAGEN TIL DE STORE, BLOT MINDRE.
- Til *særlige formål* kan skrivemaskineskrift være velegnet.
- En helt anden skriftfamilie er blokskrift (grotesk), der ofte kaldes sans serif fordi blokskrift-bogstaverne i modsætning til antikva-bogstaverne mangler serifferne (de små tværstreger for enden af bogstaverne).

Hvis man uafadeligt skifter skriftsnit, opnår man at teksten bliver temmelig ulæselig.

L^AT_EX har et meget elegant system til skift mellem de forskellige skriftformer. Ideen er at en skrift kan beskrives ved fire attributter hvis betegnelser (på engelsk) er *⟨family⟩*, *⟨series⟩*, *⟨shape⟩* og *⟨size⟩*, og som kan vælges uafhængigt af hverandre. Attributterne kan have følgende værdier:

⟨family⟩ kan være

- rm** Antikva (engelsk: Roman). – Dette er standard-værdien.
- sf** Blokskrift (eller Sans serif)
- tt** Skrivemaskineskrift (engelsk: Typewriter Type)

⟨series⟩ kan være

- md** Medium. – Dette er standard-værdien.
- bf** **Fed** (engelsk: Boldface)

⟨shape⟩ kan være

- up** Opret (engelsk: Upright). – Dette er standard-værdien.
- it** *Kursiv* (engelsk: Italic)
- sl** *Hældende* (engelsk: Slanted)
- sc** KAPITÆLER (engelsk: Small Caps)

⟨size⟩ kan være en af de 12 værdier der nævnt på side 49.

Tabel 5.1 Skriftvalg: kommandoer og erklæringer

kommando	erklæring	beskrivelse
<code>\textrm{...}</code>	<code>\rmfamily</code>	Antikva
<code>\textsf{...}</code>	<code>\sffamily</code>	Blokskrift
<code>\texttt{...}</code>	<code>\ttfamily</code>	Skrivemaskine
<code>\textmd{...}</code>	<code>\mdseries</code>	Medium
<code>\textbf{...}</code>	<code>\bfseries</code>	Fed
<code>\textup{...}</code>	<code>\upshape</code>	Opret
<code>\textit{...}</code>	<code>\itshape</code>	<i>Kursiv</i>
<code>\textsl{...}</code>	<code>\slshape</code>	<i>Hældende</i>
<code>\textsc{...}</code>	<code>\scshape</code>	KAPITÆLER
<code>\textnormal{...}</code>	<code>\normalfont</code>	Brødskriften

Man kan vælge $\langle family \rangle$, $\langle series \rangle$ og $\langle shape \rangle$ på to måder, enten med en *kommando* eller med en *erklæring*; tabel 5.1 viser hvilke der er til rådighed.

1. En font-*kommando* er af formen `\textxx{⟨tekst⟩}` hvor *xx* er en af en de nævnte totegnskoder; eksempel: `\textxx`

EKS. 5.4

Dette er **fed** skrift, og dette er *kursiv*. Dette er `\textbf{fed}` skrift,
og dette er `\textit{kursiv}`.

Kommandoerne kan kombineres:

EKS. 5.5

Hældende blokskrift. `\textsf{\textsl{Hældende}}`
`blokskrift.`

Der er ingen tvivl om hvad der er rækkevidden for en fontkommando: det er den tekst der står som argument til kommandoen `\textxx`.

2. En font-*erklæring* er af formen `\uufamily` hvor *uu* er totegnskoden for en $\langle family \rangle$, `\uufamily`
eller `\vvseries` hvor *vv* er totegnskoden for en $\langle series \rangle$, eller `\yyshape` hvor *yy* `\vvseries`
er totegnskoden for en $\langle shape \rangle$. Eksempel:

EKS. 5.6

Normal skrift. *Kursiv*. Atter normal. Normal skrift. `\itshape` *Kursiv*.
`\normalfont` Atter normal.

I eksemplet har vi benyttet kommandoen `\normalfont` til at vende tilbage til den normale skrift. Hvis der ikke kommer noget der ophæver en fonterklæring, så er den gældende indtil udgangen af den aktuelle »gruppe«. Den aktuelle »gruppe« er den mindste (input)tekst del som er omsluttet enten af matchende `{ }` eller af matchende `\begin{⟨noget⟩} \end{⟨noget⟩}`. Eksempel:

EKS. 5.7

Normal skrift. *Kursiv*. Atter normal. Normal skrift.
`\itshape` *Kursiv.* Atter normal.

Enhver af de $3 \times 2 \times 4 = 24$ kombinationer af $\langle family \rangle$, $\langle series \rangle$ og $\langle shape \rangle$ er tilladt i hver af de 12 størrelser, men dermed er ikke sagt at alle kombinationer findes i en given L^AT_EX-installation; nogle skrifter hører ikke med i standardopsætningen af L^AT_EX, og nogle skrifter eksisterer simpelthen ikke – der er f.eks. ingen kursiv blokskrift. Billedet ændrer sig også hvis man skifter fra Computer Modern skriften til f.eks. Palatino eller Times. Heldigvis er det sådan at når man beder om en ikke-eksisterende skrift, får man automatisk en erstatning – f.eks. bliver kursiv blokskrift automatisk erstattet af hældende blokskrift (og der kommer en bemærkning i log-filen).

»Gammeldags« tal

`\oldstylenums`

De så moderne »gammeldags« tal, eller rettere mediævaltal eller hængende tal, fås med kommandoen `\oldstylenums`; den udgave af kommandoen som findes i IMFUFA-L^AT_EX, respekterer de forskellige skriftvalgskommandoer:

J.S. Bach levede fra 1685 til 1750.	<code>\bfseries J.S. Bach levede fra \oldstylenums{1685} til \oldstylenums{1750}.</code>
--	--

Eks. 5.8

Hvis man bruger Palatinoskriften (via valgmuligheden `palatino`, jf. side 18), så bliver teksttallene automatisk mediævaltal, se figur 2.1(b) på side 19.

Fremhævet tekst

Hvis man ønsker at fremhæve enkelte ord, gøres det normalt med *kursiv*, altså med `\textit`-kommandoen. (Derimod bør man absolut ikke fremhæve ord ved understregning, og normalt heller ikke ved at skrive dem med fed skrift.)

`\emph`

Det kan nævnes at L^AT_EX også har en kommando `\emph{\langle tekst \rangle}` der gør det samme som `\textit`, og så alligvel ikke:

Thi <i>det var en ægte prinsesse.</i>	Thi <code>\emph{det var en \emph{ægte} prinsesse}.</code>
---------------------------------------	---

Eks. 5.9

Kommandoen `\emph` skifter mellem kursiv og opret skrift. Her er endnu et eksempel:

<i>Det går stadig fremad</i>	<code>\itshape Det går \emph{stadig} fremad</code>
------------------------------	--

Eks. 5.10

Zapf Dingbats

Foruden tekstfonte og fonte med matematiske symboler findes der mange fonte med specielle tegn. Her nøjes vi med at nævne et enkelt designmæssigt vellykket eksempel, nemlig Hermann Zapfs *dingbats* der ses i tabel 5.2. Med `pifont`-pakken bliver dingbats-symbolerne tilgængelige via kommandoen `\ding{\langle ascii-nr \rangle}`.

`\ding`

Et Aldus-blad: ☛	Et Aldus-blad: <code>{\large\ding{166}}</code>
------------------	--

Eks. 5.11

Tabel 5.2 Zapf Dingbats-symbolerne med tilhørende ascii-værdier.

	33 ✂	34 ✂	35 ✂	36 ✂	37 ☎	38 ☎	39 ☎
40 ✈	41 ☒	42 ☞	43 ☞	44 ☞	45 ☞	46 ☞	47 ☞
48 ☞	49 ☞	50 ☞	51 ✓	52 ✓	53 ✕	54 ✕	55 ✕
56 ✕	57 ✚	58 ✚	59 ✚	60 ✚	61 †	62 †	63 †
64 ☒	65 ☆	66 ✚	67 ✚	68 ♣	69 ✚	70 ♦	71 ♦
72 ★	73 ☆	74 ☼	75 ☆	76 ☆	77 ☆	78 ☆	79 ☆
80 ☆	81 ✱	82 ✱	83 ✱	84 ✱	85 ✱	86 ✱	87 ✱
88 ✱	89 ✱	90 ✱	91 ✱	92 ✱	93 ✱	94 ✱	95 ✱
96 ✱	97 ✱	98 ✱	99 ✱	100 ✱	101 ✱	102 ✱	103 ✱
104 ✱	105 ✱	106 ✱	107 ✱	108 ●	109 ○	110 ■	111 □
112 □	113 □	114 □	115 ▲	116 ▼	117 ◆	118 ◆	119 ●
120	121	122	123 ‘	124 ‘	125 “	126 ”	
	161 ☺	162 ☺	163 ☺	164 ♥	165 ☺	166 ☺	167 ☺
168 ♣	169 ♦	170 ♥	171 ♠	172 ①	173 ②	174 ③	175 ④
176 ⑤	177 ⑥	178 ⑦	179 ⑧	180 ⑨	181 ⑩	182 ①	183 ②
184 ③	185 ④	186 ⑤	187 ⑥	188 ⑦	189 ⑧	190 ⑨	191 ⑩
192 ①	193 ②	194 ③	195 ④	196 ⑤	197 ⑥	198 ⑦	199 ⑧
200 ⑨	201 ⑩	202 ①	203 ②	204 ③	205 ④	206 ⑤	207 ⑥
208 ⑦	209 ⑧	210 ⑨	211 ⑩	212 ➔	213 ➔	214 ↔	215 ↕
216 ↘	217 ➔	218 ↗	219 ➔	220 ➔	221 ➔	222 ➔	223 ➔
224 ➔	225 ➔	226 ➔	227 ➔	228 ➔	229 ➔	230 ➔	231 ➔
232 ➔	233 ➔	234 ➔	235 ➔	236 ➔	237 ➔	238 ➔	239 ➔
	241 ➔	242 ➔	243 ➔	244 ➔	245 ➔	246 ➔	247 ➔
248 ➔	249 ➔	250 ➔	251 ➔	252 ➔	253 ➔	254 ➔	

Pakken pifont definerer en dingautolist-konstruktion som er en pendant til `enumerate`; `dingautolist` man skal angive et argument til `dingautolist`, nemlig start-tegnets nummer:

EKS. 5.12

① pro primo	<code>\begin{dingautolist}{192}</code>
② pro secundo	<code>\item pro primo</code>
③ pro tertio	<code>\item pro secundo</code>
	<code>\item pro tertio</code>
	<code>\end{dingautolist}</code>

Der defineres også bl.a. `\dingfill{<ascii-nr>}` der udfylder mellemrum med et givet tegn, og `\dingline{<ascii-nr>}` der udfylder en hel linje (bortset fra en bid i hver side): `\dingfill`
`\dingline`

EKS. 5.13

✂ ✂ ✂ ✂	Klip her	✂ ✂ ✂ ✂	<code>\dingfill{34}Klip her\dingfill{34}</code>
♥ ♥ ♥ ♥ ♥ ♥			<code>\dingline{164}</code>

5.3 Læselighed

En af grundene til at skrive noget er at man har en forventning om at der vil være nogen der vil læse det man skriver, og derfor bør tekstens udseende tilrettelægges på en sådan måde at den bliver let at læse – det samme gælder naturligvis indholdet, men det er en anden historie. Der er mange parametre der spiller ind på læseligheden: bogstaverne må hverken være for store eller for små, linjerne hverken for korte eller for lange, linjeafstanden hverken for stor eller for lille, og skriften skal være klar og tydelig. Der er en lang række æstetiske regler indbygget i L^AT_EX, og for det meste står man sig ved at lade L^AT_EX administrere disse regler på egen hånd; i visse situationer kan brugeren dog måske gerne ville have en finger med i spillet.

- Linjelængden skal ikke være større end den bliver med IMFUFA-L^AT_EXs dokumentklasser, måske snarere mindre.
- Skriftgraden skal være 10 punkt til normalt-læsende voksne; hvis der optræder matematiske symboler med top- og fodtegn i mange niveauer, kan det måske være nødvendigt at benytte 11 eller 12 punkt.
- Linjeafstanden må ikke være for lille. Standard L^AT_EX bruger 12 punkts skydning af 10 punkts skriften, men det kan muligvis øge læseligheden med en lidt større skydning (f.eks. 13 punkt). IMFUFA-L^AT_EXs dokumentklasser har (jf. side 17) en valgmulighed `pplleading` der giver en lidt større skydning. Resultatet heraf ses i nærværende publikation.
- Læseligheden afhænger i høj grad af valget af skrift. Man bør næppe skrive længere afsnit i blokskrift. NÆPPE HELLER I KAPITÆLER.
- Tekstens læselighed kan øges og det generelle æstetiske indtryk forbedres med små mikrotypografiske justeringer. Den meget anbefalelsesværdige *microtype*-pakke (Schlicht, 2005) kan udføre ‘character protrusion’ der bevirker at man i stedet for en eksakt lige bagkant får en *optisk lige bagkant* (hvor visse tegn, bl.a. orddelelingsbindestregen får lov at gå en smule ud over bagkanten), og ‘font expansion’ (*fontjustering*), dvs. der komme en lille bitte smule elastik i bredden af de ikke-blanke tegn (der er i forvejen en vis elastik i de blanke tegn). En sammenligning af figur 5.1 og 5.2 demonstrerer klart at de mikrotypografiske justeringer gør tekstbilledet pænere.

Det er overordentlig nemt at bruge *microtype*, man behøver blot inkludere pakken, så sker resten (eller i hvert fald det meste) af sig selv. Pakken kan dog også konfigureres på forskellig vis, se Schlicht (2005).

- *microtype* virker kun når man benytter `pdflatex`-programmet, og ‘font expansion’ er p.t. kun mulig når man bruger `pdflatex` til at fremstille *pdf*-output. (I de fleste nyere L^AT_EX-installationer er det rent faktisk `pdflatex` der »bag kulissen« bruges til fremstilling også af *dvi*-output, og i så fald kan man udmærket få ‘character protrusion’ i sit *dvi*-output.)
- Normalt bliver teksten sat med *fast bagkant* (lige højremargin), men det kan også lade sig gøre at få *løs bagkant*, sådan som det ses i visse aviser og bøger. Det kan for eksempel opnås med kommandoen `\RaggedRight`.

Den tidligere (side 40) anvendte Musil-tekstprøve sat på to forskellige måder.

Egentlig var det kun matematikerne selv og deres disciple, naturforskerne, der levede i uvidenhed om disse farer, de fornemmede lige så lidt af alt dette i deres sjæl som cykelryttere, der flittigt tramper løs og ikke lægger mærke til andet i verden end den forankørendes baghjul. Om Ulrich kunne man derimod med sikkerhed sige det ene, at han elskede matematikken på grund af de mennesker, der ikke kunne udstå den. Han var mindre videnskabeligt end menneskeligt forelsket i videnskaben. Han så, at den i alle spørgsmål, hvor den betragter sig som kompetent, tænker anderledes end almindelige mennesker. Hvis man i stedet for videnskabelige anskuelser satte livsanskuelse, i stedet for hypotese forsøg og i stedet for sandhed handling, var der ingen betydelig naturforsker eller matematiker, hvis livsværk ikke langt overgik historiens største bedrifter.

Figur 5.1 Sat normalt.

Egentlig var det kun matematikerne selv og deres disciple, naturforskerne, der levede i uvidenhed om disse farer, de fornemmede lige så lidt af alt dette i deres sjæl som cykelryttere, der flittigt tramper løs og ikke lægger mærke til andet i verden end den forankørendes baghjul. Om Ulrich kunne man derimod med sikkerhed sige det ene, at han elskede matematikken på grund af de mennesker, der ikke kunne udstå den. Han var mindre videnskabeligt end menneskeligt forelsket i videnskaben. Han så, at den i alle spørgsmål, hvor den betragter sig som kompetent, tænker anderledes end almindelige mennesker. Hvis man i stedet for videnskabelige anskuelser satte livsanskuelse, i stedet for hypotese forsøg og i stedet for sandhed handling, var der ingen betydelig naturforsker eller matematiker, hvis livsværk ikke langt overgik historiens største bedrifter.

Figur 5.2 Sat med optisk lige bagkant og fontjustering.

- Nyt afsnit markeres almindeligvis ved linjeskift efterfulgt af en vis mindre indrykning, jf. side 10. Denne metode, som er L^AT_EX's standardmetode, er benyttet i nærværende publikation

Alternativt kan man markere nyt afsnit ved linjeskift med ekstra nedrykning efterfulgt af ingen indrykning; det gør L^AT_EX i opremsninger som for eksempel den du læser i øjeblikket.

IMFUF_A-L^AT_EX indeholder kommandoen `\imfufapar` der skifter over til den alternative måde at markere nyt afsnit på.

Man kan naturligvis diskutere om man *altid* skal stræbe efter at gøre tekstbilledet så læse-bekvemst som muligt; tekster der kræver eftertanke og fordybelse skal måske netop udformes på en måde der *nedsætter* læsehastigheden ...

5.4 Orddeling

L^AT_EX har en indbygget orddelingsalgoritme der benytter et sprogspecifikt regelsæt. Desuden er der for hvert sprog en kortere eller længere liste med undtagelser, dvs. ord som orddelingsalgoritmen ikke af sig selv kan dele rigtigt (f.eks. et ord som *ukrudtsart*).

Den enkelte bruger kan imidlertid også selv være med til at bestemme hvor ord må deles. Man kan benytte kommandoen `\-` til at markere steder hvor et ord må deles, for eksempel `ukrudts\-\art`; bemærk at der ikke sker nogen »indlæring«, man skal indsætte `\-` ved hver eneste forekomst af ordet. Hvis der et tale om et ord der optræder mange steder i teksten, kan det derfor betale sig at gøre noget andet, nemlig benytte

`\-`

`\hyphenation` kommandoen `\hyphenation` til at definere ekstra orddelelsesundtagelser. Eksempel:

```
\hyphenation{ukrudts-art ukrudts-ar-ten ukrudts-ar-ter ukrudts-ar-terne}
```

EKS. 5.14

`\showhyphens`

Undertiden kan man være interesseret i at få at vide hvordan $\text{\LaTeX}$ i givet fald vil dele et bestemt ord. Til det formål kan man benytte kommandoen `\showhyphens{<ord>}` der i *log-filen* skriver hvordan *<ord>* kan deles ifølge $\text{\LaTeX}$; kommandoen giver ikke noget synligt resultat.

I et flersproget dokument kan man placere en `\selectlanguage`-kommando lige før `\hyphenation`-kommandoen for at fortælle hvilket sprog orddelelsesundtagelserne hører til.

5.5 Streger og mellemrum

Der er talrige småting der kan genere og irritere den pedantiske læser – også selv om vi ser helt bort fra tekstens *indhold*.

Bindestreger, tankestreger og minusser er både syntaktisk og i $\text{\LaTeX}$ -forstand tre helt forskellige ting. Ydermere kan forskellige sprog have forskellige traditioner for hvordan f.eks. tankestreger ser ud. En dansk tankestreg fremstilles med kommandoen `--` med mellemrum før og efter:

Koden for tankestreg er <i>to</i> streger – og der skal mellemrum før og efter.	Koden for tankestreg er <code>\textit{to}</code> streger <code>--</code> og der skal mellemrum før og efter.
---	--

EKS. 5.15

`---` En engelsk tankestreg derimod fremstilles med `---`, og der skal ikke mellemrum mellem tankestregen og den omgivende tekst.

Bindestreg skrives med en enkelt streg: årene 2007-10, kl. 8-16.	Bindestreg skrives med en enkelt streg: årene 2007-10, kl. 8-16.
--	--

EKS. 5.16

`"-` $\text{\LaTeX}$ deler normalt ikke ord der er sat sammen med en bindestreg, andre steder end ved bindestregen. Men hvis man i stedet for bindestregen skriver `"-`, er $\text{\LaTeX}$ villig til at dele de to orddele:

I sammensatte ord (f.eks. computer-specialist) kan man indsætte en bindestreg.	I sammensatte ord (f.eks. computer-specialist) kan man indsætte en bindestreg.
I sammensatte ord (f.eks. computer-specialist) kan man indsætte en bindestreg.	I sammensatte ord (f.eks. computer"-specialist) kan man indsætte en bindestreg.

EKS. 5.17

Denne facilitet findes i $\text{\LaTeX}$ når det aktuelle sprog er dansk; der findes også kommandoen `" /` der virker på lignende måde, blot skriver den en `/` i stedet for en `-`.

- Minustegn er atter noget andet; de hører hjemme i matematiksammenhæng, og der er forskel på den dyadiske operator minus og fortegnet minus:

EKS. 5.18

Det er typografisk og æstetisk ukorrekt at skrive $5-7=-2$. Det rigtige er $5 - 7 = -2$.	Det er typografisk og æstetisk ukorrekt at skrive $5-7=-2$. Det rigtige er $\$5-7=-2\$$.
--	---

Som det ses sørger $\text{\LaTeX}$ for at sætte større mellemrum om den dyadiske operator minus end om fortegnet minus.

$\text{\LaTeX}$ sørger for at ordmellemrummene får en passende størrelse, men brugeren skal selv markere *hvor* der skal være mellemrum og hvor ikke. Det indebærer at den bruger der vil følge almindelig dansk skik, selv skal skrive mellemrumstegn *efter* (men ikke foran) tegn som punktum, semikolon, kolon, komma, udråbstegn og spørgsmålstegn. Det har samme effekt om man skriver et eller mange mellemrumstegn; i de fleste situationer som den almindelige bruger kommer ud for, har et enkelt linjeskift samme betydning som et mellemrumstegn.

Der er forskellige sammenhænge hvor man *ikke* må skrive mellemrum, for eksempel først i argumentet til en ‘sectioning command’:

EKS. 5.19

Overskriften bør flugte med venstrekanten:	Overskriften bør flugte med venstrekanten:
Forkert	<code>\subsection{ Forkert}</code>
Rigtigt	<code>\subsection{Rigtigt}</code>

Et mellemrum der ikke må erstattes af linjeskift, skrives som en tilde (~):

EKS. 5.20

Som Valdemar Atterdag sagde: I morgen er der atter en dag.	Som Valdemar Atterdag sagde: I~morgen er der atter en dag.
--	---

6 Matematik i L^AT_EX

At skrive tekster der indeholder matematiske argumentationer og udredninger, er ingen let sag. Forfatteren skal først og fremmest gøre sig klart hvad det er der skal siges og formidles, samt sørge for at argumenterne er rigtige og så simple og forståelige som muligt. Dernæst skal det hele omsættes til en læselig skriftlig fremstilling hvor formler og tekst er blandet i det rette forhold og på den rette måde, hvilket blandt andet vil sige at korte formler normalt står inde i teksten, hvorimod længere formler fremhæves.

Standard L^AT_EX indeholder udmærkede redskaber til fremstilling af matematiske tekster, men det kan godt være lidt bøvlet at få større matematikopstillinger til at se pæne ud medmindre man benytter værktøjerne i American Mathematical Society's pakke `amsmath`. **I det følgende går vi ud fra at `amsmath`-pakken er indlæst.**

- Med dokumentklasserne fra IMFUFA-L^AT_EX kan man med fordel indlæse `amsmath`-pakken ved at angive `amsmath` som valgmulighed til `\documentclass`-kommandoen.

Ovenstående råd er begrundet i at IMFUFA-L^AT_EX-klasserne altid indlæser pakken `amsthm`, og hvis man vil bruge både `amsmath` og `amsthm`, så skal `amsmath` indlæses først, og det kan brugeren ikke komme til med en `\usepackage{amsmath}`.

