

Vecka 8. Nästlade och generiska strukturer, typsättning

Programmering, grundkurs (pgk:dod)

Mattias Nordahl

Datavetenskap, LTH, Lunds universitet
<https://lunduniversity.github.io/pgk>

EDAB05, Lp1-2, HT2026

Kompilerad den 25 augusti 2026

8 Datorer och Datoranvändning: Typsättning

- Ordbehandling

Föreläsning: Latex

Förberedelse inför veckans dod-laboration!

- Ordbehandling
- $\text{\LaTeX}$
- Dokumentstruktur: dokumentklasser, omgivningar, text, stycken, listor, tabeller, ...
- Programlistor
- Matematiska formler
- Bilder

Ordbehandling

Ordbehandling: WYSIWYG vs. LaTeX

Två olika sätt att skriva text:

- WYSIWYG (What You See Is What You Get)
 - Exempel: Microsoft Word
 - Direkt återspeglning av slutresultatet på skärmen
 - Inkluderar teckensnitt, storlekar, avstånd, etc.
- $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ fungerar annorlunda!
 - Fokus på innehåll istället för utseende
 - Skriver kommandon eller "taggar" för att uttrycka formatering
 - Texten *kompileras* till ett formaterat dokument (oftast PDF)

$\text{\LaTeX}$ - Arbetssätt och Exempel

I $\text{\LaTeX}$ skriver man text och formateringskommandon i en textfil.

För rätvinkliga trianglar gäller `\emph{Pythagoras sats}`:
`$a^2 + b^2 = c^2$`

*Tecknet \$ markerar början och slutet på matematiska uttryck.
 $\text{\LaTeX}$ hanterar automatiskt stil och placering för matematiska element:*

För rätvinkliga trianglar gäller *Pythagoras sats*: $a^2 + b^2 = c^2$

Ett större exempel

If f is continuous on the closed interval $a \leq x \leq b$ and differentiable on the open interval $a < x < b$, then there exists a point ξ , $a < \xi < b$ such that:

```
\begin{displaymath}
  f(b) - f(a) = f'(\xi)(b - a).
\end{displaymath}
```

Ett större exempel

If f is continuous on the closed interval $a \leq x \leq b$ and differentiable on the open interval $a < x < b$, then there exists a point ξ , $a < \xi < b$ such that:

```
\begin{displaymath}
  f(b) - f(a) = f'(\xi)(b - a).
\end{displaymath}
```

If f is continuous on the closed interval $a \leq x \leq b$ and differentiable on the open interval $a < x < b$, then there exists a point ξ , $a < \xi < b$ such that:

$$f(b) - f(a) = f'(\xi)(b - a).$$

Interaktiv Demonstration

Demo!

Jämför WYSIWYG med $\text{\LaTeX}$

$\text{\LaTeX}$ -historik

- $\text{\TeX}$ skapades av Donald Knuth som svar på brister i dåtidens typsättning när han skrev sin andra bok.
- Leslie Lamport byggde vidare på detta och gjorde $\text{\TeX}$ tillgängligt för en bredare publik genom $\text{\LaTeX}$. $\text{\TeX}$ kan liknas vid ett lågnivå programmeringsspråk som assembly, och $\text{\LaTeX}$ vid ett högnivå programmeringsspråk som Scala.
- $\text{\LaTeX}$ tillåter forskare och författare att fokusera på innehållets kvalitet snarare än dess presentation.
- Används för vetenskapliga dokument, böcker, och akademiska publikationer. Är t.ex. utmärkt för att skriva exjobbsrapporter, avhandlingar, CV, etc.

Arbeta med $\text{\LaTeX}$

- .tex-filer är rena textfiler som kan redigeras med vilken textredigerare som helst.
- pdf $\text{\LaTeX}$ är standardkompilatorn och fungerar bra för de flesta dokument.
- Xe $\text{\LaTeX}$ och Lua $\text{\LaTeX}$ är modernare alternativ, och erbjuder t.ex. bättre stöd för moderna typografiska funktioner och fullt stöd för Unicode (kan t.ex. hantera svenska tecken).
- Det finns bra $\text{\LaTeX}$ -specifika editorer som texmaker, TeXShop, och TeXstudio. Men VS Code hanterar det också bra med LaTeX Workshop-tillägget.
- Ett annat populärt alternativ är Overleaf, vilket fungerar som Google Doc men för $\text{\LaTeX}$. Det kör helt i webbläsaren och undviker alltså installationer på den egna datorn.