6.1 Simple formler

For matematiske formler gælder der særlige typografiske regler. Det der først falder i øjnene, er at formler sættes med en særlig kursivskrift: Newtons anden lov opsummeres kort som $F = ma$ (og bestemt ikke som $F=ma$). I overensstemmelse med den sædvanlige L^AT_EX-filosofi skal forfatteren ikke bekymre sig om hvad det nu er for en skrift der skal bruges, men kun fortælle L^AT_EX hvor der skiftes fra *tekst-sammenhæng* ('text mode') til *matematik-sammenhæng* ('math mode') og tilbage igen. Disse skift foregår forskelligt afhængigt af om der er tale om en tekstformel eller en fremhævet formel.

Formler inde i teksten afgrænses af `$ $` eller `\( \)` :

`$ $` `\( \)`

Newton's anden lov $F = ma$ er et eksempel på en naturlov.

Newton's anden lov $F=ma$ er et eksempel på en naturlov.

Newton's anden lov $F = ma$ er et eksempel på en naturlov.

Newton's anden lov $\(F=ma\)$ er et eksempel på en naturlov.

Eks. 6.1

Som man hurtigt vil erfare har mellemrumstegn ingen betydning i formler, det er altså ligegyldigt om man skriver $F=ma$ eller $F = m a$.

Fremhævede formler ('displayed math') bliver sat centreret og adskilt fra den almindelige tekst; i input-teksten markeres de med `\[ \]`:

Newtons anden lov $F = ma$ er et eksempel på en naturlov.	<code>Newtons anden lov</code> <code>\[F=ma \]</code> <code>er et eksempel på en naturlov.</code>
---	--

Eks. 6.2

- Der skal *ikke* være en blank linje lige før `\[`, og der skal heller ikke være en blank linje lige efter `\]`, medmindre der skal være et »nyt afsnit«. (De blanke linjer bevirker at der kommer forkerte linjeafstande.)

Når en formel skal stå i en almindelig tekstlinje, prøver L^AT_EX temmelig meget på at klemme formelen sammen i højden så den passer til den normale linjeafstand; derfor vil den samme formel få forskelligt udseende afhængigt af om den står som tekstformel eller fremhævet formel. Det illustreres med følgende eksempel (de anvendte kommandoer forklares nærmere i det følgende):

Summen af en kvotientrække er givet ved $\sum_{k=0}^{n-1} z^k = r \frac{1-z^n}{1-z}$ hvilket vises således: Summen af en kvotientrække er $\sum_{k=0}^{n-1} z^k = \frac{1-z^n}{1-z}$ hvilket vises således:	Summen af en kvotientrække er givet ved <code>\$ \sum_{k=0}^{n-1} z^k =</code> <code>\frac{1-z^n}{1-z} \$</code> hvilket vises således: Summen af en kvotientrække er <code>\[\sum_{k=0}^{n-1} z^k =</code> <code>\frac{1-z^n}{1-z} \]</code> hvilket vises således:
--	--

Eks. 6.3

`\displaystyle`
`\textstyle`

Med kommandoerne `\displaystyle` og `\textstyle` kan man tvinge L^AT_EX til at benytte de regler der gælder for henholdsvis fremhævede formler og tekst-formler, også selv om det egentlig ikke hører til i sammenhængen:

Summen af en kvotientrække er givet ved $\sum_{k=0}^{n-1} z^k = \frac{1-z^n}{1-z}$ hvilket vises således: Summen af en kvotientrække er $\sum_{k=0}^{n-1} z^k = \frac{1-z^n}{1-z}$ hvilket vises således:	Summen af en kvotientrække er givet ved <code>\$\displaystyle</code> <code>\sum_{k=0}^{n-1} z^k =</code> <code>\frac{1-z^n}{1-z} \$</code> hvilket vises således: Summen af en kvotientrække er <code>\[\textstyle</code> <code>\sum_{k=0}^{n-1} z^k =</code> <code>\frac{1-z^n}{1-z} \]</code> hvilket vises således:
--	---

Eks. 6.4

`\boxed`

Man kan få *indrammede formler* med kommandoen `\boxed{<formel>}`:

Funktional ligningen

$$\boxed{f(x) + f(y) = f(x + y)}$$

har løsninger af formen $\boxed{f(x) = ax}$.

Funktional ligningen

`\[ \boxed{f(x)+f(y)=f(x+y)} \]`
 har løsninger af formen
`\boxed{f(x)=ax}`.

Eks. 6.5

6.2 Grundprincipperne

Al input til $\text{\LaTeX}$ sker med almindelige bogstaver og tegn som skrives på computer-tastaturet, også specifikationen af matematiske formler sker ved brug af $\text{\LaTeX}$ -kommandoer. De fleste matematikkommandoer har »bogstav-navne«, eksempelvis er `\sum` kommandoen for et sumtegn ($\sum$); visse ting angives dog med særlige tegn. Vi præsenterer nu de vigtigste matematikkommandoer og -kommandotyper.

Fod- og toptegn

Fodtegn ('subscript') markeres med tegnet `_` (understreg), og toptegn ('superscript') markeres med tegnet `^` (cirkumfleks); det der skal stå som fodtegn eller toptegn, omgives af `{ }`:

$$\begin{array}{ll} z^2 & e^{i(s+t)} \\ n_k & x_{n_k} \end{array} \qquad \begin{array}{ll} z^{\{2\}} & e^{\{i(s+t)\}} \\ n_{\{k\}} & x_{\{n_{\{k\}}\}} \end{array}$$

Eks. 6.6

Det er ikke syntaktisk forkert at udelade `{ }`, men så bliver kun det umiddelbart efterfølgende tegn fod- eller toptegn:

$$\begin{array}{ll} e^i t & e^{it} \\ e^i t & e^{\{it\}} \end{array}$$

Eks. 6.7

Sum, produkt, integral

Sumtegn, *produkttegn* og *integraltegn* fremstilles med kommandoerne `\sum`, `\prod` og `\int`; eventuelle grænser angives som fod- og toptegn. Sum-, produkt- og integraltegn (samt visse andre tegn, se side 78) får forskellig størrelse afhængigt af om de står i en fremhævet formel eller i en tekstformel, og top- og fodtegn placeres også forskelligt i de to tilfælde. (For sumtegnets vedkommende illustreres dette i eks. 6.3.)

$$\begin{array}{ll} \sum_{n=0}^{\infty} z^n = \frac{1}{1-z} & \text{\texttt{\sum_{n=0}^{\infty} z^{\{n\}} =}} \\ & \text{\texttt{\frac{1}{1-z}}} \\ \prod_{k=1}^n k = n! & \text{\texttt{\prod_{k=1}^{\{n\}} = n!}} \\ \int_0^{+\infty} e^{-t} dt = 1 & \text{\texttt{\int_{0}^{\{+\infty\}} e^{\{-t\}} dt =1}} \end{array}$$

Eks. 6.8

Mellemrum

I matematiksammenhæng skal en bogstavsekvens som for eksempel *sut* sædvanligvis ikke opfattes som ordet bestående af bogstaverne s, u og t, men derimod som produktet af de tre størrelser *s*, *u* og *t*. Da man således ikke opererer med ord, men med symboler bestående af enkeltbogstaver, er tegnet *mellemrum* blevet tildelt en anden betydning i matematiksammenhæng end i tekstsammenhæng:

Når man skriver formler, bliver mellemrum ikke til synlige mellemrum: <i>sut</i> = <i>sut</i> .	Når man skriver formler, bliver mellemrum ikke til synlige mellemrum: <code>\$sut = s u t\$</code> .
---	--

EKS. 6.9

Undertiden kan man godt ønske sig andre afstande mellem enkeltdelene i en formel end dem som L^AT_EX laver af sig selv, det kan øge læseligheden. Således burde der i integralet i eks. 6.8 nok være lidt større afstand mellem e^{-t} og dt end der faktisk er; her er en bedre udgave:

$\int_0^{+\infty} e^{-t} dt = 1$	<code>\int_{0}^{+\infty} e^{-t}\,dt = 1</code>
----------------------------------	--

EKS. 6.10

Der findes blandt andet følgende kommandoer der kan indsætte mellemrum i formler; den vandrette afstand mellem de sorte rektangler viser mellemrummets størrelse:

kommando	eksempel	beskrivelse
		intet mellemrum
<code>\!</code>		negativt mellemrum
<code>\,</code>		lille mellemrum
<code>\:</code>		middel mellemrum
<code>_</code>		et ord-mellemrum
<code>\enspace</code>		en halv geviert
<code>\quad</code>		en geviert
<code>\qquad</code>		to gevierter

(En *geviert* er lig bredden af et M i den aktuelle skrift.) Med undtagelse af `\!` og `\:` kan disse kommandoer også anvendes i tekstsammenhæng.

Lysets hastighed i vacuum er 299 792 458 m/s.	Lysets hastighed i vacuum er <code>299\,792\,458 m/s</code> .
Lysets hastighed i vacuum er 299 792 458 m/s.	Lysets hastighed i vacuum er <code>\$299\,792\,458\$ m/s</code> .

EKS. 6.11

Almindelige regneoperationer

Almindelige regnestykker skrives stort set som man umiddelbart skulle tro. Som multiplikationstegn bruges alt efter omstændighederne eller sædvane eller smag og behag `\cdot`

enten ingenting eller en centreret prik eller tegnet $\times$:

`\times`

Eks. 6.12

$ab = a \cdot b$	<code>\$ab=a \cdot b\$</code>
$3 \times 5 = 15$	<code>\$3 \times 5 = 15\$</code>
$(a+b)/c = a/c + b/c$	<code>\$(a+b)/c=a/c + b/c\$</code>

I oversigterne side 78ff kan man se L^AT_EX's sortiment af aritmetiske (og andre) operatorer.

Operatorer som $+$ og $-$ kan være både unære og binære, og de skal have forskellige størrelser mellemrum omkring sig i de to tilfælde; det får de også:

$+$ $-$

Eks. 6.13

Det inverse element til a er $-a$. Subtraktion defineres ved $b - a = b + (-a)$.	Det inverse element til <code>\$a\$</code> er <code>\$-a\$</code> . Subtraktion defineres ved <code>\$b-a=b+(-a)\$</code> .
Der er forskellige grader af ingenting: sammenlign $+b$ og $+b$.	Der er forskellige grader af ingenting: sammenlign <code>\$\{ \}+b\$</code> og <code>\$+b\$</code> .

Hvor eks. 6.13 måske hører til i petitesseafdelingen, så er det næste alvorligere:

Eks. 6.14

Negative tal bør altid skrives som matematik: -5 og ikke -5 .	Negative tal bør altid skrives som matematik: <code>\$-5\$</code> og ikke -5 .
I kommatal skal »kommaet« være et punktum: 23.7 er resultatet af $237/10$, hvorimod $23,7$ ($= 23,7$) er en liste bestående af de to tal 23 og 7.	I kommatal skal <code>\enquote{kommaet}</code> være et punktum: <code>\$23.7\$</code> er resultatet af <code>\$237/10\$</code> , hvorimod <code>\$23,7\$</code> (<code>= \$23 , 7\$</code>) er en liste bestående af de to tal 23 og 7.

Hvis man har tænkt sig at skrive kommatal med komma, kan man overveje at benytte pakken `icomma` der definerer et »intelligent komma« i den forstand at `$23,7$` giver tallet 23,7 (*uden* mellemrum foran 7), hvorimod `$23 , 7$` giver listen bestående af de to tal 23 og 7, dvs. 23,7 (*med* mellemrum foran 7).

Navngivne funktioner og operatorer

En del matematiske funktioner og operatorer har standardbetegnelser som er forkortelser af deres navne, f.eks. $\sin$, $\log$, $\limsup$. Disse må *ikke* skrives med kursiv, og det bliver de heller ikke hvis man bærer sig rigtigt ad, dvs. hvis man skriver betegnelserne som L^AT_EX-kommandoer, f.eks.

Eks. 6.15

$\exp(ix) = \cos x + i \sin x$	<code>\exp(ix)=\cos x + i \sin x</code>
--------------------------------	---

Visse af kommandoerne har den egenskab at i fremhævede formler kommer et eventuelt fodtegn kommer til at stå *under* kommandonavnet:

Eks. 6.16

$\lim_{n \rightarrow \infty} x_n$	<code>\lim_{n \rightarrow \infty} x_{n}</code>
-----------------------------------	--

På side 78 er en liste over samtlige navngivne funktioner.

Relationer

L^AT_EX har en lang række relationssymboler. Her er nogle der vedrører tal:

$$\begin{array}{ll} x = x, x^2 \geq 0, -x^2 \leq 0, & \$x=x$, \$x^2 \geq 0$, \$-x^2 \leq 0$, \\ 5 > 3, 3 < 5, 3 \neq 5. & \$5 > 3$, \$3 < 5$, \$3 \neq 5$. \end{array}$$

Eks. 6.17

Her er nogle der vedrører mængder:

$$\begin{array}{ll} x \in A, A \ni x, & \$x \in A$, \$A \ni x$, \\ A \subset B, B \supset A, & \$A \subset B$, \$B \supset A$, \\ A \subseteq B, B \supseteq A. & \$A \subseteq B$, \$B \supseteq A$. \end{array}$$

Eks. 6.18

`\not` Man kan negere relationerne med kommandoen `\not`:

$$\begin{array}{ll} 5 \not< 3, 8 \not\geq 9, & \$5 \not< 3$, \$8 \not\geq 9$, \\ x \not\in A, B \not\subset A. & \$x \not\in A$, \$B \not\subset A$. \end{array}$$

Eks. 6.19

`\ton` Hvis man vil være bagvendt, kan man bruge IMFUFA-L^AT_EX-kommandoen `\ton`:

$$\begin{array}{ll} 5 \not< 3, 8 \not\geq 9, & \$5 \not< 3$, \$8 \not\geq 9$, \\ x \not\in A, B \not\subset A. & \$x \not\in A$, \$B \not\subset A$. \end{array}$$

Eks. 6.20

I oversigterne side 78ff findes talrige andre relationssymboler.

Brøker og lignende

`\frac` Brøker fremstilles med kommandoen `\frac`. Man kan også bruge IMFUFA-L^AT_EX-kommandoen `\nicefrac` (se eks. 2.8 side 11) eller `amsmath`-kommandoerne `\dfrac` (der svarer til `\displaystyle\frac`) og `\tfrac` (der svarer til `\textstyle\frac`). Størrelsen af en brøk fremstillet med `\frac` afhænger af omgivelserne:

$$\begin{array}{ll} \text{En fjerdingvej er } \frac{1}{4} \text{ mil.} & \text{En fjerdingvej er } \$\frac{1}{4}$ mil. \\ \frac{1}{3} - \frac{1}{4} = \frac{1}{12} & \$\left[\frac{1}{3} - \frac{1}{4} \right] \\ & = \frac{1}{12} \$ \end{array}$$

Eks. 6.21

Hvis der er et langt udtryk i tæller og/eller nævner, kan man øge læseligheden ved at gøre brøkstregen lidt længere end det længste udtryk; det gøres ved at sætte et passende mellemrum før og efter udtrykket (se også eks. 8.6 side 100):

$$\begin{array}{ll} \frac{1}{\sqrt{x(1-x)}} & \$\frac{1}{\sqrt{x(1-x)}}$ \\ \frac{1}{\sqrt{x(1-x)}} & \$\frac{1}{\sqrt{x(1-x)}},\$ \end{array}$$

Eks. 6.22

`\binom` Kommandoen `\binom` fremstiller binomialkoefficienter:

En binomialkoefficient $\binom{n}{k}$.

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

En binomialkoefficient
`\binom{n}{k}`.

`\[ \binom{n}{k} = \frac{n!}{k!(n-k)!} \]`

EKS. 6.23

Ting over og under formler

Der er særlige kommandoer til accenter i matematiksammenhæng:

$\acute{x}$ $\grave{x}$ $\hat{x}$ $\tilde{x}$ $\bar{x}$

$\check{x}$ $\breve{x}$ $\mathring{x}$ $\dot{x}$ $\ddot{x}$ $\vec{x}$

`\acute{x}` `\grave{x}`
`\hat{x}` `\tilde{x}` `\bar{x}`
`\check{x}` `\breve{x}` `\mathring{x}`
`\dot{x}` `\ddot{x}` `\vec{x}`

EKS. 6.24

Man kan sætte lange hatte, tilder, streger og pile over formler:

$\widehat{x+y}$ $\widetilde{x+y}$ $\overline{x+y}$

$\underline{x+y}$ $\overrightarrow{x+y}$ $\overleftarrow{x+y}$

`\widehat{x+y}` `\widetilde{x+y}`
`\overline{x+y}` `\underline{x+y}`
`\overrightarrow{x+y}`
`\overleftarrow{x+y}`

EKS. 6.25

Man kan også have glæde af kommandoerne `\underbrace` og `\overbrace`:

$1, 2, \underbrace{3, 3, \dots, 3}_n, 4$

$9, 8, \overbrace{7, 7, \dots, 7}^k, 6$

`\$1,2,`
`\underbrace{3,3,\ldots,3}_{n},4\$`

`\$9,8,`
`\overbrace{7,7,\ldots,7}^{k},6\$`

EKS. 6.26

Kvadratrødder og andre rødder

Kvadratrødder og andre rødder fremstilles med `\sqrt`:

`\sqrt`

Tallet $\sqrt{8}$ er irrationalt. Det er $\sqrt[3]{8}$ ikke.

Tallet `\sqrt{8}` er irrationalt.
Det er `\sqrt[3]{8}` ikke.

EKS. 6.27

Differentiation

Det almindelige differentiations-mærke er den engelske ejefaldsapostrof. Nogle gange betegner man den første, anden osv. tidsafledede med en, to osv. prikker over funktionssymbolet; hertil bruges `\dot`, `\ddot` osv. Til partielle afledede benyttes symbolet `\partial`, og nablaoperatoren hedder `\nabla`:

`'`
`\dot` `\ddot`
`\partial`
`\nabla`

$f'' = (f')'$	<code>\$f''=(f')'\$</code>
$a = \dot{v} = \ddot{x}$	<code>\$a = \dot{v} = \ddot{x}\$</code>
∇f	<code>\$\nabla f\$</code>
$\frac{\partial f}{\partial t} = \frac{\partial^2 f}{\partial x^2}$	<code>\$\dfrac{\partial f}{\partial t} = \dfrac{\partial^2 f}{\partial x^2}\$</code>

Eks. 6.28

Pile

L^AT_EX har et stort udvalg af pile. Her er nogle få eksempler (der er flere på side 83):

Funktionen $x \mapsto x^2$ er kontinuert.	Funktionen <code>\$x \mapsto x^2\$</code>
$\ln x \rightarrow 0 \iff x \rightarrow 1.$	er kontinuert.
	<code>\$\ln x \rightarrow 0 \iff</code>
	<code>x \rightarrow 1\$.</code>

Eks. 6.29

`\xleftarrow`
`\xrightarrow`

Med `\xleftarrow[⟨under⟩]{⟨over⟩}` og `\xrightarrow[⟨under⟩]{⟨over⟩}` får man pile der kan have tekst over og/eller under:

$\ln x \xrightarrow{x \rightarrow 1} 0$	<code>\$\ln x \xrightarrow[x \rightarrow 1]{0}\$</code>
$x^3 \xleftarrow{f} x$	<code>\$x^3 \xleftarrow{f} x\$</code>

Eks. 6.30

Diverse tegn; talmængderne

Kommandoerne for græske bogstaver til matematikbrug er nemme, de er ganske enkelt bogstavets navn med en `\` foran. Bemærk at navnene skal staves på engelsk, således at kommandoen for α er `\alpha`, ikke `\alfa`. På side 84 er der en liste over de græske bogstaver og deres kommandoer.

Der findes adskillige matematiske tegn, f.eks. $\emptyset$ (`\emptyset`), $\varnothing$ (`\varnothing`), $\forall$ (`\forall`), $\exists$ (`\exists`), $\neg$ (`\neg`), $\hbar$ (`\hbar`). Se i øvrigt oversigterne side 78ff.

IMFUF_A-L^AT_EX omdefinerer `\Pr` (fra `Pr` til `P`) og definerer endvidere `\Exp` og `\Var`:

$P(A \cap B) = P(A) \cdot P(B)$ hvis A og B er uafhængige.	<code>\$\Pr(A \cap B) = \Pr(A) \cdot \Pr(B)\$</code>
$\text{Var } X = E(X - E X)^2$	<code>\$\Var X = \Exp(X - \Exp X)^2\$</code>

Eks. 6.31

`\N` `\Z` `\Q`
`\R` `\C`

Desuden indeholder IMFUF_A-L^AT_EX kommandoerne `\N`, `\Z`, `\Q`, `\R` og `\C` der fremstiller symboler for de almindelige talmængder:

Eks. 6.32

De naturlige tal $\mathbb{N}$	De naturlige tal $\mathbb{N}$
De hele tal $\mathbb{Z}$	De hele tal $\mathbb{Z}$
De rationale tal $\mathbb{Q}$	De rationale tal $\mathbb{Q}$
De reelle tal $\mathbb{R}$	De reelle tal $\mathbb{R}$
De komplekse tal $\mathbb{C}$	De komplekse tal $\mathbb{C}$
$(x, y) \in \mathbb{R}^2$	$(x, y) \in \mathbb{R}^2$

Det er god L^AT_EX-filosofi at opretholde en fast sammenhæng mellem L^AT_EX-symbolerne og deres *matematiske* indhold, og så justere symbolernes *typografiske* indhold efter ønske. Det betyder at hvis man af en eller anden grund hellere vil skrive de reelle tal som $\mathbb{R}$, så skal man bibeholde kommandoen `\mathbb{R}` for de reelle tal, men omdefinere dens typografiske indhold, se hvordan i eks. 8.7 på side 101. Det kan nævnes at kommandoerne `\mathbb{N}`, `\mathbb{Z}`, osv. forlods er defineret som `\mathbb{N}`, `\mathbb{Z}`, osv. (`\mathbb{N}` omtales side 76).

6.3 Tekst i formler

Undertiden har man brug for at indsætte tekst i formler. Det kan gøres ved hjælp af kommandoen `\text` der er til rådighed via `amsmath`-pakken (eller `amstext`-pakken). (Man kan også benytte de almindelige kommandoer til tekst-skriftvalg, jf. tabel 5.1 side 51.) Som det fremgår af nedenstående eksempel, tilpasser `\text` automatisk tekstens størrelse efter sammenhængen.

`\text`

Eks. 6.33

Da $x^2 \geq 0$ for alle x ,	Da $x^2 \geq 0$ <code>\text{ for alle } x</code> ,
Gnidningskraften F_{gnidn} er 0.	Gnidningskraften <code>\text{ } F_{\text{gnidn}}</code> er 0.
$1\text{\AA} = 10^{-10}\text{m}$	<code>1\text{\AA}=10^{-10}</code> <code>\text{m}</code>

- Betydningen af tegnet mellemrum er forskelligt i matematik- og i tekstsammenhæng; derfor skal mellemrummene før, mellem og efter ordene »for« og »alle« skrives *inde i* argumentet til `\text`.

Kommandoen `\text` sætter sit argument med samme skrift som der bruges til den omgivende tekst, og det kan have sine fordele og ulemper. Hvis den omgivende tekst er i kursiv (hvilket f.eks. matematiske sætninger ofte er), bliver resultatet

Eks. 6.34

<i>Da $x^2 \geq 0$ for alle x,</i>	<code>\itshape</code> Da $x^2 \geq 0$ <code>\text{ for alle } x</code> ,
<i>Gnidningskraften F_{gnidn} er 0.</i>	Gnidningskraften <code>\text{ } F_{\text{gnidn}}</code> er 0.
<i>$1\text{\AA} = 10^{-10}\text{m}$</i>	<code>1\text{\AA}=10^{-10}</code> <code>\text{m}</code>

Det er rigtigt at »for alle« skal i kursiv, men det er en alvorlig fejl at skrive enhederne ($\text{\AA}$ og m) med kursiv; desuden bør »gnidn« skrives på samme måde altid. Her er en bedre løsning hvor vi bruger `\textnormal` til de ting der ikke må kursiveres:

`\textnormal`

Man kan bruge andet end parenteser (nemlig alle ‘delimiters’ (se side 79, afsnittet Parenteser og parentesagtige tegn)) i forbindelse med **big**-kommandoerne.

Eks. 6.39

$$\left(\sum_{i=1}^{n-1} \left| \frac{z_i}{z_{i+1}} \right| \right)^2$$

```
\[ \biggl( \sum_{i=1}^{n-1} \Bigl|
\frac{z_{i}}{z_{i+1}} \Bigr|
\biggr)^2 \]
```

Bogstaverne **l** og **r** står for left og right; hvis man ønsker en forstørret ‘delimiter’ som hverken skal være en venstre- eller en højre-»parentes«, kan man udelade bogstavet:

Eks. 6.40

$$\left. \frac{\partial x}{\partial t} \right|_{t=0}$$

```
\[ \frac{\partial x}{\partial t} \Big|_{t=0} \]
```

Tre prikker

Ved opremsninger hvor man erstatter mellemliggende elementer med prikker, bruges som bekendt *tre* prikker, hverken mere eller mindre. De frembringes med kommandoen `\ldots` (der også kan anvendes i tekstsammenhæng):

`\ldots`

Eks. 6.41

$$1 + 2 + \dots + n = n(n+1)/2.$$

Lad $z_1, z_2, \dots, z_n$ være komplekse tal.

$$\$1+2+\ldots+n = n(n+1)/2\$.$$

Lad $\$z_{\{1\}}, z_{\{2\}}, \ldots, z_{\{n\}}\$$ være komplekse tal.

Nu kan man imidlertid mene at i et udtryk som $1 + 2 + \dots + n$ ville det se pænere ud hvis prikkerne kom i samme højde som den vandrette streg i +’et, altså $1 + 2 + \dots + n$. I **amsmath**-pakken findes kommandoen `\dots` der selv (til en vis grad) justerer placeringen af de tre prikker efter omgivelserne:

`\dots`

Eks. 6.42

$$1 + 2 + \dots + n = n(n+1)/2.$$

Lad $z_1, z_2, \dots, z_n$ være komplekse tal.

$$\$1+2+\dots+n = n(n+1)/2\$.$$

Lad $\$z_{\{1\}}, z_{\{2\}}, \dots, z_{\{n\}}\$$ være komplekse tal.

Justeringen sker i forhold til det der kommer umiddelbart efter `\dots`. Hvis formelen slutter med kommandoen `\dots` som derfor ikke kan gætte sammenhængen, kan man bruge en af kommandoerne `\dotsc` (prikker ifm. komma), `\dotsm` (prikker ifm. multiplikationsprikker), `\dotsb` (prikker ifm. en binær operator), `\dotsi` (prikker ifm. integraler) og `\dotso` (andet). Eksempel:

Eks. 6.43

$$1 = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots$$

$$1 = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots$$

$$\$1=\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\dots+\frac{1}{8}+\dots\$$$

$$\$1=\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\dots+\frac{1}{8}+\dots\sb{\$}$$

Der findes andre udgaver af »tre prikker« (de kan ses i aktion i eks. 6.50):

...	er prikker på linjen,	<code>\$\ldots\$</code>	er prikker på linjen,
...	er centrerede prikker,	<code>\$\cdots\$</code>	er centrerede prikker,
:	er vertikale prikker,	<code>\$\vdots\$</code>	er vertikale prikker,
⋯	er diagonale prikker.	<code>\$\ddots\$</code>	er diagonale prikker.

Eks. 6.44

`\ntup` IMFUFA-L^AT_EX definerer en kommando `\ntup{⟨x⟩}{⟨n⟩}` der laver n -tupler:

Lad $X_1, X_2, \dots, X_k$ være uafhængige stokastiske variable.	Lad <code>$\\$ \text{\ntup{X}{k}} \\$</code> være uafhængige stokastiske variable.
--	--

Eks. 6.45

Den fulde syntaks er imidlertid `\ntup[⟨bin-op⟩]{⟨x⟩}{⟨n⟩}` hvor `⟨bin-op⟩` er en binær operator (hvis defaultværdi er komma):

$x_1 + x_2 + \dots + x_t$	<code>$\\$ \text{\ntup{+}{x}{t}} \\$</code>
$B_1 \cap B_2 \cap \dots \cap B_n$	<code>$\\$ \text{\ntup[\cap]{B}{n}} \\$</code>

Eks. 6.46

6.5 Matricer og arrays

`array` Standard-L^AT_EX har en `array`-konstruktion der kan benyttes til skematiske opstillinger i matematiksammenhæng. Der er mange ligheder mellem `array`- og `tabular`-konstruktioner (side 21): Efter `\begin{array}` skriver man en liste af bogstaver (som kan være `l`, `r` eller `c`) svarende til det ønskede antal søjler; de enkelte felter i hver linje adskilles af `&` og de enkelte linjer adskilles af `\\`. – Et `array` kan kun optræde i matematiksammenhæng.

Et array med tre søjler og to rækker:	Et array med tre søjler og to rækker:
$\begin{array}{ccc} x+y & z & a+\frac{1}{3} \\ i & K^2 & A \cup B \cup C \end{array}$	<pre> \[\begin{array}{cccl} x+y & & z & & a+\frac{1}{3} \\ i & & K^2 & & A \cup B \cup C \end{array}\]</pre>

Eks. 6.47

Som eksemplet antyder bliver de enkelte felter sat efter 'textstyle'-reglerne (jf. side 60); det kan man lave om på ved at indsætte en `\displaystyle` i de felter hvor det skal ændres.

`matrix` Matricer kan fremstilles med `array`, men det er dog oftest lettere at benytte `amsmath`-pakkens `matrix`-konstruktioner. Med `matrix` får man en »matrix« uden parenteser:

$\begin{matrix} a & -b & c \\ d & e & f \end{matrix}$	<pre> \[\begin{matrix} a & -b & c \\ d & e & f \end{matrix}\]</pre>
---	---

Eks. 6.48

Til forskel fra `array` skal `matrix` ikke have en specifikation af hvor mange søjler der skal være, og om de skal centrerer eller højre- eller venstrejusteres (de bliver altid centreret). Med `pmatrix` får man en matrix med runde parenteser:

`pmatrix`En 3×3 -matrix:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

En 3×3 -matrix:

```
\[ A=\begin{pmatrix}
  a_{11} & a_{12} & a_{13}\\
  a_{21} & a_{22} & a_{23}\\
  a_{31} & a_{32} & a_{33}
\end{pmatrix}\]
```

Eks. 6.49

Her er endnu et eksempel (der også demonstrerer forskellige typer prikker (eks. 6.44)):

En $n \times k$ -matrix:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1k} \\ a_{21} & a_{22} & \dots & a_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nk} \end{pmatrix}$$

En $n \times k$ -matrix:

```
\[ A=\begin{pmatrix}
  a_{11} & a_{12} & \ldots & a_{1k}\\
  a_{21} & a_{22} & \ldots & a_{2k}\\
  \vdots & \vdots & \ddots & \vdots\\
  a_{n1} & a_{n2} & \ldots & a_{nk}
\end{pmatrix}\]
```

Eks. 6.50

Man kan endvidere benytte `bmatrix` og `Bmatrix` der sætter kantede hhv. krøllede parenteser ('brackets' og 'braces'), og `vmatrix` og `Vmatrix` der sætter hhv. enkelte og dobbelte lodrette streger omkring. Her er eksempler på resultaterne af disse konstruktioner:

`bmatrix` `Bmatrix`
`vmatrix` `Vmatrix`

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad \begin{Bmatrix} a & b \\ c & d \end{Bmatrix} \quad \left| \begin{array}{cc} a & b \\ c & d \end{array} \right| \quad \left\| \begin{array}{cc} a & b \\ c & d \end{array} \right\|$$

De omtalte matrixkonstruktioner kan tilsyneladende have lige så mange søjler der skal være, men der er dog en grænse. Som default er der en øvre grænse på 10, men den kan om nødvendigt ændres ved at ændre værdien af tælleren `MaxMatrixCols`.

En såkaldt *tuborgfunktion* (eller gaffelfunktion) kan fås med en `cases`-konstruktion:

`cases`Funktionen `sgn` er defineret ved

$$\text{sgn}(x) = \begin{cases} 1 & \text{hvis } x > 0, \\ 0 & \text{hvis } x = 0, \\ -1 & \text{hvis } x < 0. \end{cases}$$

Funktionen $\text{\sgn}$ er defineret ved

```
\[ \text{\sgn}(x) =
\begin{cases}
  1 & \text{\text{hvis } }x>0, \\
  0 & \text{\text{hvis } }x=0, \\
 -1 & \text{\text{hvis } }x<0.
\end{cases} \]
```

Eks. 6.51

Kommandoen `\sgn` »eksisterer ikke«, men er defineret til lejligheden ved hjælp af `\DeclareMathOperator`, se eks. 8.10 side 102.

6.6 Opstillinger og nummererede formler

Nogle formler eller omskrivninger er så lange at de bør fordeles over flere linjer, således at lighedstegnene (eller ulighedstegnene eller hvad det nu er) står pænt under hverandre; i sådanne situationer kan `align*`- eller `align`-konstruktioner bruges.

Opstillinger med `align*`

`align*` Man benytter `align*` til formelopstillinger der strækker sig over flere linjer, og hvor visse ting (f.eks. lighedstegn) skal stå under hinanden. Man bruger tegnet `&` som »tabulator tegn«, og man bruger `\\` til at markere linjeskift:

Der gælder at $\begin{aligned}(a+b)^2 &= a^2 + 2ab + b^2 \\ &\leq a^2 + 2 ab + b^2 \\ &= (a + b)^2\end{aligned}$ for alle reelle tal a og b.	Der gælder at <pre>\begin{align*} (a+b)^2 &= a^2 + 2ab + b^2 \\ &\leq a^2 + 2 ab + b^2 \\ &= (a + b)^2 \end{align*}</pre> for alle reelle tal a og b.
---	--

Eks. 6.52

• Der skal ikke `\\` efter sidste linje.

Hvis den første formel er meget lang, kan resultatet godt se lidt underligt ud:

$\begin{aligned}a + b + c + d + e + f &= g + h \\ &= x^2\end{aligned}$	<pre>\begin{align*} a+b+c+d+e+f &= g+h \\ &= x^2 \end{align*}</pre>
--	---

Eks. 6.53

En bedre løsning kunne være

$\begin{aligned}a + b + c + d + e + f \\ &= g + h \\ &= x^2\end{aligned}$	<pre>\begin{align*} a+b &+ c+d+e+f \\ &= g+h \\ &= x^2 \end{align*}</pre>
---	---

Eks. 6.54

Man kan have flere `&`'er i hver formellinje, så får man en »flerspaltet« formel:

$\begin{aligned}\dot{x} &= f(x,y), & x(0) &= a \\ \dot{y} &= g(x,y), & y(0) &= b\end{aligned}$	<pre>\begin{align*} \dot{x} &= f(x,y), & x(0) &= a \\ \dot{y} &= g(x,y), & y(0) &= b \end{align*}</pre>
--	---

Eks. 6.55

Nummererede formler

Man kan få L^AT_EX til at nummerere fremhævede formler, både enlinjesformler og de enkelte linjer i større opstillinger.

Man får nummereret en enlinjesformel ved at udskifte de omgivende `\[ \]` med `\begin{equation} \end{equation}`:

equation

Eks. 6.56

Den berømte formel		Den berømte formel
$E = mc^2$	(6.1)	<pre>\begin{equation} E=mc^2 \end{equation}</pre>

Formelnumre skrives traditionelt altid til højre på formellinjen (hvilket ikke ser pænt ud i eksempelkasserne); man kan få skrevet dem til venstre på formellinjen ved at angive `leqno` som valgmulighed til dokumentklassen.

Man får nummereret formlerne i en `align*`-konstruktion ved at skrive `align` i stedet for `align*`; her viser vi det med eks. 6.52 (men se også eks. 6.59):

Eks. 6.57

Der gælder at		Der gælder at
$(a+b)^2 = a^2 + 2ab + b^2$	(6.2)	<pre>\begin{align} (a+b)^2 &= a^2 + 2ab + b^2 \\ &\leq a^2 + 2 ab + b^2 \\ &= (a + b)^2 \end{align}</pre>
$\leq a^2 + 2 ab + b^2$	(6.3)	<pre>& \leq a^2 + 2 ab + b^2 \\ & = (a + b)^2</pre>
$= (a + b)^2$	(6.4)	<pre>\end{align} for alle reelle tal \$a\$ og \$b\$.</pre>

Man henviser til de nummererede formler med `\ref` og `\label` sådan som det er beskrevet på side 87ff. En label skal stå på den formellinje den hører til (husk stadig at de enkelte formellinjer adskilles af `\\`).

Der er tradition for at formelnumre står i runde parenteser, ikke bare i formlerne men også når man henviser til dem. Imidlertid skriver $\text{\LaTeX}$ ikke parenteser om de numre der frembringes af `\ref`; man må så selv sætte parenteserne hver gang, eller man kan definere sig en kommando `\eref` der gør det (se hvordan i eks. 8.5 på side 100).

Eks. 6.58

Dette er formel (6.5):		Dette er formel (<code>\ref{formel1}</code>):
$E = mc^2$	(6.5)	<pre>\begin{equation} E=mc^2 \label{formel1} \end{equation} Ovenstående er formel \eref{formel1}.</pre>
Ovenstående er formel (6.5).		

Ét nummer til en flerlinjersformel; `split`

I eks. 6.57 er det klart urimeligt at nummerere *alle* linjerne; man kan forhindre nummeringen af en linje ved at anbringe kommandoen `\nonumber` (eller `\notag`) et eller andet sted på linjen. Men hvis formålet er at tildele den samlede formel et enkelt nummer, er det mere hensigtsmæssigt at benytte en `split`-konstruktion til at fremstille én enkelt flerlinjersformel:

`\nonumber``split`

Der gælder at

$$\begin{aligned}(a+b)^2 &= a^2 + 2ab + b^2 \\ &\leq a^2 + 2|ab| + b^2 \\ &= (|a| + |b|)^2\end{aligned}\quad (6.6)$$

for alle reelle tal a og b .

Der gælder at

```
\begin{equation}
\begin{split}
(a+b)^2 &= a^2 + 2ab + b^2 \\
&\leq a^2 + 2|ab| + b^2 \\
&= (|a|+|b|)^2
\end{split}
\end{equation}
```

for alle reelle tal a og b .

Eks. 6.59

6.7 Sætning og bevis

Matematiske tekster indeholder ofte sætninger og beviser. Disse kan med fordel skrives i theorem- og proof-konstruktioner:

theorem

En vigtig sætning:

Sætning 6.1

Et polynomium af grad n har n komplekse rødder.

En vigtig sætning:

```
\begin{theorem} \label{algft}
  Et polynomium af grad  $n$ 
  har  $n$  komplekse rødder.
\end{theorem}
```

Eks. 6.60

proof

Et **proof** sætter et »bevis-slut«-mærke ($\square$) til sidst:

og her er beviset:

Bevis

Beviset forbigås. $\square$

og her er beviset:

```
\begin{proof}
  Beviset forbigås.
\end{proof}
```

Eks. 6.61

Undertiden kan det være hensigtsmæssigt at præcisere hvad det er for en sætning der bevises; det kan man skrive i [] efter `\begin{proof}`:

og her er beviset:

Bevis for sætning 6.1:

Beviset forbigås. $\square$

og her er beviset:

```
\begin{proof}[for
  sætning \ref{algft}:]
  Beviset forbigås.
\end{proof}
```

Eks. 6.62

Man kan navngive en sætning ved at angive et navn i []:

En vigtig sætning:

Sætning 6.2: Algebraens Fundamentalsætning

Et polynomium af grad n har n komplekse rødder.

En vigtig sætning:

```
\begin{theorem}[Algebraens
  Fundamentalsætning]
  Et polynomium af grad  $n$ 
  har  $n$  komplekse rødder.
\end{theorem}
```

Eks. 6.63

Når man benytter IMFUFA-L^AT_EX, er `theorem` og `proof` automatisk defineret; hvis man benytter andre dokumentklasser (plus `amsmath`), skal man selv definere en sætningstype `theorem` ved hjælp af nedennævnte `\newtheorem`-kommando.

Ofte opererer man ikke blot med sætninger, men også med f.eks. lemmaer, hovedsætninger og korollarer. Så kan man definere nye sætningstyper ved hjælp af kommandoen `\newtheorem{<navn>}{<overskrift>}`. Vi kan eksempelvis definere en `maintheorem`-type til hovedsætninger sådan her: `\newtheorem{maintheorem}{Hovedsætning}`. Den benyttes således:

`\newtheorem`

Eks. 6.64

Der gælder	Der gælder
Hovedsætning 1	<code>\begin{maintheorem}</code>
<i>Dette er en hovedsætning.</i>	Dette er en hovedsætning.
	<code>\end{maintheorem}</code>

Kommandoen `\newtheorem` har forskellige varianter:

```
\newtheorem{<navn>}{<overskrift>}[<nummereres-indenfor>],
\newtheorem{<navn>}[<nummereres-som>]{<overskrift>},
\newtheorem*{<navn>}{<overskrift>}      – her sker ingen nummerering.
```

`\newtheorem*`

Eksempler: Hvis hovedsætningerne ikke skal nummereres fortløbende, men ønskes nummereret inden for kapitler efter systemet `<kapitel-nr>.<hovedsætning-nr>`, skal man definere dem som `\newtheorem{maintheorem}{Hovedsætning}[chapter]`. Hvis man ønsker lemmaer der nummereres med samme nummerering som sætningerne, skal man definere dem som `\newtheorem{lemma}[theorem]{Lemma}`.

Eks. 6.65

Først vises	Først vises
Lemma 6.3	<code>\begin{lemma}</code>
<i>Dette er et lemma.</i>	Dette er et lemma.
	<code>\end{lemma}</code>

Hvis man synes at hovedsætninger overhovedet ikke skal nummereres, kan man definere dem som `\newtheorem*{maintheorem}{Hovedsætning}`.

• `\newtheorem`-kommandoer skal stå i præamblen.

Theoremstyle

Som det ses i eksemplerne, bliver sætningernes `<overskrift>` sat med fed skrift på en linje for sig selv, og sætningens brødtekst bliver sat med kursiv. Sådanne ting bliver fastlagt gennem valg af »sætningsstil« (‘theorem style’) der også fastlægger f.eks. afstanden fra den foregående tekst til overskriften og fra overskriften til sætningens indhold. Sætningsstilen vælges med `\theoremstyle{<stil>}`. En sætningstype får den sætningsstil der er gældende på det sted hvor sætningstypen defineres (i præamblen).

`\theoremstyle`

Der er følgende stile til rådighed: `ibreak` (den der er benyttet ovenfor), `iddefinition` (adskiller sig fra `ibreak` ved at teksten skrives med normal skrift og ikke kursiv), `iproof` (anvendes til `proof`), `plain`, `definition` og `remark`.

De stile hvis navn begynder med `i`, findes kun i IMFUFA-L^AT_EX. Default-stilen er `ibreak` (eller `plain`).

Det er muligt at definere nye »sætningsstile« med kommandoen `\newtheoremstyle`, se nærmere i (Mittelbach m.fl., 2004, afsnit 3.3.3) og/eller (AMS-thm).

6.8 Fed matematik; særlige skrifttyper

For matematiske formler gælder der en række særlige notationsmæssige konventioner med hensyn til brug af skrifttype; en del af de typografiske konsekvenser deraf er indbygget i L^AT_EX:

1. En variabel eller en funktion betegnes meget ofte med et enkelt bogstav, der i den trykte tekst normalt skal skrives med kursiv. Eksempelvis kan man skrive $f(x, y) = 2xy$ for at fortælle at funktionen f anvendt på argumentparret (x, y) returnerer værdien $2xy$; dette sidste udtryk skal pr. konvention forstås som $2 \cdot x \cdot y$ (så der er også en konvention om underforståede multiplikationstegn i spil).
2. Den kursivskrift der benyttes til matematiske symboler, er en særlig matematik-kursiv, '`math italic`', der er anderledes end tekstkursiv. Hvis man i inputteksten f.eks. skriver `$fisk$`, så betyder det (ifølge konventionen om de underforståede multiplikationstegn) produktet af de fire faktorer f , i , s og k , og resultatet ser følgelig sådan ud: *fisk*, altså med et vist mellemrum mellem de enkelte bogstaver. (Ordet *fisk* skrevet med tekstkursiv fremstilles med `\textit{fisk}`.)
3. Matematikkursivskriften benyttes kun til enkeltbogstavssymboler, hvorimod flerbogstavsforkortelser skrives med normalskrift. Det sker automatisk når man bruger de autoriserede kommandoer:

$\log(ab) = \log a + \log b$	<code>\$\log(ab)=\log a + \log b\$</code>
$\sinh x = (e^x - e^{-x})/2$	<code>\$\sinh x = (e^{\{x\}} - e^{\{-x\}})/2\$</code>

Eks. 6.66

4. De matematiske symbols betydning afhænger af hvilken skrift de er sat med, f.eks. kan de tre varianter Q , $\mathbb{Q}$ og $\mathcal{Q}$ af bogstavet Q repræsentere tre helt forskellige matematiske objekter.

Der findes forskellige fontkommandoer til brug i matematiksammenhæng:

`\mathcal`
`\mathscr`

Til »skrevne« (»kalligraferede«) bogstaver findes `\mathcal` og `\mathscr`; man kan kun bruge dem med de store bogstaver fra A til Z:

$ABC\ \mathcal{PQR}\ \mathcal{XYZ}$	<code>\$\mathcal{ABC\ PQR\ XYZ}\$</code>
$ABC\ \mathscr{PQR}\ \mathscr{XYZ}$	<code>\$\mathscr{ABC\ PQR\ XYZ}\$</code>

Eks. 6.67

`\mathbb`

Den såkaldte *blackboard bold* findes også kun for store bogstaver:

$ABC\ \mathbb{PQR}\ \mathbb{XYZ}$	<code>\$\mathbb{ABC\ PQR\ XYZ}\$</code>
-----------------------------------	---

Eks. 6.68

`\mathfrak`

Matematik-fraktur (»krøllede bogstaver«) findes for både store og små bogstaver samt cifre:

Eks. 6.69

 $\mathfrak{PQR} \mathfrak{XYZ} 567 \mathfrak{xyz}$ $\mathfrak{\mathfrak{PQR} \mathfrak{XYZ} 567 \mathfrak{xyz}}$

Kommandoerne `\mathfrak` og `\mathbb` er defineret i `amssymb`- eller `amsfonts`-pakken. Kommandoen `\mathscr` er defineret i pakken `mathrsfs`.

Andre matematikfontkommandoer er `\mathit`, `\mathrm`, `\mathtt` og `\mathsf`.

Fede symboler

Ikke blot bogstavets form har betydning for et symbols betydning, det kan også være af afgørende betydning om det er sat med fed eller normal skrift. Der er afgørende forskel på x , $\boldsymbol{x}$ og $\mathbf{x}$, og på ξ og $\boldsymbol{\xi}$.

Man får en fed udgave af et matematisk symbol med `\bm{\langle formel \rangle}`:

`\bm`

Eks. 6.70

 $3ta + 8a\alpha = 0$ $\mathfrak{3t\bm{a} + 8a\bm{\alpha} = \bm{0}}$

- Kommandoen `\bm` er defineret i pakken `bm`. Denne pakke bør indlæses *efter* eventuelle andre pakker der laver om på matematikfontopsætningen.

Man kan sagtens putte større formeldele ind i `\bm`:

Eks. 6.71

 $\hat{\beta} = 2.43$ $\mathfrak{\bm{\hat{\beta}}} = 2.43$

Hvis man skal skrive mange fede α -er og β -er, er det en fordel at definere sine egne kommandoer dertil; det gøres med kommandoen `\bmdefine{\langle kommando \rangle}{\langle symbol \rangle}`:

`\bmdefine`

Eks. 6.72

For alle $\beta \in \mathbb{R}^n$ er ...`\bmdefine\bbeta{\beta}`For alle $\mathfrak{\bbeta} \in \mathfrak{R}^n$ er ...

Latinske bogstaver der bruges som matematiske symboler, skrives som nævnt næsten altid med matematikkursiv (som f.eks. i $y = f(x)$), og som det ses, giver `\bm` en *fed* matematikkursiv (som i $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$). Hvis man af en eller anden grund skulle få brug for fede oprette bogstaver som matematiksymboler, er løsningen `\mathbf`:

`\mathbf`

Eks. 6.73

 $P \neq \mathbf{P}$ $\mathfrak{\bm{P} \neq \mathbf{P}}$

Vi kan endvidere nævne kommandoen `\mathversion{bold}` der bevirker at *al* matematik bliver sat med fed skrift; den modsatte kommando er `\mathversion{normal}`. (Kommandoerne skal stå i tekst-sammenhæng, altså ikke inde i en formel.)

6.9 Skematiske oversigter

I dette afsnit gives oversigter over en del af de især matematiske symboler der er til rådighed i L^AT_EX (for nærmere oplysninger se Mittelbach m.fl. (2004, afsnit 8.6.2 og 8.9)). Symbolerne er søgt grupperet efter betydning eller i visse tilfælde udseende. En stor del af symbolerne er kun defineret i pakken `amssymb`, hvilket der gøres opmærksom på i oversigterne.

Navngivne operatorer og funktioner

Funktionerne i den venstre spalte har den ekstra egenskab at eventuelle fodtegn (toptegn) kommer til at stå under (over) symbolet i en fremhævet formel.

$\max$	<code>\max</code>	$\log$	<code>\log</code>	$\exp$	<code>\exp</code>
$\min$	<code>\min</code>	$\lg$	<code>\lg</code>	$\ln$	<code>\ln</code>
$\sup$	<code>\sup</code>	$\sin$	<code>\sin</code>	$\cos$	<code>\cos</code>
$\inf$	<code>\inf</code>	$\arcsin$	<code>\arcsin</code>	$\arccos$	<code>\arccos</code>
$\lim$	<code>\lim</code>	$\sinh$	<code>\sinh</code>	$\cosh$	<code>\cosh</code>
$\limsup$	<code>\limsup</code>	$\tan$	<code>\tan</code>	$\cot$	<code>\cot</code>
$\liminf$	<code>\liminf</code>	$\tanh$	<code>\tanh</code>	$\coth$	<code>\coth</code>
$\projlim$	<code>\projlim</code>	$\arctan$	<code>\arctan</code>	$\arg$	<code>\arg</code>
$\injlim$	<code>\injlim</code>	$\sec$	<code>\sec</code>	$\csc$	<code>\csc</code>
$\overline{\lim}$	<code>\varlimsup</code>	$\ker$	<code>\ker</code>	$\dim$	<code>\dim</code>
$\underline{\lim}$	<code>\varliminf</code>	$\hom$	<code>\hom</code>	$\deg$	<code>\deg</code>
$\varprojlim$	<code>\varprojlim</code>				
$\varinjlim$	<code>\varinjlim</code>	Re	<code>\operatorname{Re}</code>	Im	<code>\operatorname{Im}</code>
$\det$	<code>\det</code>	P	<code>\operatorname{P}</code>		
$\gcd$	<code>\gcd</code>				

Læs på side 101 om hvordan man kan definere flere sådanne operatorer og funktioner.

Kommandoerne `\operatorname{Re}`, `\operatorname{Im}` og `\operatorname{Pr}` omdefinieres i IMFUFA-L^AT_EX til de værdier som er vist ovenfor. I Standard L^AT_EX giver `\operatorname{Re}` og `\operatorname{Im}` hhv. $\Re$ and $\Im$, og `\operatorname{Pr}` giver Pr .

Symboler med variabel størrelse

Afhængigt af om de står i en fremhævet formel eller i en tekstformel, ændrer disse symboler størrelse, og eventuelle fod- og toptegn får ændret placering.

$\sum$	<code>\sum</code>	$\biguplus$	<code>\biguplus</code>	$\bigoplus$	<code>\bigoplus</code>
$\prod$	<code>\prod</code>	$\coprod$	<code>\coprod</code>	$\bigotimes$	<code>\bigotimes</code>
$\bigwedge$	<code>\bigwedge</code>	$\bigcap$	<code>\bigcap</code>	$\bigodot$	<code>\bigodot</code>
$\bigvee$	<code>\bigvee</code>	$\bigcup$	<code>\bigcup</code>	$\bigsqcup$	<code>\bigsqcup</code>
$\int$	<code>\int</code>	$\iint$	<code>\iint</code>	$\iiint$	<code>\iiint</code>
$\iiint$	<code>\iiint</code>	$\int \cdots \int$	<code>\int \cdots \int</code>	$\oint$	<code>\oint</code>

De *multiple* integraltegn placerer ikke automatisk fodtegnet nede under integraltegnene, selv i fremhævede formler. Men man kan selv fremtvinge det med kommandoen `\limits`. Eksempelvis giver `\iint_{A} g = 0` resultatet $\iint_A g = 0$, hvorimod `\iint\limits_{A} g = 0` giver resultatet $\iint_A g = 0$.

Parenteser og parentesagtige tegn

Nedenstående tegn kan bruges alene, men de kan også indgå i `\big-`, `\Big-`, osv. kommandoer, og i så fald kan de skifte størrelse, se side 68.

(	(	)	)	[	[	]	]
{	<code>\lbrace</code>	}	<code>\rbrace</code>	<	<code>\langle</code>	>	<code>\rangle</code>
\	<code>\backslash</code>	/	/				<code>\l</code>
	<code>\lvert</code>		<code>\rvert</code>		<code>\lVert</code>		<code>\rVert</code>
⌈	<code>\lceil</code>	⌋	<code>\rceil</code>	⌊	<code>\lfloor</code>	⌋	<code>\rfloor</code>
↕	<code>\updownarrow</code>	↑	<code>\uparrow</code>	↓	<code>\downarrow</code>		
↕	<code>\Updownarrow</code>	↑	<code>\Uparrow</code>	↓	<code>\Downarrow</code>		

I stedet for kommandoerne `\lbrace` og `\rbrace` kan man bruge `\{` og `\}`. I stedet for `[` og `]` kan man bruge `\lbrack` og `\rbrack`.

Plus, minus osv.

+	+	−	−
±	<code>\pm</code>	∓	<code>\mp</code>
×	<code>\times</code>	÷	<code>\div</code>
·	<code>\cdot</code>	/	/
⊙	<code>\odot</code>		

Findes kun i `amssymb`:

⊕	<code>\boxplus</code>	⊖	<code>\boxminus</code>
⊗	<code>\boxtimes</code>	⊙	<code>\boxdot</code>
⊕	<code>\oplus</code>	⊖	<code>\ominus</code>
⊗	<code>\otimes</code>	⊘	<code>\oslash</code>
⋈	<code>\ltimes</code>	⋉	<code>\rtimes</code>
⊕	<code>\dotplus</code>	⋅	<code>\centerdot</code>
⋈	<code>\divideontimes</code>		

Lighed og ulighed

Lighedstegn o.lgn.

=	=	≠	<code>\neq</code>	≡	<code>\equiv</code>
⋮	<code>\doteq</code>				
~	<code>\sim</code>	≈	<code>\simeq</code>	≈	<code>\approx</code>
∝	<code>\propto</code>	⋈	<code>\asymp</code>	≅	<code>\cong</code>

Findes kun i `amssymb`:

⊖	<code>\circeq</code>	△	<code>\triangleq</code>	≡	<code>\eqcirc</code>
⋮	<code>\doteqdot</code>	⋮	<code>\risingdotseq</code>	⋮	<code>\fallingdotseq</code>
∝	<code>\varpropto</code>	⋈	<code>\bumpeq</code>	⋈	<code>\Bumpeq</code>
⋈	<code>\backsim</code>	⋈	<code>\backsimeq</code>	⋈	<code>\eqsim</code>
~	<code>\thicksim</code>	≈	<code>\thickapprox</code>	≈	<code>\approxeq</code>
≈	<code>\nsim</code>	≇	<code>\ncong</code>		

Ulighedstegn

$<$	<code><</code>	$>$	<code>></code>
$\ll$	<code>\ll</code>	$\gg$	<code>\gg</code>
$\leq$	<code>\leq</code>	$\geq$	<code>\geq</code>

Findes kun i amssymb:

$\lessdot$	<code>\lessdot</code>	$\gtrdot$	<code>\gtrdot</code>
$\lll$	<code>\lll</code>	$\ggg$	<code>\ggg</code>
$\leqq$	<code>\leqq</code>	$\geqq$	<code>\geqq</code>
$\leqslant$	<code>\leqslant</code>	$\geqslant$	<code>\geqslant</code>
$\lesseqgtr$	<code>\lesseqgtr</code>	$\gtreqless$	<code>\gtreqless</code>
$\eqslantless$	<code>\eqslantless</code>	$\eqslantgtr$	<code>\eqslantgtr</code>
$\lessapprox$	<code>\lessapprox</code>	$\gtrapprox$	<code>\gtrapprox</code>
$\lesssim$	<code>\lesssim</code>	$\gtrsim$	<code>\gtrsim</code>
$\lvertneqq$	<code>\lvertneqq</code>	$\gvertneqq$	<code>\gvertneqq</code>
$\nleq$	<code>\nleq</code>	$\ngeq$	<code>\ngeq</code>
$\nless$	<code>\nless</code>	$\ngtr$	<code>\ngtr</code>
$\lneqq$	<code>\lneqq</code>	$\gneqq$	<code>\gneqq</code>
$\nleqslant$	<code>\nleqslant</code>	$\ngeqslant$	<code>\ngeqslant</code>
$\lneq$	<code>\lneq</code>	$\gneq$	<code>\gneq</code>
$\nleqq$	<code>\nleqq</code>	$\ngeqq$	<code>\ngeqq</code>
$\lnsim$	<code>\lnsim</code>	$\gnsim$	<code>\gnsim</code>
$\lnapprox$	<code>\lnapprox</code>	$\gnapprox$	<code>\gnapprox</code>
$\lessgtr$	<code>\lessgtr</code>	$\gtrless$	<code>\gtrless</code>
$\lesseqqgtr$	<code>\lesseqqgtr</code>	$\gtreqqless$	<code>\gtreqqless</code>

Krøllede ulighedstegn

$\prec$	<code>\prec</code>	$\succ$	<code>\succ</code>
$\preceq$	<code>\preceq</code>	$\succeq$	<code>\succeq</code>

Findes kun i amssymb:

$\curlyeqprec$	<code>\curlyeqprec</code>	$\curlyeqsucc$	<code>\curlyeqsucc</code>
$\preccurlyeq$	<code>\preccurlyeq</code>	$\succcurlyeq$	<code>\succcurlyeq</code>
$\precapprox$	<code>\precapprox</code>	$\succapprox$	<code>\succapprox</code>
$\nprec$	<code>\nprec</code>	$\nsucc$	<code>\nsucc</code>
$\npreceq$	<code>\npreceq</code>	$\nsucceq$	<code>\nsucceq</code>
$\precnsim$	<code>\precnsim</code>	$\succnsim$	<code>\succnsim</code>
$\precneqq$	<code>\precneqq</code>	$\succneqq$	<code>\succneqq</code>
$\precnapprox$	<code>\precnapprox</code>	$\succnapprox$	<code>\succnapprox</code>

Logik og mængdelære

Delmængde-symboler

$\subset$	<code>\subset</code>	$\supset$	<code>\supset</code>
$\subseteq$	<code>\subseteq</code>	$\supseteq$	<code>\supseteq</code>
$\sqsubseteq$	<code>\sqsubseteq</code>	$\sqsupseteq$	<code>\sqsupseteq</code>
Findes kun i amssymb:			
$\subseteqq$	<code>\subseteqq</code>	$\supseteqq$	<code>\supseteqq</code>
$\Subset$	<code>\Subset</code>	$\Supset$	<code>\Supset</code>
$\sqsubset$	<code>\sqsubset</code>	$\sqsupset$	<code>\sqsupset</code>
$\subsetneq$	<code>\subsetneq</code>	$\supsetneq$	<code>\supsetneq</code>
$\varsubsetneq$	<code>\varsubsetneq</code>	$\varsupsetneq$	<code>\varsupsetneq</code>
$\nsubseteq$	<code>\nsubseteq</code>	$\nsupseteq$	<code>\nsupseteq</code>
$\subsetneqq$	<code>\subsetneqq</code>	$\supsetneqq$	<code>\supsetneqq</code>
$\varsubsetneqq$	<code>\varsubsetneqq</code>	$\varsupsetneqq$	<code>\varsupsetneqq</code>
$\nsubseteqq$	<code>\nsubseteqq</code>	$\nsupseteqq$	<code>\nsupseteqq</code>

Andre mængdelære-symboler

$\in$	<code>\in</code>	$\ni$	<code>\ni</code>
$\cap$	<code>\cap</code>	$\cup$	<code>\cup</code>
		$\uplus$	<code>\uplus</code>
$\setminus$	<code>\setminus</code>	$\emptyset$	<code>\emptyset</code>
$\sqcap$	<code>\sqcap</code>	$\sqcup$	<code>\sqcup</code>
Findes kun i amssymb:			
$\Cap$	<code>\Cap</code>	$\Cup$	<code>\Cup</code>
$\smallsetminus$	<code>\smallsetminus</code>	$\varnothing$	<code>\varnothing</code>
$\complement$	<code>\complement</code>		

Logik

$\forall$	<code>\forall</code>	$\exists$	<code>\exists</code>
$\wedge$	<code>\wedge</code>	$\vee$	<code>\vee</code>
$\neg$	<code>\neg</code>	$\models$	<code>\models</code>
$\vdash$	<code>\vdash</code>	$\dashv$	<code>\dashv</code>
Findes kun i amssymb:			
		$\nexists$	<code>\nexists</code>
$\curlywedge$	<code>\curlywedge</code>	$\curlyvee$	<code>\curlyvee</code>
$\bar{\wedge}$	<code>\bar{\wedge}</code>	$\veebar$	<code>\veebar</code>
$\overline{\wedge}$	<code>\overline{\wedge}</code>	$\nvdash$	<code>\nvdash</code>
$\Vdash$	<code>\Vdash</code>	$\nVdash$	<code>\nVdash</code>
$\vdash$	<code>\vdash</code>	$\nvDash$	<code>\nvDash</code>
$\Vdash$	<code>\Vdash</code>	$\nVDash$	<code>\nVDash</code>
$\therefore$	<code>\therefore</code>	$\because$	<code>\because</code>

Geometri og geometriske symboler

Cirkler og stjerner

$\circ$	<code>\circ</code>	$\bigcirc$	<code>\bigcirc</code>
$\bullet$	<code>\bullet</code>		
$\star$	<code>\star</code>	$*$	<code>\ast</code>
Findes kun i amssymb:			
$\odot$	<code>\circledcirc</code>	$\otimes$	<code>\circledast</code>
$\bigstar$	<code>\bigstar</code>	$\ominus$	<code>\circleddash</code>

Trekanter og firkanter

$\triangle$	<code>\triangle</code>	$\diamond$	<code>\diamond</code>
$\triangleleft$	<code>\triangleleft</code>	$\triangleright$	<code>\triangleright</code>
$\bigtriangleup$	<code>\bigtriangleup</code>	∇	<code>\bigtriangledown</code>
Findes kun i amssymb:			
$\triangle$	<code>\vartriangle</code>	∇	<code>\triangledown</code>
$\triangleleft$	<code>\vartriangleleft</code>	$\triangleright$	<code>\vartriangleright</code>
$\trianglelefteq$	<code>\trianglelefteq</code>	$\trianglerighteq$	<code>\trianglerighteq</code>
$\ntriangleleft$	<code>\ntriangleleft</code>	$\ntriangleright$	<code>\ntriangleright</code>
$\ntrianglelefteq$	<code>\ntrianglelefteq</code>	$\ntrianglerighteq$	<code>\ntrianglerighteq</code>
$\blacktriangle$	<code>\blacktriangle</code>	$\blacktriangledown$	<code>\blacktriangledown</code>
$\blacktriangleleft$	<code>\blacktriangleleft</code>	$\blacktriangleright$	<code>\blacktriangleright</code>
$\square$	<code>\square</code>	$\blacksquare$	<code>\blacksquare</code>
$\lozenge$	<code>\lozenge</code>	$\blacklozenge$	<code>\blacklozenge</code>

Geometri

$\perp$	<code>\perp</code>	$\angle$	<code>\angle</code>
$\parallel$	<code>\parallel</code>	$ $	<code>\mid</code>
Findes kun i amssymb:			
$\sphericalangle$	<code>\sphericalangle</code>	$\sphericalangle$	<code>\sphericalangle</code>
$\between$	<code>\between</code>		
$\shortparallel$	<code>\shortparallel</code>	$\shortmid$	<code>\shortmid</code>
$\nparallel$	<code>\nparallel</code>	$\nmid$	<code>\nmid</code>
$\nshortparallel$	<code>\nshortparallel</code>	$\nshortmid$	<code>\nshortmid</code>
$\leftthreetimes$	<code>\leftthreetimes</code>	$\rightthreetimes$	<code>\rightthreetimes</code>

Pile

	$\mapsto$	<code>\mapsto</code>
	$\longmapsto$	<code>\longmapsto</code>
$\leftarrow$	<code>\leftarrow</code>	<code>\rightarrow</code>
$\Leftarrow$	<code>\Leftarrow</code>	<code>\Rightarrow</code>
$\longleftarrow$	<code>\longrightarrow</code>	<code>\longrightarrow</code>
$\Longleftarrow$	<code>\Longrightarrow</code>	<code>\Longrightarrow</code>
$\leftrightarrow$	<code>\leftrightharpoonup</code>	<code>\longleftrightarrow</code>
$\Leftrightarrow$	<code>\Leftrightarrow</code>	<code>\Longleftrightarrow</code>
$\Rightarrow$	<code>\rightleftharpoons</code>	<code>\iff</code>
$\lleftarrow$	<code>\leftharpoonup</code>	<code>\rightarrow</code>
$\rleftarrow$	<code>\leftharpoonupdown</code>	<code>\rightarrow</code>
$\hookleftarrow$	<code>\hookleftarrow</code>	<code>\hookrightarrow</code>
$\Uparrow$	<code>\uparrow</code>	<code>\downarrow</code>
$\Uparrow$	<code>\Uparrow</code>	<code>\Downarrow</code>
$\Updownarrow$	<code>\updownarrow</code>	<code>\Updownarrow</code>
$\nearrow$	<code>\nearrow</code>	<code>\searrow</code>
$\swarrow$	<code>\swarrow</code>	<code>\nwarrow</code>
Findest kun i amssymb:		
$\leftarrowtail$	<code>\leftarrowtail</code>	$\rightarrowtail$
$\twoheadleftarrow$	<code>\twoheadleftarrow</code>	$\twoheadrightarrow$
$\leftrightsquigarrow$	<code>\leftrightsquigarrow</code>	$\rightsquigarrow$
$\looparrowleft$	<code>\looparrowleft</code>	$\looparrowright$
$\Lsh$	<code>\Lsh</code>	$\Rsh$
$\curvearrowleft$	<code>\curvearrowleft</code>	$\curvearrowright$
$\circlearrowleft$	<code>\circlearrowleft</code>	$\circlearrowright$
$\leftleftarrows$	<code>\leftleftarrows</code>	$\rightrightarrows$
$\upuparrows$	<code>\upuparrows</code>	$\downdownarrows$
$\leftrightarrows$	<code>\leftrightarrows</code>	$\rightleftarrows$
$\leftrightharpoons$	<code>\leftrightharpoons</code>	$\rightleftharpoons$
$\upharpoonright$	<code>\upharpoonright</code>	$\downharpoonright$
$\upharpoonleft$	<code>\upharpoonleft</code>	$\downharpoonleft$
$\nleftarrow$	<code>\nleftarrow</code>	$\nrightarrow$
$\nLeftarrow$	<code>\nLeftarrow</code>	$\nRightarrow$
$\nLeftrightarrow$	<code>\nLeftrightarrow</code>	$\nleftrightarrow$
$\Lleftarrow$	<code>\Lleftarrow</code>	$\Rrightarrow$

Fremmede bogstaver som matematiske symboler

Hebraiske bogstaver

$\aleph$	<code>\aleph</code>				
Findes kun i amssymb:					
$\beth$	<code>\beth</code>	$\gimel$	<code>\gimel</code>	$\daleth$	<code>\daleth</code>

Græske bogstaver til matematikbrug

dansk navn		lille bogstav		stort bogstav
alfa	α	<code>\alpha</code>	A	<code>\mathrm{A}</code>
beta	β	<code>\beta</code>	B	<code>\mathrm{B}</code>
gamma	γ	<code>\gamma</code>	Γ	<code>\Gamma</code>
delta	δ	<code>\delta</code>	Δ	<code>\Delta</code>
epsilon	ε	<code>\eps</code>	E	<code>\mathrm{E}</code>
eller	ϵ	<code>\epsilon</code>		
digamma			F	<code>\digamma</code>
zeta	ζ	<code>\zeta</code>	Z	<code>\mathrm{Z}</code>
eta	η	<code>\eta</code>	H	<code>\mathrm{H}</code>
theta	θ	<code>\theta</code>	Θ	<code>\Theta</code>
eller	ϑ	<code>\vartheta</code>		
iota	ι	<code>\iota</code>	I	<code>\mathrm{I}</code>
kappa	κ	<code>\kappa</code>	K	<code>\mathrm{K}</code>
eller	$\varkappa$	<code>\varkappa</code>		
lambda	λ	<code>\lambda</code>	Λ	<code>\Lambda</code>
my	μ	<code>\mu</code>	M	<code>\mathrm{M}</code>
ny	ν	<code>\nu</code>	N	<code>\mathrm{N}</code>
ksi	ξ	<code>\xi</code>	Ξ	<code>\Xi</code>
omikron	o	<code>o</code>	O	<code>\mathrm{O}</code>
pi	π	<code>\pi</code>	Π	<code>\Pi</code>
eller	ϖ	<code>\varpi</code>		
ro	ρ	<code>\rho</code>	P	<code>\mathrm{P}</code>
eller	ϱ	<code>\varrho</code>		
sigma	σ	<code>\sigma</code>	Σ	<code>\Sigma</code>
tau	τ	<code>\tau</code>	T	<code>\mathrm{T}</code>
ypsilon	υ	<code>\upsilon</code>	Υ	<code>\Upsilon</code>
fi	ϕ	<code>\phi</code>	Φ	<code>\Phi</code>
eller	φ	<code>\varphi</code>		
khi	χ	<code>\chi</code>	X	<code>\mathrm{X}</code>
psi	ψ	<code>\psi</code>	Ψ	<code>\Psi</code>
omega	ω	<code>\omega</code>	Ω	<code>\Omega</code>

Bemærk: Kommandoen `\eps` findes kun i IMFUFA-L^AT_EX. I standard-L^AT_EX hedder den `\varepsilon`. Tegnene F og $\varkappa$ er defineret i amssymb-pakken.

Diverse

Fra dagligdagen

†	<code>\dagger</code>	‡	<code>\ddagger</code>
♠	<code>\spadesuit</code>	♣	<code>\clubsuit</code>
♥	<code>\heartsuit</code>	♦	<code>\diamondsuit</code>
♯	<code>\sharp</code>	♭	<code>\flat</code>
♮	<code>\natural</code>		
€	<code>\euro</code>	‰	<code>\promille</code>
°C	<code>\celsius</code>		

Kommandoerne under linjen er defineret i IMFUFA- $\text{\LaTeX}$ (og de kan i øvrigt også benyttes i tekstsammenhæng).

Blandet

⤴	<code>\top</code>	⊥	<code>\bot</code>	ℓ	<code>\wr</code>
⤵	<code>\frown</code>	⤶	<code>\smile</code>	☞	<code>\bowtie</code>
ℓ	<code>\ell</code>	∅	<code>\wp</code>	ℏ	<code>\hbar</code>
Findes kun i amssymb:					
⤵	<code>\smallfrown</code>	⤶	<code>\smallsmile</code>	℔	<code>\Bbbk</code>
/	<code>\diagup</code>	\	<code>\diagdown</code>	∖	<code>\backprime</code>
⊖	<code>\multimap</code>	⤴	<code>\intercal</code>	⤴	<code>\pitchfork</code>
⊥	<code>\Finv</code>	⊃	<code>\Game</code>	℧	<code>\mho</code>
ε	<code>\backepsilon</code>	ø	<code>\eth</code>	ℏ	<code>\hslash</code>

7 Henvisninger på kryds og tværs

7.1 Henvisninger

Ofte har man brug for at henvide fra et sted i teksten til et andet – og ofte kunne læseren ønske sig at forfatteren havde lavet nogle flere henvisninger end tilfældet er. Der er ingen undskyldninger for ikke at lave mange og gode henvisninger, for det er uhyre let.

Man kan henvide til alt hvad $\LaTeX$ selv nummererer (og man bør overlade til $\LaTeX$ at nummerere alt hvad der skal nummereres): kapitler, afsnit, sidenumre, nummererede underpunkter, fodnoter, figurer, tabeller, formler osv. Fremgangsmåden er hver gang den samme: på det sted der skal henvises *til*, placerer man et mærke, en ‘label’, med kommandoen `\label{<mærke>}`, og på det sted man henviser *fra*, skriver man enten `\ref{<mærke>}` for at henvide til den nummererede struktur hvor mærket er anbragt, eller `\pageref{<mærke>}` for at henvide til den *side* hvor mærket er anbragt. Eksempel:

`\label`
`\ref`
`\pageref`

EKS. 7.1

	<code>\subsection{Numre}\label{sec:nr}</code>
7.1.1 Numre	Dette underafsnit har
Dette underafsnit har nummer 7.1.1 og står på side 87.	nummer <code>\ref{sec:nr}</code> og står på side <code>\pageref{sec:nr}</code> .

I memoir-klassen og dermed også i IMFUFA- $\LaTeX$ -klasserne findes en række ekstra henvisningskommandoer (Wilson, 2004, afsnit 17.18):

<code>\pref{<mærke>}</code>	~	side <code>\pageref{<mærke>}</code>	<code>\pref</code>
<code>\fref{<mærke>}</code>	~	figur <code>\ref{<mærke>}</code>	<code>\fref</code>
<code>\tref{<mærke>}</code>	~	tabel <code>\ref{<mærke>}</code>	<code>\tref</code>
<code>\Sref{<mærke>}</code>	~	afsnit <code>\ref{<mærke>}</code>	<code>\Sref</code>
<code>\Cref{<mærke>}</code>	~	kapitel <code>\ref{<mærke>}</code>	<code>\Cref</code>
<code>\Pref{<mærke>}</code>	~	del <code>\ref{<mærke>}</code>	<code>\Pref</code>

Eksempel:

EKS. 7.2

Se tabel 2.2 på side 13.	Se <code>\tref{tab:accent}</code> på <code>\pref{tab:accent}</code> .
--------------------------	--

(Disse henvisningskommandoer er opmærksomme på hvilket sprog der er valgt, dvs. hvis gældende sprog eksempelvis er engelsk, så vil f.eks. `\pref` automatisk skrive page *xx* i stedet for side *xx*. Se også afsnit 8.2 side 101.)

`\titleref` Man kan også henvide til *titlen* på f.eks. et afsnit; det gøres med `\titleref{⟨mærke⟩}` hvilket illustreres med følgende variant af eks. 7.1:

	<code>\subsection{Numre}\label{sec:nr2}</code>
7.1.2 Flere numre	Dette underafsnit hedder
Dette underafsnit hedder Flere numre og står på side 88.	<code>\textbf{\titleref{sec:nr2}}</code> og står på <code>\pref{sec:nr2}</code> .

Eks. 7.3

Placering af labelen

En henvisning skal jo gerne henvide til det man gerne vil have den til at henvide til, og derfor skal man være lidt opmærksom på hvor `\label`-kommandoen placeres. En god tommelfingerregel er: Placer `\label`-kommandoen *umiddelbart efter* den kommando der fremstiller det nummer der skal henvises til.

Eksempel: Hvis man ønsker at mærket `sec:formaal` skal henvide til afsnittet »Formål«, kan man skrive

<pre>\section{Formål} \label{sec:formaal} I dette afsnit beskriver vi ...</pre>

Eks. 7.4

Det er `\section`-kommandoen der fremstiller afsnitsnummeret (og afsnitsoverskrift osv.). I eks. 7.4 er `\label`-kommandoen korrekt placeret: den kommer *efter* `\section`-kommandoen, nemlig efter den afsluttende `}`, og den kommer *umiddelbart* efter i den forstand at der ikke mellem `\section{Formål}` og `\label{sec:formaal}` står noget der producerer output.

- Ved figurer og tabeller (se afsnit 3.2 side 26) er det `\caption`-kommandoen der fremstiller nummeret. Man skal derfor skrive en `\label`-kommando *umiddelbart efter* `\caption`-kommandoens afsluttende `}`, se f.eks. eks. 3.15 side 27.

Ingen regler uden undtagelser: en label der skal henvide til en *fodnote*, skal placeres inde i argumentet til `\footnote`.

Værd at vide

Et givent `⟨mærke⟩` må naturligvis kun optræde i ét `\label`-kald. Hvis man har brugt det samme `⟨mærke⟩` i flere labels, får man advarsler af formen

LaTeX Warning: Label 'abs' multiply defined.

LaTeX Warning: There were multiply-defined labels.

Hvis man henviser til et ikke-defineret `⟨mærke⟩`, får man en advarsel af formen

LaTeX Warning: Reference 'xyz' on page 15 undefined on input line 300

Man kan også få advarsler om ikke-definerede labels selv om man lige har siddet og tastet de rigtige `\label`-kommandoer ind i sit dokument. I det tilfælde vil advarslerne

forsvinde næste gang man kører teksten gennem (pdf)latex. (Forklaringen er at L^AT_EX altid kun kender de ting der var defineret *forrige* gang.)

Den tekststreng man bruger som $\langle m\grave{a}rke \rangle$, kan i princippet være hvad som helst, sålænge man holder sig fra de mere specielle bogstaver og tegn som f.eks. æ ø å é â ö ½ §. Af hensyn til forfatteren er det dog praktisk at have en vis systematik i navngivningen af mærker, eksempelvis kan man lade dem der henviser til afsnit, begynde med **sec:**, dem der henviser til figurer med **fig:** osv.

Når det hele alligevel trods de gode intentioner på et tidspunkt er ved at ende i kaos, kan man have stor glæde af pakken **showkeys**.

7.2 Stikord og stikordsregister

I visse typer publikationer er det en stor hjælp med et fornuftigt stikordsregister. Man kan sagtens lave stikordsregistre til L^AT_EX-dokumenter, men det er en proces der består af flere trin hvoraf nogle er lidt omstændelige:

1. Man skal beslutte sig for hvilke ord i teksten der skal tilknyttes et stikord. Det gøres af forfatteren eller en anden person.
2. I **tex**-filerne skal man indsætte **\index**-kommandoer i overensstemmelse med vedtagelsen i forrige punkt. Kommandoen **\index** omtales i det følgende.
3. I præamblen skal man indsætte kommandoen **\makeindex** for at fortælle L^AT_EX at der skal laves et stikordsregister; konkret vil der da blive fremstillet en hjælpefil (med efternavn **idx**) hvor der for hver **\index**-kommando kommer en linje med det pågældende stikord og det tilhørende sidenummer.
4. Stikordslisten skal sorteres så stikordene kommer i alfabetisk orden og uden dubletter. Det gøres af et særligt stikordsbehandlingsprogram, f.eks. det klassiske **makeindex** der findes i almindelige L^AT_EX-installationer; hvis stikordene skal sorteres efter danske regler, er programmet **mkind** en mulighed. Under alle omstændigheder bliver resultatet en såkaldt **indexfil** (med efternavnet **ind**).

\makeindex

Programmet **mkind** er et stikordsregisterprogram som er skrevet af Jørgen Larsen, og som sorterer efter danske alfabetiseringsregler. Man kan sige at **mkind** er en delmængde af **makeindex** i den forstand at **\index**-kommandoer skrevet med henblik på det første også forstås (og – med en enkelt undtagelse – forstås på samme måde) af det andet, forudsat at man i **makeindex** eller i **mkind** omdefinierer specialtegnene (idet de to programmer har forskellige defaultværdier). Undtagelsen er at **makeindex** i modsætning til **mkind** opfatter *alle* blanke tegn som betydende.

5. Indexfilen skal inkluderes i dokumentet. Hvis primærfilen hedder **foo.tex**, så kommer indexfilen til at hedde **foo.ind**, og man skal derfor skrive **\input{foo.ind}** i slutningen af primærfilen.

Herefter skal man køre dokumentet igennem (pdf)latex et par gange og kalde stikordsbehandlingsprogrammet efter hver gang således at stikordsregisteret kommer med, også i indholdsfortegnelsen, samt læse korrektur en ekstra gang.

Kommandoen **\index**

Den simpleste form for kald af **\index** er **\index{ $\langle stikord \rangle$ }**, for eksempel

Vi er underkastet åndsfyrsternes\index{åndsfyrste} herredømme.

Eks. 7.5

Man bør så vidt muligt undgå mellemrum mellem \index-kommandoen og det ord den henviser til; ethvert mellemrum rummer jo en mulighed for et sideskift, og i så fald kan der blive henvist til en forkert side.

Som $\langle stikord \rangle$ kan man skrive næsten hvad som helst, dog skal man være opmærksom på at der er visse tegn der fungerer som specialtegn i forhold til stikordsbehandlingsprogrammet. Hvis man bruger mkind, er de fire tegn ! > = | specialtegn med følgende betydning:

- = • Man bruger tegnet = hvis stikordet skal alfabetiseres på en anden måde end det skrives. Syntaksen er \index{<sorteringsnøgle>=<stikord>}.
Eks. 7.6 viser hvad man skal skrive for at få stikordet Aalborg placeret under Å (og i øvrigt umiddelbart efter stikordet Ålborg, hvis der skulle være et sådant).

\index{Ålborg = Aalborg}

Eks. 7.6

I eks. 7.7 ses to andre eksempler; den første linje bevirker at stikordet β placeres i stikordsregisteret som om det var et b, den anden linje at stikordet \beta placeres som om det var stavet beta.

\index{b = \$\beta\$}
\index{beta = \texttt{\textbackslash beta}}

Eks. 7.7

- > • Man bruger tegnet > for at lave underpunkter til et stikord. Syntaksen er \index{<punkt> > <underpunkt>} hhv. \index{<punkt> > <underpunkt> > <underunderpunkt>}. Eksempel:

\index{åndsfyrste > mindre}

Eks. 7.8

Et $\langle punkt \rangle$ eller $\langle underpunkt \rangle$ kan være af formen $\langle sorteringsnøgle \rangle = \langle stikord \rangle$.

- | • Man kan bruge tegnet | for at få sidetallet skrevet med en særlig skrift (for eksempel for at markere at emnet får en særlig udførlig behandling på denne side). I eks. 7.9 bliver sidetallet skrevet med fed:

\index{åndsfyrste |textbf }

Eks. 7.9

Resultatet af eks. 7.10 er at der i den endelige stikordsliste ved »åndsfyrste« kommer en henvisning til »intellektuelt udfordret person«.

\index{åndsfyrste |see intellektuelt udfordret person }

Eks. 7.10

- ! • Man bruger tegnet ! for at ophæve et af specialtegnenes virkning. Eksempel: Vi vil gerne have formlen $E = mc^2$ til at optræde i stikordsregisteret under E. Da = er et specialtegn, kan vi ikke skrive \index{\$E=mc^2\$}, vi er nødt

til at normalisere lighedstegnet ved at sætte et `!` foran: `\index{$E!=mc^2$}`. Dette `\index`-kald resulterer i stikordet $E = mc^2$, det kommer bare ikke til at stå under E, men under \$. Derfor må vi bruge `=` for at fortælle hvordan det skal alfabetiseres; den rigtige løsning er

```
\index{E!=mc^2 = $E!=mc^2$}
```

EKS. 7.11

7.3 Referencer og litteraturhenvisninger

Visse typer skriftlige arbejder skal indeholde en litteraturliste (også kaldet bibliografi eller liste over referencer), dvs. en liste over bøger, artikler mv. som der henvises til forskellige steder i teksten. Hvert fag eller fagområde har sine egne traditioner for hvordan litteraturlisten og henvisningerne skal se ud, og hvert tidsskrift har sine egne særlige regler. Det er derfor en stor fordel at operere med en generel litteraturlistebase *plus* et system der henter de fornødne informationer fra denne database *plus* et sæt »bibliografi-stile« der formatterer dem efter givne retningslinjer. I det følgende omtales hvordan disse ting gøres med `BIBTEX`.

BibTeX-databaser

Litteraturlistebaser organiseres med fordel som `BIBTEX`-databaser. `BIBTEX`-databaser kan være meget store (man kunne f.eks. have en base der omfattede »alt« om matematiske modeller), `LATEX` kan selv finde ud af at uddrage de nødvendige oplysninger til en litteraturliste til en konkret tekst.

En `BIBTEX`-database er en tekstfil (med efternavn `bib`) hvor hver enkelt bog, artikel osv. er opført som en post efter et særligt system. I figur 7.1 ses to eksempler på sådanne poster. For hver post angives først hvilken *publikationstype* der er tale om, i eksemplet er det en `manual` og en `book`; dernæst følger den *nøgle* man skal bruge når man henviser til publikationen, i eksemplet er det hhv. `memman` og `Ellegaard:81`; endelig kommer en liste af felter med *bibliografiske oplysninger*. Den generelle post er af formen

```
@<type>{<nøgle>,
  <felt1> = {<værdi1>},
  <felt2> = {<værdi2>},
  <felt3> = {<værdi3>},
  ...
}
```

- Bemærk at der skal komma mellem de forskellige bibliografiske oplysninger.
- `LATEX`-kommandoer skal omgives af `{ }`.
- I `<nøgle>` må man ikke bruge `æ` `ø` `å` og tilsvarende eksotiske tegn.
- Hvis der er flere forfattere, skriver man **and** mellem de enkelte navne. I mange bibliografi-stile skal litteraturlistens indgange sorteres efter forfatterens efternavn, og det tager `bibtex`-programmet sig af; efternavnet er pr. definition det sidste af forfatterens navne,

```

@MANUAL{memman,
  author={Peter Wilson},
  title={The Memoir Class for Configurable Typesetting. User Guide},
  year=2004,
  url={http://dante.ctan.org/CTAN/macros/latex/contrib/memoir/memman.pdf}
}

@BOOK{Ellegaard:81,
  author = {Erik {Ellegaard Frederiksen}},
  title = {Bogens Funktion og Æstetik},
  year = 1981,
  publisher = {Christian Ejlers' Forlag},
  address = {København}
}

```

Figur 7.1 Bib_T_EX-posterne svarende til Wilson (2004) og Ellegaard Frederiksen (1981).

men hvis man ønsker noget andet, skal man sætte { } om det der skal opfattes som efternavn (f.eks. Ellegaard Frederiksen).

Tabel 7.1 viser de gængse publikationstyper og tilhørende felter til bibliografiske oplysninger.

Kommandoen \bibliography

Man benytter kommandoen \bibliography{\langle *liste af bib-filer* \rangle} til at fortælle L^AT_EX hvilke Bib_T_EX-databaser man ønsker at benytte. Kommandoen skal stå på det sted hvor litteraturlisten skal være.

Eksempel: Hvis man ønsker at benytte bib-filerne `privat.bib` og `model.bib`, skriver man \bibliography{privat,model}

- Man skal *ikke* skrive filernes »efternavn« `.bib`.

Henvisninger og deres udformning

Når man vil henvide til en publikation, kan man gøre det med Standard L^AT_EX-kommandoen \cite{\langle *nøgle(r)* \rangle}. For eksempel henviser man til de bøger der er vist i figur 7.1, sådan her:

Wilson (2004)	\cite{memman}
Ellegaard Frederiksen (1981)	\cite{Ellegaard:81}
Wilson (2004); Ellegaard Frederiksen (1981)	\cite{memman,Ellegaard:81}

EKS. 7.12

Det synlige resultat af \cite afhænger af hvilken bibliografistil man betjener sig af, og af om man har indlæst pakker der omdefinerer \cite; en sådan pakke kunne f.eks. være `natbib` der omtales i det følgende, hvor vi i øvrigt af illustrative grunde også viser eksempler med et værk med flere forfattere, nemlig Mittelbach m.fl. (2004).

Brugen af **natbib**-pakkens **\cite**-kommandoer

I eks. 7.12 ses hvad der kommer ud af **\cite** når man benytter **natbib**-pakken med valgmuligheden **round** og med bibliografistilen **plnatdk**; hvis man benytter andre (eller ingen) pakker, vil resultatet af **\cite** blive anderledes.

I **natbib**-pakken findes forskellige varianter af **\cite**-kommandoen; derved har brugeren mulighed for at tilpasse henvisningens udformning. De to grundlæggende kommandoer er **\citet**[*<før>*][*<efter>*]{*<nøgle(r)>*} der skriver en tekst-agtig henvisning, og **\citep**[*<før>*][*<efter>*]{*<nøgle(r)>*} der skriver henvisningen i parentes:

\citet
\citep

Mittelbach m.fl. (2004)	\citet{companion}
(Mittelbach m.fl., 2004)	\citep{companion}

EKS. 7.13

Hvis man kun angiver ét valgfrit argument, opfattes det (lidt ulogisk) som argumentet *<efter>*, dvs. tekst der skal indsættes efter henvisningen:

Mittelbach m.fl. (2004, side 10)	\citet[side 10]{companion}
(Mittelbach m.fl., 2004, side 10)	\citep[side 10]{companion}

EKS. 7.14

Her bruges begge valgfrie argumenter:

(se Mittelbach m.fl., 2004, side 10)	\citep[se][side 10]{companion}
(se Mittelbach m.fl., 2004)	\citep[se][]{companion}

EKS. 7.15

Kommandoerne findes også i »stjernede« udgaver der skriver alle forfatterne:

Mittelbach, Goossens m.fl. (2004)	\cite*{companion}
Mittelbach, Goossens m.fl. (2004)	\citet*{companion}
(Mittelbach, Goossens m.fl., 2004)	\citep*{companion}

EKS. 7.16

Man kan skrive en liste af citationsnøgler som argument til **\citez**:

Mittelbach m.fl. (2004); Wilson (2004)	\cite{companion,memman}
Mittelbach m.fl. (2004); Wilson (2004)	\citet{companion,memman}
(Mittelbach m.fl., 2004; Wilson, 2004)	\citep{companion,memman}

EKS. 7.17

Man kan henvise til et årstal uden at angive forfatteren eller omvendt:

2004	\citeyear{companion}
(2004)	\citeyearpar{companion}
Mittelbach m.fl.	\citeauthor{companion}
Mittelbach, Goossens m.fl.	\citeauthor*{companion}

EKS. 7.18

Endvidere kan vi nævne kommandoerne **\citealt** og **\citealp** der virker som **\citet** og **\citep**, bortset fra at alle parenteser bliver udeladt. Endelig kan vi gøre opmærksom på kommandoen **\nocite**{*<nøgle>*} hvis (eneste) effekt er at det værk som *<nøgle>*

\nocite

henviser til, kommer med i litteraturlisten; hvis man skriver `\nocite{*}`, så kommer `\nocite{*}` litteraturlisten til at indeholde alt hvad der står i den benyttede BibTeX-database.

Valgmuligheder til natbib

Man indlæser `natbib`-pakken med `\usepackage[⟨options⟩]{natbib}` hvor `⟨options⟩` er en liste (som kan være tom) af valgmuligheder; listens elementer adskilles med `,` (komma). Valgmulighederne bestemmer noget om henvisningernes udseende:

1. `round`, `square`, `curly` og `angle` ændrer parentesernes udseende.
2. `comma` ændrer skilletegnet fra semikolon til komma.
3. `numbers` ændrer henvisningerne fra forfatter-og-år- til [tal]-modellen, f.eks. [3].
4. `super` ændrer henvisningerne fra forfatter-og-år modellen til tal-som-toptegn-modellen, f.eks.³.
5. `sort` bevirker at multiple henvisninger bliver skrevet i samme rækkefølge som i litteraturlisten.
Hvis man har tal-henvisninger kan man bruge `sort&compress` for at få henvisningerne komprimeret (f.eks. 3-5, 7 i stedet for 3, 4, 5, 7).
6. `longnamesfirst` bevirker at første gang en henvisning optræder, kommer alle forfatternavnene med, de følgende gange kommer blot den forkortede liste.
7. `sectionbib` bevirker at litteraturlisten ikke bliver et kapitel men en 'section'.

Litteraturlistens udseende; bibliografistil

Der kan føres endeløse diskussioner om hvordan en litteraturliste skal udformes, og om hvordan henvisningerne inde i teksten skal se ud. Set fra et L^AT_EX-synspunkt bestemmes litteraturlistens udseende af den valgte *bibliografistil*. Bibliografistilen specificeres ved hjælp af kommandoen `\bibliographystyle` der placeres umiddelbart før `\bibliography`-kommandoen. Der findes en lang række bibliografistile; her nævnes nogle af dem der kan benyttes sammen med `natbib`:

- `plainnat` som er en `natbib`-udgave af standard-stilen `plain`.
- `abbrvnat` som er en komprimeret udgave af `plainnat`.
- `unsrtnat` som er en usorteret udgave af `plainnat`.
- `kluwer` fra forlaget Kluwer.
- `agsm` som er Australian Government publications style.
- `dcu` stammer fra Design Computing Unit, Department of Architecture and Design Science, University of Sidney.
- På IMFUFA findes endvidere `plnnatdk` som er en uofficiel dansk oversættelse af `plainnat`.

Nærværende tekst benytter `plnnatdk`.

De nævnte bibliografistile har det til fælles at de er beregnet til engelsksprogede tekster; det kan bl.a. ses på de småord og forkortelser som BibTeX indsætter, ord som for eksempel *and*, *in*, *editor* (eller *ed.*). I tekster på andre sprog end engelsk kan dette se lidt aparte ud, og man vil derfor måske overveje at benytte en bibliografistil der er beregnet til det aktuelle sprog. Der eksisterer bl.a. følgende danske bibliografistile: `dk-abbrev`, `dk-alpha`, `dk-plain` og `dk-unsrtnat`; bemærk at disse *ikke* (p.t.) er kompatible med `natbib`-pakken.

Bemærk at hvis man ændrer bibliografistilen, så skal man køre $\text{\LaTeX}$, $\text{\BibTeX}$ og endnu engang $\text{\LaTeX}$ før man kan se virkningen, jf. næste afsnit.

$\text{\bibfont}$ Man kan ændre den skrifttype som litteraturlisten sættes med, ved at definere $\text{\bibfont}$ til en passende værdi, f.eks. sådan: $\text{\newcommand*{\bibfont}{\normalfont\small}}$

Hvordan gør man?

$\text{\LaTeX}$ og $\text{\BibTeX}$ samarbejder om at hente oplysninger fra bibliografidatabasen:

1. Når man kører $(\text{pdf})\text{latex}$, bliver der i aux -filen skrevet hvilke publikationer der henvises til.
2. Derefter skal man køre programmet bibtex med primærfilen (eller rettere primærfilens fornavn) som argument. Dette program læser aux -filen, finder de nævnte henvisninger i databasen, og skriver de nødvendige bibliografiske data i en ny hjælpefil der har efternavn bbl .
3. Næste gang man kører $(\text{pdf})\text{latex}$, bliver bbl -filen automatisk indlæst og benyttet til at fremstille litteraturlisten.
4. Næste gang igen har $\text{\cite}$ -kommandoerne adgang til den rigtige information, og alt er nu som det skal være (eller i hvert fald som man har bedt om).

Det ser måske besværligt ud, men det er det ikke; og man behøver kun at køre bibtex (altså lave en ny bbl -fil) når man har indført henvisninger til ikke tidligere citerede publikationer og/eller når man har ændret bibliografistilen.

7.4 Hyperlinks med hyperref-pakken

Formålet med $\text{\TeX}$ / $\text{\LaTeX}$ -projektet var oprindeligt fremstilling af smukke *trykte* tekster, men efterhånden er der opstået et delformål der hedder fremstilling af smukke tekster på digital form. I den forbindelse kan man måske ønske sig at indbygge interne og eksterne hyperlinks i sin tekst, og til det formål kan pakken hyperref (Rahtz og Oberdiek, 2004) benyttes. Grundlæggende er det meget enkelt at benytte hyperref , man skriver blot $\text{\usepackage{hyperref}}$ i præamblen, så går resten af sig selv. Det der »går af sig selv« er at alle dokumentets interne krydshenvisninger bliver til klikbare hyperlinks, og det samme gælder url-angivelser med kommandoen $\text{\url}$, se eks. 7.20.

$\text{\url}$

Den færdige tekst får – naturligvis – præcis samme udseende som hvis man ikke bruger hyperref , eksempelvis bliver siderne i A4-format. Hvis man ønsker et mere skærm-egnet layout, kan man bruge ebook som valgmulighed i dokumentklasseerklæringen.

- hyperref -pakken skal indlæses som noget af det allersidste i præamblen, dvs. efter indlæsning af eventuelle andre pakker og efter brugerens egne definitioner.
- Når man bruger hyperref sammen med memoir eller $\text{IMFUF}\text{\LaTeX}$, skal man efter hyperref inkludere endnu en pakke, nemlig memhfixc (se eks. 7.19).

Man kan indstille adskillige af de parametre der styrer hyperref -pakkens opførsel. Det gøres mest overskueligt ved brug af $\text{\hypersetup}$ -kommandoen, f.eks. sådan (eksemplet viser hvordan hyperref aktiveres i nærværende tekst):


```

\usepackage{hyperref}
\hypersetup{
  colorlinks=true,
  bookmarks=true,
  pdftitle={LaTeX for forfattere},
  pdfauthor={J\o rgen Larsen},
  pdfsubject={introduktion til LaTeX},
  pdfkeywords={LaTeX, IMFUFA-LaTeX}
}
\usepackage{memhfixc}

```

EKS. 7.19

Her bestemmer `colorlinks` (der kan være `true` eller `false`) om et link skal markeres med farve eller ved en indramning, og `bookmarks` bestemmer om vieweren skal vise en liste med »bogmærker« eller ej. – Brugeren kan justere talrige andre størrelser, blandt andet farverne på de forskellige typer af links, se nærmere i Rahtz og Oberdiek (2004).

Man kan endvidere indsætte links til eksterne dokumenter; når brugeren klikker på linket, vil hans/hendes webbrowser straks vise den pågældende side (forudsat at brugerens computer er sat rigtigt op!). Her er et par simple eksempler:

Hjemmesiden for TeX User Group er <code>http://tug.daimi.au.dk/</code>	Hjemmesiden for TeX User Group er <code>\url{http://tug.daimi.au.dk/}</code>
--	---

EKS. 7.20

Kommandoen `\url` defineres i pakken `url` og også i `hyperref`-pakken.

Hjemmesiden for TUG.	Hjemmesiden for <code>\href{http://tug.daimi.au.dk/}{TUG}</code> .
----------------------	---

EKS. 7.21

Hyperlinks er ikke bundet til pdf-formatet, man kan sagtens lave dvi-filer med hyperlinks, og nyere dvi-viewere kan også vise dem! (Visse faciliteter, f.eks. »bogmærker«, findes dog kun i pdf-formatet.

Mulige problemer

Pakken `hyperref` omdefinerer en stor mængde interne $\text{\LaTeX}$ -kommandoer; i de fleste tilfælde lykkes det uden at brugeren behøver ændre noget i sine `tex`-filer, men en gang imellem kommer der uventede fejlmeddelelser. (Den første gang man kompilerer et dokument efter at have sat det til at bruge `hyperref`, kommer der næsten altid uventede meddelelser og advarsler, men de skulle gerne forsvinde hurtigt.)

En mulig problemkilde er bogmærke-funktionen; denne henter sine informationer fra argumenterne til `\chapter`-, `\section`-, `\subsection`- osv. kommandoerne, og der kan undertiden opstå problemer hvis disse argumenter eksempelvis indeholder matematik.

Eksempel: Antag at man vil skrive om formelen $e^{i\pi} = -1$ og derfor skriver afsnitskommandoen `\section{Formlen  $e^{i\pi} = -1$ }` – man får i øvrigt et noget pænere resultat med `\section{Formlen  $\text{\bm{e}}^{i\pi} = -1$ }`, men uanset om man bruger den ene eller den anden, kommer der problemer i form af advarsler og/eller fatale fejl. Problemet er at `hyperref` ikke kan klare at omforme matematikken til noget der kan

`\texorpdfstring`

bruges i pdf-bogmærkerne. Løsningen på problemet består i at anvende kommandoen `\texorpdfstring{⟨tex-kode⟩}{⟨pdf-kode⟩}` hvor `⟨tex-kode⟩` er den ønskede L^AT_EX-kode, og `⟨pdf-kode⟩` er pdf-kode der bruges til bogmærkerne (hvis man ikke kender noget til pdf-kode, må man klare sig med almindelig tekst); `\texorpdfstring` sørger så for at sende den rigtige kode de rigtige steder hen. I eks. 7.22 vises to brugbare løsninger (hvoraf den anden giver det pæneste resultat).

```
\section{Formlen \texorpdfstring{$e^{i\pi}=-1$}{exp(i pi)=-1}}

\section{Formlen \texorpdfstring{$\bm{e^{i\pi}=-1}$}{exp(i pi)=-1}}
```

Eks. 7.22

Kommandoen `\texorpdfstring` er defineret i `hyperref`-pakken.

- Hvis man har skrevet en tekst som indeholder `hyperref`-definerede kommandoer, så kan man meget let kompilere den som en almindelig tekst uden hyper-referencer: man skal blot benytte pakken `nohyperref` i stedet for `hyperref`.

8 Videregående emner

8.1 Om at definere L^AT_EX-kommandoer

Brugeren kan – for egen regning og risiko – definere egne kommandoer og omdefinere allerede eksisterende kommandoer. Kommandoer defineres med `\newcommand*` og omdefineres med `\renewcommand*`. Definitionerne placeres normalt i præamblen (fordi de skal gælde i hele dokumentet), men de kan faktisk stå hvor som helst.

Der findes også kommandoerne `\newcommand` og `\renewcommand` der fungerer stort set som `\newcommand*` og `\renewcommand*`. Forskellen er at hvis en kommando, lad os sige kommandoen `\hest`, er defineret/omdefineret med `\newcommand*/\renewcommand*`, så må et eventuelt argument til `\hest` ikke indeholde blanke linjer eller andre linjeskiftkommandoer (som f.eks. `\par`). Derudover bruger de ustjernede versioner en smule mere af latex-programmets interne hukommelse.

Kommandoen `\newcommand*{<navn>}[<narg>]{<definition>}` definerer kommandoen `<navn>` der har `<narg>` argumenter (`<narg>` er et tal mellem 0 og 9). I definitionsdelen skal man bruge `#1`, `#2`, osv. som pladsholdere for argument nr. 1, 2, osv. Hvis kommandoen ikke har nogen argumenter, kan man udelade `[<narg>]`. Syntaksen for `\renewcommand*` er fuldstændig tilsvarende.

`\newcommand*`

`\renewcommand*`

- Et kommandonavn skal bestå af en backslash efterfulgt af et eller flere engelske bogstaver, dvs. `abc...z` og `ABC...Z` (der er forskel på store og små bogstaver). Det er ikke tilladt at bruge cifre eller f.eks. `æøå`.

Eksempler på brugen af `\newcommand*` og `\renewcommand*`:

1. Hvis man skriver om IMFUFA-L^AT_EX og vil undgå at skulle skrive IMFUFA-`\LaTeX` et hav af gange, kan det måske være hensigtsmæssigt at definere en kort kommando der gør det:

IMFUFA-L ^A T _E X's fortræffeligheder	<code>\newcommand*{\iltx}{IMFUFA-<code>\LaTeX</code>}</code>
	<code>\iltx's fortræffeligheder</code>

EKS. 8.1

2. Som læseren måske har bemærket, er der i nærværende tekst benyttet et system med at navne på L^AT_EX-pakker bliver skrevet med en særlig skrift (sans serif), f.eks. `bm`. Dette kunne bringes i værk ved at forfatteren huskede at skrive alle pakkenavne i `\textsf{ }`, men hvis forfatteren pludselig ændrer mening og synes at pakkenavne skal skrives med fed skrift, så ville det være temmelig besværligt at skulle ændre hele tex-filen. Det er langt mere hensigtsmæssigt at definere en kommando (der f.eks. kunne hedde `\pakke`) som man derefter benytter hver gang

man skal skrive et pakkenavn; på den måde skal man i givet fald kun ændre ét sted, nemlig i definitionen af kommandoen. I eks. 8.2 ses et forslag til hvordan kunne definere `\pakke`. Kommandoen defineres til at have ét argument, og definitionsdelen går ud på at dette argument puttes ind i `\textsf{ }`; det betyder at når man f.eks. skriver `\pakke{bm}`, så »ekspanderer« $\text{\LaTeX}$ det til `\textsf{bm}`.

Pakken <code>bm</code> til Bold Math.	<pre>\newcommand*\pakke[1]{\textsf{#1}}</pre>
Pakken <code>\pakke{bm}</code> til Bold Math.	

Eks. 8.2

3. Antag at man i forrige punkt gerne ville have at pakker blev optaget i stikordsregisteret. Det ville være smart hvis man kunne få `\pakke`-kommandoen til at sørge for det. Det kan gøres ved at indføje `\index{#1}` i definitionsdelen, men hvis vi gerne vil have at pakkenavnet også i stikordsregisteret skrives med sans serif, så kan vi gøre sådan (jf. afsnit 7.2 side 89):

<pre>\newcommand*\pakke[1]{\textsf{#1}\index{#1=\textsf{#1}}}</pre>

Eks. 8.3

I nærværende dokument er benyttet en `\pakke`-kommando defineret sådan her:

<pre>\newcommand*\pakke[1]{\textsf{#1}\index{#1=\textsf{#1} (pakke)}}</pre>

Eks. 8.4

4. Når man henviser til nummererede matematiske formler, skriver man traditionelt formlens nummer i parentes og på en sådan måde at nummeret i henvisningen fremtræder på samme måde som i formlen; eksempel: (6.5). Imidlertid er resultatet af $\text{\LaTeX}$'s `\ref`-kommando et nummer *uden* parenteser, sådan: 6.5. Hvis man vil slippe for selv at skulle (huske at) skrive parenteserne, kan man definere en særlig kommando `\eref` som man så altid bruger når man henviser til en formel. Kommandoen kan defineres på følgende måde:

`\eref`

<pre>\newcommand*\eref[1]{(\ref{#1})}</pre>

Eks. 8.5

5. Som eksempel på definition af en kommando med to argumenter viser vi definitionen af `\lfrac` ('long fraction') til brøker hvor brøkstregen er lidt længere end ellers.

$\frac{1}{\sqrt{a^2+b^2}} \text{ er bedre end } \frac{1}{\sqrt{a^2+b^2}}$	<pre>\newcommand*\lfrac[2]{\frac{\, #1 \,}{\, #2 \,}} \[\lfrac{1}{\sqrt{a^2+b^2}} \text{ er bedre end } \frac{1}{\sqrt{a^2+b^2}} \]</pre>
---	--

Eks. 8.6

6. Man kan *omdefinere* en eksisterende kommando, for eksempel for at give den et andet typografisk indhold. Her vises hvordan man kan omdefinere kommandoen for de reelle tal (jf. eks. 6.32 side 67) så den resulterer i $\mathbb{R}$ i stedet for $\mathbb{R}$:

Eks. 8.7

```
\renewcommand*{\R}{\mathsf{R}}
```

7. I `itemize`-konstruktioner bliver punkterne i niveau 1 markeret med en `•`. Hvis man generelt vil omdefinere det symbol som »punkter« markeres ved, skal man omdefinere kommandoen `\labelitemi`. Eksempel:

Eks. 8.8

¶ Et punkt	<code>\renewcommand*{\labelitemi}{\P}</code>
¶ Et andet punkt	<code>\begin{itemize}</code>
	<code>\item Et punkt</code>
	<code>\item Et andet punkt</code>
	<code>\end{itemize}</code>

De kommandoer der bestemmer »punkternes« udseende på niveau 2, 3 og 4 hedder `\labelitemii`, `\labelitemiii` og `\labelitemiv`.

Brugerdefinerede kommandoer har normalt kun obligatoriske argumenter, dog er det muligt at definere kommandoer med ét valgfrit argument (og et antal obligatoriske). Syntaksen er i dette tilfælde `\newcommand*{\langle navn \rangle}[\langle narg \rangle][\langle default \rangle]{\langle definition \rangle}`. Det valgfrie argument (hvis defaultværdi er `\langle default \rangle`) er altid #1. Som eksempel vises definitionen af kommandoen `\ntup` (se eks. 6.45 og 6.46 side 70): `\newcommand*{\ntup}[3][,]{\ensuremath{\langle \#2 \rangle_{\langle \#1 \rangle \langle \#2 \rangle_{\langle \#2 \rangle_{\dots \langle \#2 \rangle_{\langle \#3 \rangle}}}}`

8.2 De indbyggede tekstkonstanter

Der er en række indbyggede tekstkonstanter der indeholder navne som $\LaTeX$ benytter i forskellige sammenhænge. Eksempelvis véd $\LaTeX$ øjensynligt at når man har valgt dansk som gældende sprog, så hedder figurer Figur, indholdsfortegnelser Indhold, stikordsregistre Stikord osv., og hvis man skifter til engelsk, så kommer de samme ting til at hedde Figure, Contents og Index. – Dette administreres af `babel`-pakken, der automatisk indlæses af `IMFUFA- $\LaTeX$` .

Men hvad nu hvis man synes at nogle af de indbyggede tekstkonstanter skal have andre værdier, hvis f.eks. indholdsfortegnelsen (på dansk) skal hedde Indholdsfortegnelse i stedet for Indhold? Løsningen er at omdefinere den relevante tekstkonstant – forudsat at man ved hvad den hedder. I tabel 8.1 ses de vigtigste tekstkonstanter samt deres standardværdier i `IMFUFA- $\LaTeX$` ; de omdefineres med `\renewcommand*`, f.eks.

Eks. 8.9

```
\renewcommand*{\contentsname}{Indholdsfortegnelse}
```

- Tekstkonstanterne er sprogafhængige. Derfor skal man omdefinere dem i en sammenhæng hvor det er det rigtige sprog der er gældende. Normalt skal man gøre det umiddelbart efter `\begin{document}` (eller umiddelbart efter en `\selectlanguage`-kommando).

8.3 Operatornavne i matematik

$\LaTeX$ har kommandoer til en række navne på matematiske operatorer og funktioner (jf. side 78), og man kan definere nye med `\DeclareMathOperator{\langle navn \rangle}{\langle tekst \rangle}`. Som

Tabel 8.1 Tekstkonstanter i IMFUFA- $\LaTeX$. De konstanter der har et navn af formen $\backslash xxxrefname$, benyttes af kommandoerne $\backslash pref$, $\backslash fref$, $\backslash tref$, $\backslash Sref$, $\backslash Cref$ og $\backslash Pref$, jf. side 87.

$\backslash abstractname$	Resumé	$\backslash proofname$	Bevis
$\backslash alsoname$	Se også	$\backslash refname$	Referencer
$\backslash appendixname$	Appendiks	$\backslash seenname$	Se
$\backslash appendixpagename$	Appendiks	$\backslash chaptername$	Kapitel
$\backslash bibname$	Referencer	$\backslash chapterrefname$	kapitel
$\backslash boxname$	Boks	$\backslash figurename$	Figur
$\backslash ccname$	Kopi til	$\backslash figurerefname$	figur
$\backslash contentsname$	Indhold	$\backslash pagename$	Side
$\backslash enclname$	Vedlagt	$\backslash pagerefname$	side
$\backslash glossaryname$	Gloseliste	$\backslash partname$	Del
$\backslash headtoname$	Til	$\backslash partrefname$	del
$\backslash indexname$	Stikord	$\backslash sectionrefname$	afsnit
$\backslash listboxname$	Bokse	$\backslash tablename$	Tabel
$\backslash listfigurename$	Figurer	$\backslash tablerefname$	tabel
$\backslash listtablename$	Tabeller	$\backslash theoremname$	Sætning
$\backslash ofname$	af	$\backslash theoremrefname$	sætning
$\backslash prefacename$	Forord		

$\backslash DeclareMathOperator$ eksempel vises definitionen af funktionen $\backslash sgn$ der optræder i eks. 6.51 side 71:

```
\DeclareMathOperator{\sgn}{sgn}
```

Eks. 8.10

$\backslash DeclareMathOperator*$ Hvis man i stedet benytter den stjernede variant $\backslash DeclareMathOperator*$, så får den definerede kommando den egenskab at placeringen af eventuelle fodtegn og/eller toptegn afhænger af om det foregår i en tekstlinjeformel eller en fremhævet formel. Som eksempel viser vi hvordan man kan definere kommandoen $\backslash plim$ (konvergens i sandsynlighed):

Der gælder at	$\backslash DeclareMathOperator*{\plim}{\plim}$
$\plim_{n \rightarrow \infty} X_n = 0$	Der gælder at
og at $\plim_{n \rightarrow \infty} Z_n = 0$.	$\backslash [\backslash plim_{n \rightarrow \infty} X_n = 0 \backslash$
	og at
	$\backslash \$\backslash plim_{n \rightarrow \infty} Z_n = 0 \$.$

Eks. 8.11

$\backslash DeclareMathOperator$ og $\backslash DeclareMathOperator*$ er defineret i amsmath -pakken. – Bemærk i øvrigt at de to kommandoer kun kan anvendes i præamblen.

8.4 Længder og tællere

Længdeangivelser

I typografisammenhæng opererer man med særlige længdeenheder, bl.a. enheden punkt. I $\LaTeX$ -sammenhæng benyttes det amerikanske punkt, pt , hvor $1 \text{ pt} = 0.351 \text{ mm} = 1/72.27 \text{ inch}$, så når skriftgraden hævdes at være 10 punkt, er det retteligen 10 pt. (Det

europæiske punkt, Didôt-punktet, **dd**, er givet ved $1\text{ dd} = 0.376\text{ mm} = \frac{1}{72}\text{ fransk tomme}$. I PostScript-sammenhæng benyttes enheden **bp** (big point) givet ved $1\text{ bp} = 0.351\text{ mm} = \frac{1}{72}\text{ inch}$.

To andre typografiske enheder er **em** og **ex**; 1 **em** (også kaldet en geviert) er bredden af et M i den aktuelle font, 1 **ex** (også kaldet x-højden) er højden af et x i den aktuelle font.

L^AT_EX kender ovennævnte enheder samt de almindelige længdeenheder **in** (inch), **mm** og **cm**.

Når man i L^AT_EX-sammenhæng skal angive en værdi som er en længde (eller bredde eller højde) – f.eks. når man skal angive bredden af en `minipage` eller en `\makebox` –, kan det gøres ved brug af ovennævnte længdeenheder; tilladte længder er f.eks. `3em`, `7mm`, `2.6cm`, `21pt`, `0pt`. Bemærk at længder altid er benævnte tal.

En længde kan endvidere angives som et multiplum af en længde-kommando, f.eks. `0.62\textwidth` (`\textwidth` er en Standard L^AT_EX-kommando som returnerer bredden af tekstblokken).

Tællere

Som det er fremgået, nummererer L^AT_EX en lang række ting: sider, formler, kapitler, fodnoter osv. Det foregår ved hjælp af nogle interne *tællere* ('counters'); disse har typisk det samme (engelske) navn som det der tælles: `page`, `equation`, `chapter`, `footnote` osv. Som bekendt er et tal ikke det samme som den måde tallet skrives på, heller ikke i L^AT_EX; der findes kommandoer på brugerniveau der skriver værdien af en tæller enten på den sædvanlige måde (med arabiske tal) eller som store eller små romertal eller som store eller små bogstaver:

EKS. 8.12

Vi er i Kapitel 8.	<code>Vi er i Kapitel \arabic{chapter}.</code>
Vi er i Kapitel VIII.	<code>Vi er i Kapitel \Roman{chapter}.</code>
Vi er i Kapitel viii.	<code>Vi er i Kapitel \roman{chapter}.</code>
Vi er i Kapitel H.	<code>Vi er i Kapitel \Alph{chapter}.</code>
Vi er i Kapitel h.	<code>Vi er i Kapitel \alph{chapter}.</code>

Til hver tæller hører en intern kommando der leverer tællerens såkaldte *ref-værdi*, dvs. den værdi der faktisk skrives, og som man får som resultat når man henviser (med `\ref`) til det som tælleren tæller. Disse kommandoer hedder `\the...` hvor der på prikkernes plads står navnet på tælleren, f.eks. `\thechapter`, `\thepage` og `\thefootnote`.

EKS. 8.13

Dette står på side 103.	<code>Dette står på side \thepage.</code>
-------------------------	---

Hvis man ønsker tællerens værdi skrevet på en anden måde, så er fremgangsmåden den at man omdefinerer `\the...`-kommandoen. Hvis man for eksempel ønsker kapitlenumre skrevet med store romertal, så skal man skrive man sådan i præamblen:

EKS. 8.14

<code>\renewcommand*\thechapter{\Roman{chapter}}</code>

Man kan få fodnoter »nummereret« med de særlige fodnotesymboler * † ‡ § ¶ osv. ved at skrive

```
\renewcommand*{\thefootnote}{\fnsymbol{footnote}}
```

Eks. 8.15

Visse objekter nummereres fortløbende, det gælder for eksempel sider og kapitler. Andre objekter, for eksempel fodnoter, nummereres fortløbende inden for kapitler. Atter andre objekter, for eksempel formler, figurer og tabeller (og de indrammede eksempler her i disse noter), nummereres fortløbende inden for kapitler, men deres ref-værdi er af formen $\langle kapitelnr \rangle . \langle objektnr \rangle$. Eksempelvis er `\thefigure` defineret på en måde der svarer til

```
\newcommand*{\thefigure}{\thechapter.\arabic{figure}}
```

Eks. 8.16

Det betyder at hvis man omdefinerer `\thechapter` til romertal, så nedarves det automatisk til `\thefigure` således at f.eks. den fjerde figur i det tredje kapitel får ref-værdien III.4.

I enkelte tilfælde kan man have brug for at gribe ind i den automatiske nummerering. Man kan selv sætte værdien af en tæller med `\setcounter{tæller}{værdi}`, for eksempel vil `\setcounter{chapter}{140}` sætte værdien af `chapter`-tælleren til 140.

`\setcounter`

8.5 Verbatim tekst

Læseren har måske spekuleret over hvordan man egentlig bærer sig ad med at skrive den tekst der viser L^AT_EX-kommandoerne. Svaret er at det gøres med `\verb`- og `\verb*`-kommandoerne og med `verbatim`- og `verbatim*`-konstruktionerne.

Man kan bruge `\verb` og `\verb*` til kortere ting der skal stå inde i teksten:

Sære tegn: 230\ ?% # abc	Sære tegn: \verb 230\ ?% # abc
Sære tegn: 230_\?%#_abc	Sære tegn: \verb* 230_\?%#_abc

Eks. 8.17

`\verb` Kommandoen `\verb` virker på følgende måde: Først ser den hvad det er for et tegn der står umiddelbart efter kommandoen; lad os kalde dette tegn for *hegn-tegnet* (fordi det indhegner den tegnsekvens der skal gengives verbatim). I eksemplet er hegn-tegnet |, men det kan være stort set hvad som helst, dog ikke et bogstav eller en *. Derefter sker der en nøjagtig kopiering af alle de efterfølgende tegn, sluttende umiddelbart før den næste forekomst af hegn-tegnet. Kommandoen `\verb*` virker på samme måde, blot skriver den mellemrum som _.

`\verb*`

- Der må ikke være linjeskift i argumentet til `\verb` eller `\verb*`. Bemærk også at `\verb` og `\verb*` ikke må optræde inde i argumentet til en anden kommando (de kan således ikke bruges i fodnoter eller i kapitel- eller afsnitsoverskrifter).

Hvis man har brug for at skrive `\verb` mange gange, kan man eventuelt benytte kommandoen `\MakeShortVerb{<hegn-tegn>}`:

`\MakeShortVerb`

Eks. 8.18

Kommandoen "- har en særlig betydning. Kender du den?	<pre>\MakeShortVerb{\?} Kommandoen "?-? har en særlig betydning. \DeleteShortVerb{\?} Kender du den?</pre>
--	--

I eksemplet udnævnes tegnet ? til generelt hegn-tegn sådan at man kan skrive ?*<tegn>*? i stedet for f.eks. `\verb|<tegn>|`. Dette udnyttes til at skrive "-. Derefter benyttes `\DeleteShortVerb` til at ophæve den særlige betydning af ? som så bruges i sin normale betydning. – Et generelt hegn-tegn skal helst være et tegn der ikke bruges ret meget i den almindelig tekst, så ? er måske et dårligt valg.

`\DeleteShortVerb`

Tekst mellem `\begin{verbatim}` og `\end{verbatim}` kopieres uændret; hvis man erstatte `verbatim` med `verbatim*`, bliver mellemrum gengivet som `␣`.

`verbatim`
`verbatim*`

Eks. 8.19

Her er tekst. <pre>f"!?* =&\ \endinput abcøå</pre>	Her er tekst. <pre>\begin{verbatim} f"!?* =&\ \endinput abcøå \end{verbatim}</pre>
---	---

Sådanne konstruktioner må ikke optræde i argumentet til en anden kommando.

Normalt skrives `verbatim`-tekst med `\ttfamily`-skriften og i en størrelse der passer til omgivelserne. Med kommandoen `\setverbatimfont{<fontklæring>}` kan brugeren vælge en anden skrift til `verbatim`-tekst, f.eks. `\setverbatimfont{\small\ttfamily}` eller `\setverbatimfont{\sfamily}`.

I afsnittene A.1 og A.2 (side 107 og side 108) vises indholdet af to filer; dette foregår ved hjælp af `memoir`-kommandoen `\verbatiminput{<filnavn>}` der indsætter indholdet af filen *<filnavn>* i en `verbatim`-konstruktion.

`\verbatiminput`

Til listning af computerprogrammer og dele deraf kan man ofte have glæde af faciliteterne i `listings`-pakken (Heinz, 2004).

8.6 »Hvis De vil vide mere ...«

Hovedreferencen til $\text{\LaTeX}$ er Lamport (1994). En omfattende håndbog er Mittelbach m.fl. (2004) der behandler mange $\text{\LaTeX}$ -pakker, men udviklingen overhaler hurtigt de trykte medier, så undertiden er det klogt at konsultere online-dokumentationen. Brugere af `IMFUFA- $\text{\LaTeX}$` eller `memoir`-klassen vil før eller senere have stort udbytte af at konsultere Wilson (2004). Endelig bør alle $\text{\LaTeX}$ -brugere kende `TeX Users Group` (TUG) fra hvis hjemmeside <http://tug.daimi.au.dk/> man kan finde frem til mange nyttige oplysninger, og danske brugere bør vel også kende `DK-TUG`, <http://www.tug.dk/>.

Næsten al $\text{\LaTeX}$ -software findes på CTAN ('Comprehensive TeX Archive Network'), <http://dante.ctan.org/CTAN/>.

A Et eksempel

A.1 Primærfilen

Som et eksempel fra det virkelige liv viser vi her den (uretoucherede) primærfil der benyttet til nærværende publikation (se også afsnit A.3).

```
\documentclass[amsmath,pplleading,ngerman,english,danish]{imfreport}

\usepackage[round]{natbib} % natbib, med runde parenteser
\usepackage{graphicx}      % definerer bl.a. \includegraphics
\usepackage{pict2e}        % forbedret picture-environment
\usepackage{imftekst}      % IMFUFAtekst-forside
\usepackage{mflogo}        % Metafont og Metapost logoer
\usepackage{pifont}        % interface til Zapf Dingbats
\usepackage{url}           % definerer kommandoen \url
\usepackage{amssymb}       % ekstra matematiske symboler
\usepackage{mathrsfs}      % definerer \mathscr
\usepackage{bm}            % bold math
\usepackage{microtype}     % mikrotypografiske justeringer
\usepackage{ltxnoter}      % ltxnoter.sty - egne definitioner

\title{Den nye {\LaTeX} for forfattere\\ \
      {\normalsize En introduktion til {\LaTeX} og \ilatex}}

\author{Jørgen Larsen}
\date{Juni 2005}

%% Én af de to næste linjer skal udkommenteres
% \usepackage{hyperref}
\usepackage{nohyperref}
\hypersetup{
%   bookmarks=true,
%   colorlinks=true,
%   pdftitle={LaTeX for forfattere},
%   pdfauthor={J\o rgen Larsen},
%   pdfsubject={introduktion til LaTeX},
%   pdfkeywords={LaTeX, IMFUFA-LaTeX}
}
\usepackage{memhfixc} % aht. kombinationen memoir + hyperref
```

```

\makeindex

%% Her slutter præamblen

\begin{document}

\input{iforside.tex} % IMFUFA-tekst-forside

\cleartorecto      % gå til førstkommende højreside
\tableofcontents*

\input{ch-forord.tex}
\input{ch-first.tex}
\input{ch-next.tex}
\input{ch-boxed.tex}
\input{ch-nice.tex}
\input{ch-math.tex}
\input{ch-hrefs.tex} % \ref, \label, \index, bibliografi
\input{ch-advc.tex} % Videregående emner

\appendix
\input{ap-lists.tex} % listning af primærfil mm.

% \bibliographystyle{plainnat}
\bibliographystyle{plnnatdk}
\bibliography{typobibl} % typobibl.bib

\cleartorecto
\input{ltxnoter.ind} % stikordsregisteret

\end{document}

```

A.2 Forsiden

Her er den fil der indeholder L^AT_EX-koden til IMFUFA-tekst-forsiden:

```

\begin{imfufaforside}[Den nye {\LaTeX} for forfattere.
  En introduktion til {\LaTeX} og {\ilatex}.]{445}{2005}

  {\LaTeX} er et edb-værktøj der er velegnet for brugere der måske ikke
  besidder den store indsigt i typografiens mysterier, men alligevel ønsker
  at fremstille pæne, velafbalancerede tekster hvor layoutet medvirker til
  formidlingen af indholdet. Nærværende skrift giver en kort introduktion
  til {\LaTeX} og den overbygning hertil, {\ilatex}, som er udviklet
  ved IMFUFA. Fremstillingen er blevet til på baggrund af forfatterens
  mangeårige virke som IMFUFAs {\LaTeX}-vedligeholder og -vejleder, og den
  indeholder svar på mange af de utallige {\LaTeX}-spørgsmål han har fået

```

stillet i tidens løb, og også på nogle af de spørgsmål der burde være stillet. Værket henvender sig især til personer der har brug for at fremstille en almindelig tekst skrevet med det latinske alfabet og muligvis med en større eller mindre portion matematiske formler. Derimod står der ikke noget om hvordan man skriver kemiske formler, noder, poesi, tekster med flere forskellige alfabeter osv. osv. Der står heller ikke noget om hvordan man fremstiller billeder eller grafiske fremstillinger på $\LaTeX$ -egnede måder -- det ville kræve et særskilt hæfte (for ikke at sige bog).

`\textit{Den nye  $\LaTeX$  for forfattere}` er en stærkt revideret udgave af `\textit{ $\LaTeX$  for forfattere}` (IMFUFA-tekst nr. 366), blandt andet er matematikafsnittet fuldstændig ændret, og den øvrige tekst er søgt bragt i overensstemmelse med vore dages udbud og funktionalitet af de forskellige dokumentklasser og pakker.

`\bpar`

`{\ilatex}` er et sæt dokumentklasser der er udviklet som supplement til Standard $\LaTeX$, oprindeligt med det formål at gøre livet lidt lettere for danskskrivende $\LaTeX$ -brugere. `{\ilatex}` kan findes på nettet på `\url{http://dirac.ruc.dk/imfufalateX/}`

`\end{imfufaforside}`

A.3 En primærfil til Standard $\LaTeX$

Primærfilen i afsnit A.1 benytter IMFUFA- $\LaTeX$. Hvis man ville benytte Standard $\LaTeX$, kunne man indlede primærfilen sådan:

```
\documentclass{report}
\usepackage[T1]{fontenc}
\usepackage[ansinew]{inputenc}
\usepackage[ngerman,english,danish]{babel}
\usepackage{amsmath}
\usepackage{amsthm}

\usepackage[round]{natbib} % natbib, med runde parenteser
\usepackage{graphicx}      % definerer bl.a. \includegraphics

.
.
.
```

Det færdige resultat vil dog se væsentlig anderledes ud end med IMFUFA- $\LaTeX$!

Referencer

- AMS-thm. *Using the amsthm Package*. American Mathematical Society, 2004. URL CTAN/macros/latex/required/amslatex/classes/amsthdoc.pdf.
- Karen Blixen. *Sandhedens Hævn*. Gyldendals Tranebøger, 1960.
- Johannes Braams. *Babel, a multilingual package for use with L^AT_EX's standard document classes*, 2005. URL CTAN/macros/latex/required/babel/user.pdf.
- Thomas Bredsdorff. *Den brogede oplysning*. Gyldendal, 2003.
- Erik Ellegaard Frederiksen. *Bogens Funktion og Æstetik*. Christian Ejlers' Forlag, København, 1981.
- Michel Goossens, Sebastian Rahtz og Frank Mittelbach. *The L^AT_EX Graphics Companion: Illustrating documents with T_EX and PostScript*. Addison-Wesley, Reading, Massachusetts, 1997.
- Hubert Gäblein og Rolf Niepraschk. *The pict2e package*, 2004. URL CTAN/macros/latex/contrib/pict2e/pict2e.pdf.
- Carsten Heinz. *The Listings package*, 2004. URL CTAN/macros/latex/contrib/listings/listings-1.3.pdf.
- Immanuel Kant. Beantwortung der Frage: Was ist Aufklärung? *Berlinische Monatschrift*, 1784.
- Leslie Lamport. *L^AT_EX – A Document Preparation System*. Addison-Wesley Publishing Company, Reading, Massachusetts, 2. udgave, 1994.
- Jørgen Larsen. *Guide to IMFUFA L^AT_EX*, 2005. URL <http://dirac.ruc.dk/imfufalateX/imfguide.pdf>.
- Philipp Lehman. *The csquotes package. Context sensitive markup for inline quotations*, 2005. URL CTAN/macros/latex/contrib/csquotes/csquotes.pdf.
- Frank Mittelbach, Michel Goossens m.fl. *The L^AT_EX Companion*. Addison-Wesley, 2. udgave, 2004.
- Robert Musil. *Manden uden egenskaber*, bind 1. Gyldendal, 1994. På dansk ved Karsten Sand Iversen.
- Sebastian Rahtz og Heiko Oberdiek. *Hypertext marks in L^AT_EX: a manual for hyperref*, 2004. URL CTAN/macros/latex/contrib/hyperref/doc/manual.pdf.
- R. Schlicht. *The microtype package. An interface to the micro-typographic extensions of pdfT_EX*, 2005. URL CTAN/macros/latex/contrib/microtype/microtype.pdf.
- Peter Wilson. *The Memoir Class for Configurable Typesetting. User Guide*, 2004. URL CTAN/macros/latex/contrib/memoir/memman.pdf.

Stikord

<code>\!</code>	62	<code>9pt</code>	klasse-opt. 17
<code>#</code>	12	<code>a4paper</code>	klasse-opt. 17
<code>#</code>	12	<code>a5paper</code>	klasse-opt. 17
<code>\$</code>	12	<code>\AA</code>	13
<code>\$</code>	12, 59	<code>\aa</code>	13
<code>%</code>	12	<code>abbrvnat</code>	bib-stil 95
<code>%</code>	12, 34	<code>abstract</code>	env. 19
<code>%o</code>	13	<code>\abstractname</code>	102
<code>&</code>	12	<code>accenter</code>	13
<code>&</code>	12, 21, 70, 72	<code>i</code>	matematik 65
<code>\(\)</code>	59	<code>\acute</code>	65
<code>\,</code>	62	<code>\AE</code>	13
<code>\-</code>	55	<code>\ae</code>	13
<code>\:</code>	62	<code>afsnit</code>	10, 15
<code>^</code>	12, 61	<code>agsm</code>	bib-stil 95
<code>^</code>	12	<code>\aleph</code>	84
<code>~</code>	12, 57	<code>align</code>	env. 72, 73
<code>~</code>	12	<code>align*</code>	env. 72
<code>\</code>	9	<code>\Alph</code>	103
<code>\</code>	12	<code>\alph</code>	103
<code>\\</code>	19, 21, 36, 70	<code>\alpha</code>	84
<code>{</code>	12	<code>\alsoname</code>	102
<code>}</code>	12	<code>amsfonts</code>	pakke 77
<code>"</code>	12	<code>amsmath</code>	pakke 18, 59, 64, 67, 69, 70, 75,
<code>"-</code>	56		102
<code>"/</code>	56	<code>amsmath</code>	klasse-opt. 18, 59
<code>_</code>	12, 19, 62	<code>amssymb</code>	pakke 77, 78, 84
<code>_</code>	104, 105	<code>amstext</code>	pakke 67
<code>\[\]</code>	60	<code>amsthm</code>	pakke 18, 59
<code>{ }</code>	12	<code>anførselstegn</code>	11, 12
<code>\ </code>	79	<code>\angle</code>	82
<code>—</code>	12	<code>antikva</code>	49, 50, 51
<code>_</code>	12, 61	<code>appendiks</code>	16
<code>--</code>	56	<code>\appendix</code>	16
<code>---</code>	56	<code>\appendixname</code>	102
<code>10pt</code>	klasse-opt. 17	<code>\appendixpagename</code>	102
<code>11pt</code>	klasse-opt. 17	<code>\approx</code>	79
<code>12pt</code>	klasse-opt. 17	<code>\approxeq</code>	79
<code>14pt</code>	klasse-opt. 17	<code>\arabic</code>	103
<code>17pt</code>	klasse-opt. 17		

<code>\arccos</code>	78	<code>\bigstar</code>	82
<code>\arcsin</code>	78	<code>\bigtriangledown</code>	82
<code>\arctan</code>	78	<code>\bigtriangleup</code>	82
<code>\arg</code>	78	<code>\biguplus</code>	78
argument (til L ^A T _E X-kommando)	9	<code>\bigvee</code>	78
array env.	70, 71	<code>\bigwedge</code>	78
article klasse	17	bindestreg	56
<code>\ast</code>	82	<code>\binom</code>	64
<code>\asymp</code>	79	<code>\blacklozenge</code>	82
<code>\author</code>	19	<code>\blacksquare</code>	82
		<code>\blacktriangle</code>	82
<code>\b</code>	13	<code>\blacktriangledown</code>	82
b5paper klasse-opt.	17	<code>\blacktriangleleft</code>	82
babel pakke	12, 18, 32, 33, 101	<code>\blacktriangleright</code>	82
<code>\backepsilon</code>	85	blank linje	10, 36
<code>\backprime</code>	85	<code>\blockquote</code>	33
<code>\backsim</code>	79	blokskrift	50, 51
<code>\backsimeq</code>	79	<code>\bm</code>	77
backslash	9	bm pakke	8, 77, 99, 100
<code>\backslash</code>	79	Bmatrix env.	71
bagkant	54	bmatrix env.	71
<code>\bar</code>	65	<code>\bmdefine</code>	77
<code>\barwedge</code>	81	bokse	37
<code>\baselineskip</code>	36	matematik	60
<code>\Bbbk</code>	85	bold math	77
<code>\because</code>	81	boldface	<i>Se fed skrift</i>
<code>\beta</code>	84	book klasse	17
<code>\beth</code>	84	booktabs pakke	22
<code>\between</code>	82	<code>\bot</code>	85
Bevis	74	<code>\bottomrule</code>	22
Bézier-kurve	47	<code>\bowtie</code>	85
<code>\bfseries</code>	51	<code>\boxdot</code>	79
<code>\bibfont</code>	96	<code>\boxed</code>	37, 60, 73
<code>\bibliography</code>	92, 95	<code>\boxminus</code>	79
<code>\bibliographystyle</code>	95	<code>\boxname</code>	102
<code>\bibname</code>	102	<code>\boxplus</code>	79
BIB _T _E _X	91, 92, 93, 95, 96	<code>\boxtimes</code>	79
bibtex program	96	bp (typografisk enhed)	103
<code>\Big</code>	79	<code>\bpar</code>	10, 36, 49
<code>\big</code>	79	breve	19, 65
<code>\bigcap</code>	78	brødskrift	49
<code>\bigcirc</code>	82	brøker	64
<code>\bigcup</code>	78	<code>\bullet</code>	82
<code>\biggl</code>	68	<code>\Bumpeq</code>	79
<code>\biggr</code>	68	<code>\bumpeq</code>	79
<code>\bigodot</code>	78		
<code>\bigoplus</code>	78	<code>\C</code>	66
<code>\bigotimes</code>	78	<code>\c</code>	13
<code>\bigsqcup</code>	78	<code>\Cap</code>	81

`\cap` 81
`\caption` 26, 27, 88
 cases env. 71
`\cbezier` 47
 ccfonts klasse-opt. 18
`\ccname` 102
`\cdot` 62, 63, 79
`\cdots` 70
`\celsius` 13, 85
 center env. 26
`\centerdot` 79
`\centering` 25
`\chapter` 15, 35
`\chaptername` 102
`\chapterrefname` 102
 character protrusion 54
`\check` 65
`\chi` 84
`\circ` 82
`\circeq` 79
`\circle` 44
`\circle*` 44
`\circlearrowleft` 83
`\circlearrowright` 83
`\circledast` 82
`\circledcirc` 82
`\circleddash` 82
`\cite` 92, 94
`\cite*` 94
`\citealp` 94
`\citealt` 94
`\citeauthor` 94
`\citeauthor*` 94
`\citep` 94
`\citep*` 94
`\citet` 94
`\citet*` 94
`\citeyear` 94
`\citeyearpar` 94
`\cleardoublepage` 35
 cleared side-stil 36
`\clearpage` 27, 35
`\cleartorecto` 16, 36
`\cleartoverso` 36
`\closing` 20
`\clubsuit` 85
`\cmidrule` 22
`\commentsoff` 35
`\complement` 81
 Computer Modern 49
`\cong` 79
`\contentsname` 102
`\coprod` 78
`\cos` 63, 78
`\cosh` 78
`\cot` 78
`\coth` 78
`\Cref` 87, 102
`\csc` 78
 csquotes pakke 11, 33
 CTAN 105
`\Cup` 81
`\cup` 81
`\curlyeqprec` 80
`\curlyeqsucc` 80
`\curlyvee` 81
`\curlywedge` 81
`\curvearrowleft` 83
`\curvearrowright` 83

 D 23
`\d` 13
`\dagger` 85
`\daleth` 84
 danish klasse-opt. 18, 32
`\dashbox` 45
`\dashv` 81
`\date` 19
 dcu bib-stil 95
 dd (typografisk enhed) 103
`\ddagger` 85
`\ddot` 65, 66
`\ddots` 70
 decimaltal 63
`\DeclareMathOperator` 101, 102
`\DeclareMathOperator*` 102
`\DeclareQuoteAlias` 11
 definition sætnings-stil 75
`\deg` 78
`\DeleteShortVerb` 105
`\Delta` 84
`\delta` 84
`\depth` 38, 40
 description env. 14, 36
`\descriptionlabelfont` 14
`\det` 78
`\dfrac` 64
`\DH` 13

- \dh 13
- \diagdown 85
- \diagup 85
 - diakritiske tegn 13
- \diamond 82
- \diamondsuit 85
 - Didôt-punkt 103
 - differentiation 65
- \digamma 84
- \dim 78
- \ding 52
 - dingautolist env. 53
- dingbats 53
- \dingfill 53
- \dingline 53
- \displaystyle 60
- \div 79
- \divideontimes 79
 - dk-abbrev bib-stil 95
 - dk-alpha bib-stil 95
 - dk-plain bib-stil 95
 - DK-TUG 105
 - dk-unsrt bib-stil 95
 - document env. 9
- \documentclass 7, 9, 17
 - dokumentklasse 9, 17
 - dokumentklasseerklæring 7
 - dokumentklasseoptions 17
- \dot 65, 66
- \doteq 79
- \doteqdot 79
- \dotplus 79
- \dots 69
- \dotsb 69
- \dotsc 69
- \dotsi 69
- \dotsm 69
- \dotso 69
- \doublebarwedge 81
- \Downarrow 79, 83
- \downarrow 79, 83
- \downdownarrows 83
- \downharpoonleft 83
- \downharpoonright 83
 - draft 42
 - dvips program 41
- ebook klasse-opt. 17, 96
- \ell 85
 - em (typografisk enhed) 103
- \emph 52
 - empty side-stil 35
- \emptypages 36
- \emptyset 66, 81
- \enclname 102
- \enlargethispage 36
- \enquote 11, 33
- \enquote* 11, 34
- \enspace 62
 - enumerate env. 14, 36, 53
- \eps 84
- \epsilon 84
 - epstopdf program 41
- \eqcirc 79
- \eqsim 79
- \eqslantgtr 80
- \eqslantless 80
 - equation env. 73
- \equiv 79
- \eref 73, 100
- \eta 84
- \eth 85
- \euro 13, 85
 - European Computer Modern 49
 - ex (typografisk enhed) 103
- \exists 66, 81
- \Exp 66
- \exp 63, 78
- \fallingdotseq 79
 - fast bagkant 54
 - fast mellemrum 12, 57
- \fbox 37
 - fed skrift 50, 51
 - matematik 77
- figure env. 26
- \figurename 102
- \figurerefname 102
 - filnavne 8
- final 42
- \Finv 85
- \flat 85
- fleqn klasse-opt. 18
- floats 26
- fmpage env. 41
- \fnsymbol 104
 - fodnoter 11
 - nummerering 104

fodnotesymboler 104
 fodtegn 61
 font *Se* skriftsnit
 font expansion 54
 fontjustering 54
 \footnote 11
 \footnotesize 49
 \forall 66, 81
 \foreignblockquote 34
 \foreignlanguage 33
 \foreignquote 33
 \foreignquote* 34
 formler
 fremhævede 60
 i teksten 59
 nummererede 72
 forside 18
 \frac 64
 \framebox 37
 vedr. picture 45
 \fref 87, 102
 fremhævet formel 60
 fremhævet skrift 51
 \frown 85

 gaffelfunktion 71
 \Game 85
 \Gamma 84
 \gamma 84
 \gcd 78
 \geq 64, 80
 \geqq 80
 \geqslant 80
 geviert 62, 103
 \gg 80
 \ggg 80
 \gimel 84
 \glossaryname 102
 \gnapprox 80
 \gneq 80
 \gneqq 80
 \gnsim 80
 graphicx pakke 39, 41, 42
 \grave 65
 grotesk *Se* blokskrift
 græske bogstaver (matematik) 66, 84
 \gtrapprox 80
 \gtrdot 80
 \gtreqless 80
 \gtreqqlless 80
 \gtrless 80
 \gtrsim 80
 \gvertneqq 80

 \H 13
 \hat 65
 \hbar 66, 85
 headings side-stil 35
 \headtoname 102
 \heartsuit 85
 hebraiske bogstaver (matematik) 84
 \height 38, 40
 henvisninger 87
 \hfill 25, 28
 hiderotate 42
 hidescale 42
 \hline 21
 \hom 78
 \hookleftarrow 83
 \hookrightarrow 83
 hovedafsnit 15
 \hslash 85
 \HUGE 49
 \Huge 49
 \huge 49
 hyperlink 96
 hyperref pakke 96, 97, 98
 \hypersetup 96
 \hyphenation 56
 \hyphenblockquote 34
 \hyphenquote 33
 \hyphenquote* 34
 hældende skrift 49, 50, 51
 hængende tal 52

 \i 13
 ibox env. 30, 37
 ibreak sætnings-stil 75, 76
 icomma pakke 63
 idefinition sætnings-stil 75
 \idotsint 78
 \iff 66, 83
 \iiint 78
 \iint 78
 \iint 78
 \Im 78
 imfart klasse 17, 19
 imfbook klasse 9, 17, 19, 35
 imfletter klasse 9, 17, 19, 20, 35

`imfnote klasse` 9, 17, 19, 35
`imfreport klasse` 9, 17, 19, 35
`\imfufapar` 10, 55
`\in` 64, 81
`\includegraphics` 27, 41, 42
`\indent` 36
`\index` 89, 90
`\indexname` 102
`indholdsfortegnelse` 16
`indrammede formler` 60
`indrammet tekst` 37
`indrykning` 36
`\inf` 78
`\injl` 78
`\input` 8
`\int` 61, 78
`integraltegn` 61, 78
`\intercal` 85
`\iota` 84
`iproof sætnings-stil` 75
`italic` *Se kursiv*
`\item` 14
`itemize env.` 14, 36, 101
`\itshape` 51

`\j` 13

`\k` 13
`kapitel` 15
`kapitæler` 50, 51
`\kappa` 84
`kasser` 37
`\ker` 78
`kluwer bib-stil` 95
`kolumnetitel` 35
`kommando` 7, 9, 99
`kommatal` 63
`kommentartegn` 8, 12, 34
`kursiv` 49, 50, 51
`kvadratrod` 65

`\L` 13
`\l` 13
`\label` 16, 26, 27, 28, 73, 87, 88
`\labelitemi` 101
`\Lambda` 84
`\lambda` 84
`landscape klasse-opt.` 17
`\langle` 68, 79
`\LARGE` 49

`\Large` 49
`\large` 49
`\LaTeX` 11
`latex program` 41, 42
`Latin Modern` 49
`\lbrace` 79
`\lbrack` 79
`\lceil` 79
`\ldots` 69
`\Leftarrow` 83
`\leftarrow` 83
`\leftarrowtail` 83
`\leftharpoondown` 83
`\leftharpoonup` 83
`\leftleftarrows` 83
`\Leftrightarrow` 83
`\leftrightarrow` 83
`\leftrightharpoons` 83
`\leftrightsquigarrow` 83
`\leftthreetimes` 82
`\leq` 64, 80
`leqno klasse-opt.` 18, 73
`\leqq` 80
`\leqslant` 80
`\lessapprox` 80
`\lessdot` 80
`\lesseqgtr` 80
`\lesseqqgtr` 80
`\lessgtr` 80
`\lesssim` 80
`letter klasse` 17
`letter env.` 19
`\letterstyle` 20
`levende kolumnetitel` 35
`\lfloor` 79
`\lfrac` 100
`\lg` 78
`lheadings side-stil` 35
`\lim` 63, 78
`\liminf` 78
`\limits` 78
`\limsup` 78
`\line` 45
`linjeafstand` 49, 54
`linjelængde` 54
`linjeskift` 19, 36
`\listboxname` 102
`\listfigurename` 102

listings pakke 105
 $\listoffigures$ 27
 $\listoftables$ 27
 $\listtablename$ 102
 litteraturdatabase 91
 litteraturliste 91
 $\ll$ 80
 $\Lleftarrow$ 83
 $\lll$ 80
 $\ln$ 78
 $\lnapprox$ 80
 $\lneq$ 80
 $\lneqq$ 80
 $\lnsim$ 80
 $\location$ 19
 $\log$ 78
 $\Longleftarrow$ 83
 $\longleftarrow$ 83
 $\Longleftrightarrow$ 83
 $\longleftrightarrow$ 83
 $\longmapsto$ 83
 $\longrightarrow$ 83
 $\rightarrow$ 83
 $\looparrowleft$ 83
 $\looparrowright$ 83
 $\lozenge$ 82
 $\lplain$ side-stil 35
 $\Lsh$ 83
 $\ltimes$ 79
 $\lVert$ 79
 $\lvert$ 79
 $\lvertneqq$ 80
 længder 102
 læselighed 54
 løs bagkant 54
 $\makebox$ 37, 103
 vedr. picture 45
 $\makeindex$ 89
 makeindex program 89
 $\MakeShortVerb$ 104
 $\maketitle$ 8, 18, 19
 $\mapsto$ 66, 83
 $\marginpar$ 31
 $\mathbb$ 67, 76, 77
 $\mathbf$ 77
 $\mathcal$ 76
 $\mathfrak$ 76, 77
 $\mathit$ 77
 $\mathring$ 65
 $\mathrm$ 68, 77
 mathrsfs pakke 77
 $\mathscr$ 76, 77
 $\mathsf$ 77
 $\mathtt$ 77
 $\mathversion$ 77
 matricer 70
 matrix env. 70, 71
 $\max$ 78
 MaxMatrixCols 71
 $\maxsecnumdepth$ 15
 $\maxtocdepth$ 16, 17
 $\mbox$ 25, 36, 37
 $\mdseries$ 51
 $\measuredangle$ 82
 medium skrift 50, 51
 mediævaltal 52
 mellemrubrik 15
 mellemrum 67
 fast 12, 57
 forskellige størrelser 62
 forsvundne 11
 i formler 59, 62
 kommando 12
 mellemslag 10
 memhfixc pakke 96
 memoir klasse 15, 16, 17, 22, 30, 31, 35,
 36, 87, 96, 105
 METAPOST 44
 $\mho$ 85
 microtype pakke 54
 $\mid$ 82
 middelværdi 66, 68
 $\midrule$ 22
 mikrotypografi 54
 $\min$ 78
 minipage env. 28, 37, 40, 103
 $\miniscule$ 49
 minustegn 56
 mkind program 89, 90
 $\models$ 81
 $\mp$ 79
 μ 84
 $\multicolumn$ 23, 24
 $\multimap$ 85
 $\N$ 66
 ∇ 65, 66

<code>\natbib</code> pakke	92, 94, 95	<code>\nrightarrow</code>	83
<code>\natural</code>	85	<code>\nshortmid</code>	82
<code>\ncong</code>	79	<code>\nshortparallel</code>	82
<code>\nearrow</code>	83	<code>\nsim</code>	79
<code>\neg</code>	66, 81	<code>\nsubseteq</code>	81
negative tal	63	<code>\nsubseteqq</code>	81
<code>\neq</code>	64, 79	<code>\nsucc</code>	80
<code>\newcommand</code>	99	<code>\nsucceq</code>	80
<code>\newcommand*</code>	99, 101	<code>\nsupseteq</code>	81
<code>\newcomment</code>	34, 35	<code>\nsupseteqq</code>	81
<code>\newline</code>	36	<code>\ntriangleleft</code>	82
<code>\newpage</code>	35	<code>\ntrianglelefteq</code>	82
<code>\newsfloat</code>	30	<code>\ntriangleright</code>	82
<code>\newtheorem</code>	75	<code>\ntrianglerighteq</code>	82
<code>\newtheorem*</code>	75	<code>\ntup</code>	70, 101
<code>\newtheoremstyle</code>	76	<code>\nu</code>	84
<code>\nexists</code>	81	nummererede formler	72
<code>\NG</code>	13	<code>\nVDash</code>	81
<code>\ng</code>	13	<code>\nVdash</code>	81
<code>\ngeq</code>	80	<code>\nvDash</code>	81
<code>\ngeqq</code>	80	<code>\nvdash</code>	81
<code>\ngeqslant</code>	80	<code>\nwarrow</code>	83
<code>\ngtr</code>	80	ny linje	36
nheadings side-stil	35	nyt afsnit	10, 36, 55
<code>\ni</code>	64, 81	<code>\O</code>	13
<code>\nicefrac</code>	11	<code>\o</code>	13
<code>\nLeftarrow</code>	83	<code>\odot</code>	79
<code>\nleftarrow</code>	83	<code>\OE</code>	13
<code>\nLeftrightarrow</code>	83	<code>\oe</code>	13
<code>\nleftrightarrow</code>	83	<code>\ofname</code>	102
<code>\nleq</code>	80	<code>\oint</code>	78
<code>\nleqq</code>	80	<code>\oldstylenums</code>	52
<code>\nleqslant</code>	80	<code>\Omega</code>	84
<code>\nless</code>	80	<code>\omega</code>	84
<code>\nmid</code>	82	<code>\ominus</code>	79
<code>\nocite</code>	94	oneside klasse-opt.	17
nohyperref pakke	98	openany klasse-opt.	17
<code>\noindent</code>	36	<code>\opening</code>	20
<code>\nonumber</code>	73	openleft klasse-opt.	17
<code>\noprelistbreak</code>	36	openright klasse-opt.	17
<code>\normalfont</code>	51	operatorer, aritmetiske	62
<code>\normalsize</code>	49	<code>\oplus</code>	79
<code>\not</code>	64	opremsning	14
<code>\notag</code>	73	opret skrift	50, 51
<code>\nparallel</code>	82	option	8
nplain side-stil	35	optisk lige bagkant	54
<code>\nprec</code>	80	orddeling	55
<code>\npreceq</code>	80	<code>\oslash</code>	79
<code>\nrightarrow</code>	83		

`otherlanguage` env. 32
`otherlanguage*` env. 33
`\otimes` 79
`\oval` 46
`\overbrace` 65
`\overleftarrow` 65
`\overline` 65
`\overrightarrow` 65
`overskrift` 15
`overspring tekst` 34

`\pagename` 102
`\pageref` 87
`\pagerefname` 102
`\pagestyle` 35
`pakke` 8
`palatino klasse-opt.` 17, 18, 52
`\par` 10, 25, 36, 49
`\paragraph` 15
`\parallel` 82
`parenteser` 68, 79
`\part` 15
`\partial` 65, 66
`\partname` 102
`\partrefname` 102
`pdflatex program` 6, 41, 42, 54
`\perp` 82
`\phantom` 37
`\Phi` 84
`\phi` 84
`\Pi` 84
`\pi` 84
`pict2e pakke` 46
`picture` env. 44
`pifont pakke` 52, 53
`pile` 66, 83
`\pitchfork` 85
`plain bib-stil` 95
`plain side-stil` 35
`plain s etnings-stil` 75, 76
`plainnat bib-stil` 95
`plnnatdk bib-stil` 94, 95
`\pm` 79
`pmatrix` env. 71
`pplleading klasse-opt.` 17, 54
`\Pr` 66, 78
`\prec` 80
`\precapprox` 80
`\preccurlyeq` 80

`\preceq` 80
`\precnapprox` 80
`\precneqq` 80
`\precnsim` 80
`\precsim` 80
`\Pref` 87, 102
`\pref` 87, 102
`\prefacename` 102
`prikker` 69
`prim erfil` 8
`\prod` 61, 78
`produkttegn` 61
`\projlim` 78
`\promille` 13, 85
`proof` env. 74, 75
`\proofname` 102
`\propto` 79
`pr ambel` 8
`\Psi` 84
`\psi` 84
`pt (typografisk enhed)` 102
`punkt (typografisk enhed)` 102
`\put` 44

`\Q` 66
`\qbezier` 47
`\qqquad` 62
`\quad` 62
`quotation` env. 31
`quote` env. 31

`\R` 66
`\r` 13
`\RaggedRight` 54
`\raisebox` 37, 38
`\rangle` 68, 79
`\rbrace` 79
`\rbrack` 79
`\rceil` 79
`\Re` 78
`\ref` 16, 73, 87
`ref-v erdi` 103
`referencer` 91
`\reflectbox` 37, 39
`\refname` 102
`relationer` 64
`remark s etnings-stil` 75
`\renewcommand` 99
`\renewcommand*` 99, 101
`report klasse` 17

`\resizebox` 37, 39
`\resizebox*` 39
`\reversesideparfalse` 31
`\rfloor` 79
`\rho` 84
`\Rightarrow` 83
`\rightarrow` 66, 83
`\rightarrowtail` 83
`\rightharpoondown` 83
`\rightharpoonup` 83
`\rightleftarrows` 83
`\rightleftharpoons` 83
`\rightrightarrows` 83
`\rightsquigarrow` 83
`\rightthreetimes` 82
`\risingdotseq` 79
`\rmfamily` 51
 rodtegn 65
 Roman 49
`\Roman` 103
`\roman` 103
`\rotatebox` 40
`\Rrightarrow` 83
`\Rsh` 83
`\rtimes` 79
`\rule` 39
`\rVert` 79
`\rvert` 79

 sans serif *Se* blokskrift
`\scalebox` 37, 39
`\scriptsize` 49
`\scshape` 51
`\searrow` 83
`\sec` 78
`\section` 15, 35
`\section*` 16
`\sectionrefname` 102
`\seename` 102
`\selectlanguage` 32, 56
`\setcounter` 15, 17, 104
`\setminus` 81
`\setsecnumdepth` 15
`\settocdepth` 16, 17
`\setverbatimfont` 105
`\sffamily` 51
`\sgn` 71, 102
`\sharp` 85
`\shortmid` 82

`\shortparallel` 82
`\showhyphens` 56
 showkeys pakke 89
`\sidebar` 30, 31
`\sidebarfont` 30
`\sidebarhsep` 30
`\sidebarwidth` 30
`\sidepar` 31
 sideskift 35
`\Sigma` 84
`\sigma` 84
`\sim` 79
`\simeq` 79
`\sin` 63, 78
`\sinh` 78
 skemaer 21
 skriftgrad 49, 54
 skriftsnit 49
 skriftstørrelse *Se* skriftgrad
 skriftvalg i matematik 76
 skrivemaskineskrift 50, 51
 skydning 17, 54
 slanted *Se* hældende skrift
`\slshape` 51
`\small` 49
 small caps *Se* kapitæler
`\smallfrown` 85
`\smallsetminus` 81
`\smallsmile` 85
`\smile` 85
`\spadesuit` 85
`\sphericalangle` 82
 split env. 73
`\sqcap` 81
`\sqcup` 81
`\sqrt` 65
`\sqsubset` 81
`\sqsubseteq` 81
`\sqsupset` 81
`\sqsupseteq` 81
`\square` 82
`\Sref` 87, 102
`\SS` 13
`\ss` 13
`\star` 82
 stikord 89
 stikordsregister 89
`\strictpagechecktrue` 31
`\subbottom` 29

<code>\subcaption</code>	28	<code>\texorpdfstring</code>	98
<code>\subcaptionref</code>	29	<code>\text</code>	67
<code>\subparagraph</code>	15	<code>\textasciicircum</code>	12
subscript	<i>Se</i> fodtegn	<code>\textasciitilde</code>	12
<code>\subsection</code>	15	<code>\textbackslash</code>	12
<code>\Subset</code>	81	<code>\textbf</code>	51
<code>\subset</code>	64, 81	<code>\textbraceleft</code>	12
<code>\subseteq</code>	64, 81	<code>\textbraceright</code>	12
<code>\subseteqq</code>	81	<code>\textit</code>	10, 51, 52
<code>\subsetneq</code>	81	<code>\textmd</code>	51
<code>\subsetneqq</code>	81	<code>\textnormal</code>	51, 67
<code>\subsubsection</code>	15	<code>\textrm</code>	51
<code>\subtop</code>	29	<code>\textsc</code>	51
<code>\succ</code>	80	<code>\textsf</code>	51
<code>\succapprox</code>	80	<code>\textsl</code>	51
<code>\succcurlyeq</code>	80	<code>\textstyle</code>	60
<code>\succeq</code>	80	<code>\texttt</code>	51
<code>\succnapprox</code>	80	<code>\textup</code>	51
<code>\succneqq</code>	80	<code>\textwidth</code>	103
<code>\succnsim</code>	80	<code>\textxx</code>	51
<code>\succsim</code>	80	<code>\tfrac</code>	64
<code>\sum</code>	61, 78	<code>\TH</code>	13
sumtegn	61	<code>\th</code>	13
<code>\sup</code>	78	<code>\the...</code>	103
superscript	<i>Se</i> toptegn	<code>\thechapter</code>	103
<code>\Supset</code>	81	<code>\thefootnote</code>	103, 104
<code>\supset</code>	64, 81	theorem env.	74, 75
<code>\supseteq</code>	64, 81	<code>\theoremname</code>	102
<code>\supseteqq</code>	81	<code>\theoremrefname</code>	102
<code>\supsetneq</code>	81	<code>\theoremstyle</code>	75
<code>\supsetneqq</code>	81	<code>\thepage</code>	103
<code>\swarrow</code>	83	<code>\therefore</code>	81
symboler med variabel størrelse	78	<code>\Theta</code>	84
Sætning	74	<code>\theta</code>	84
		<code>\thickapprox</code>	79
tabel	21	<code>\thicklines</code>	46
<code>table</code> env.	26	<code>\thicksim</code>	79
<code>\tablename</code>	102	<code>\thinlines</code>	46
<code>\tableofcontents</code>	16	<code>\thispagestyle</code>	35
<code>\tableofcontents*</code>	16	<code>\tilde</code>	65
<code>\tablerefname</code>	102	<code>\times</code>	63, 79
<code>tabular</code> env.	21, 70	times klasse-opt.	18
tabulatortegn	21, 72	<code>\tiny</code>	49
talmængderne	66	<code>\title</code>	19
<code>\tan</code>	78	<code>\titleref</code>	88
<code>\tanh</code>	78	<code>\titlingpage</code> env.	19
tankestreg	56	<code>\today</code>	19
<code>\tau</code>	84	tom side	36
tekst i formler	67	<code>\ton</code>	64

<code>\top</code> 85	<code>\varliminf</code> 78
<code>\toprule</code> 22	<code>\varlimsup</code> 78
<code>toptegn</code> 61	<code>\varnothing</code> 66, 81
<code>\totalheight</code> 38, 40	<code>\varphi</code> 84
<code>tplain side-stil</code> 19, 35	<code>\varpi</code> 84
<code>\tref</code> 87, 102	<code>\varprojlim</code> 78
<code>\triangle</code> 82	<code>\varpropto</code> 79
<code>\triangledown</code> 82	<code>\varrho</code> 84
<code>\triangleleft</code> 82	<code>\varsubsetneq</code> 81
<code>\trianglelefteq</code> 82	<code>\varsubsetneqq</code> 81
<code>\triangleq</code> 79	<code>\varsupsetneq</code> 81
<code>\triangleright</code> 82	<code>\varsupsetneqq</code> 81
<code>\trianglerighteq</code> 82	<code>\vartheta</code> 84
<code>\ttfamily</code> 51	<code>\vartriangle</code> 82
<code>tuborgfunktion</code> 71	<code>\vartriangleleft</code> 82
<code>TUG</code> 105	<code>\vartriangleright</code> 82
<code>\twoheadleftarrow</code> 83	<code>\Vdash</code> 81
<code>\twoheadrightarrow</code> 83	<code>\vDash</code> 81
<code>twoside klasse-opt.</code> 17	<code>\vdash</code> 81
<code>typewriter type</code> <i>Se</i> skrivemaskineskrift	<code>\vdots</code> 70
<code>typografiske enheder</code> 102	<code>\vec</code> 65
<code>tællere ('counters')</code> 103	<code>\vector</code> 45
<code>\u</code> 13	<code>\vee</code> 81
<code>underafsnit</code> 15	<code>\veebar</code> 81
<code>\underbrace</code> 65	<code>\verb</code> 104
<code>\underline</code> 65	<code>\verb*</code> 104
<code>\unitlength</code> 44	<code>verbatim env.</code> 104, 105
<code>unsrnat bib-stil</code> 95	<code>verbatim* env.</code> 104, 105
<code>\Uparrow</code> 79, 83	<code>\verbatiminput</code> 105
<code>\uparrow</code> 79, 83	<code>Vmatrix env.</code> 71
<code>\Uparrow</code> 79, 83	<code>vmatrix env.</code> 71
<code>\updownarrow</code> 79, 83	<code>\Vvdash</code> 81
<code>\updownarrow</code> 79, 83	<code>\wedge</code> 81
<code>\upharpoonleft</code> 83	<code>\widehat</code> 65
<code>\upharpoonright</code> 83	<code>\widetilde</code> 65
<code>\uplus</code> 81	<code>\width</code> 38, 40
<code>\upshape</code> 51	<code>\wp</code> 85
<code>\Upsilon</code> 84	<code>\wr</code> 85
<code>\upsilon</code> 84	 <code>x-højde</code> 103
<code>\upuparrows</code> 83	<code>\Xi</code> 84
<code>\url</code> 96, 97	<code>\xi</code> 84
<code>url pakke</code> 97	<code>\xleftarrow</code> 66
<code>\usepackage</code> 8, 95	<code>\xrightarrow</code> 66
<code>usynlig tekst</code> 37	 <code>\Z</code> 66
<code>\v</code> 13	<code>Zapf Dingbats</code> 53
<code>valgmulighed</code> 8, 17	<code>\zeta</code> 84
<code>\Var</code> 66	 <code>ångstrøm</code> 68
<code>\varinjlim</code> 78	
<code>\varkappa</code> 84	