Fördelar med $\text{\LaTeX}$

- $\text{\LaTeX}$ ger dokument ett professionellt och konsekvent utseende automatiskt.
- Ren text-format gör det lätt att använda verktyg som Git för versionskontroll, vilket är idealiskt för samarbetsprojekt.
- $\text{\LaTeX}$ -communityt erbjuder omfattande stöd och resurser, vilket gör det lätt att få hjälp och lära sig mer om avancerade funktioner.
- *Lätt varning* för det stora antalet paket och funktioner som finns. Det kan ibland kännas överväldigande och leda till att man stöter på underliga problem.

Dags att hoppa in i $\text{\LaTeX}$!

Nu dyker vi ner i detaljerna!

Använda Kommandon i LaTeX

- $\text{\LaTeX}$ -kommandon styr hur text formateras och struktureras.
- Kan ändra typsnitt, storlek, färg, och mer.
- De kan även organisera text i avsnitt, listor och tabeller.
- Kommandon finns i två former: enkla kommandon och omgivningar.

- Enkla kommandon:

```
\textbf{fetstil}
```

fetstil

```
\textit{kursiv stil}
```

kursiv stil

- Omgivningar:

```
\begin{itemize}
```

```
\item Första elementet i en punktlista!
```

```
\end{itemize}
```

Exempel

% Ex: enkelt kommando

`\textbf{Fetstil}` gör texten fet, `\textit{kursiv stil}` gör texten kursiv. Ofta används `\emph{emfas}` för att markera `\emph{viktig text}`.

Fetstil gör texten fet, *kursiv stil* gör texten kursiv. Ofta används *emfas* för att markera *viktig text*.

% Ex: listor enumerate/itemize

```
\begin{enumerate}
  \item Listor hjälper till att organisera information.
  \item Varje \item skapar en ny punkt i listan.
\end{enumerate}
```

- 1 Listor hjälper till att organisera information.
- 2 Varje
- 3 skapar en ny punkt i listan.

Kommandon i LaTeX skrivs med en backslash (\) och kan ha parametrar både före och efter texten som de formaterar. T.ex.:

- `\textbf{text}` applicerar fetstil.
- Omgivningar som `\begin{itemize}` används för att skapa strukturerade format som listor.

Dokumentklasser

- *Dokumentklassen* styr dokumentets övergripande utseende.
- Vanliga dokumentklasser: article, report, book, letter, beamer.
- article används mest; beamer för presentationer.

```
% Exempel på användning av dokumentklassen article
\documentclass{article}

% "Preamble" - inställningar och paket, t.ex.:
\usepackage{graphicx}           % för att inkludera bilder
\title{Enkelt LaTeX-dokument}  % sätt dokumentets titel

\begin{document}
\maketitle                     % skapa titelsida
Hej världen! Detta är ett enkelt dokument.
\end{document}
```

- article är idealisk för vetenskapliga artiklar, rapporter och anteckningar.
- beamer används för att skapa visuellt engagerande presentationer.

Omgivningar

- Kommandon som innehåller flera rader text kallas *omgivningar*.
- En omgivning definieras av `\begin{...}` och `\end{...}`.
- Omgivningar påverkar utseendet för allt innehåll de omsluter.

% Exempel med omgivningar

```
\begin{itemize}
  \item Detta är en punktlista.
  \item Listor hjälper till att organisera information.
  \item Denna punkt har flera delar
  \begin{enumerate}
    \item Första delen
    \item Och resten
  \end{enumerate}
\end{itemize}
```

- `itemize` omgivningen är bra för att skapa punktlister.
- Varje `item` representerar en punkt i listan..

Löpande text

- Radslut och antal mellanslag mellan ord har ingen betydelse.
- En eller flera blanka rader ger ett nytt stycke.

```
Det här      är en text  
som jag  har skrivit.
```

```
Det är  
en lång text med flera  
rader.
```

```
Här börjar det ett  
nytt stycke i texten.
```

Rubriker

$\text{\LaTeX}$ numrerar rubriker automatiskt. Man anger en rubrik med `\section` eller `\subsection`.

```
\section{Inledning}  
\section{Utförande}  
\subsection{Del 1}  
\subsection{Del 2}  
\section{Slutsatser}
```

Ovan motsvarar:

```
{\Large\bfseries 1 Inledning} \\\[1ex]  
{\Large\bfseries 2 Utförande} \\\[1ex]  
{\large\bfseries 2.1 Del 1} \\\[1ex]  
{\large\bfseries 2.2 Del 2} \\\[1ex]  
{\Large\bfseries 3 Slutsatser}
```

Ändra textens utseende

- `\emph{viktigt}` – Betonar text
- `\texttt{Square}` – Skrivmaskinstypsnitt, används ofta för kod

```
Här skriver jag något \emph{viktigt}.  
Och i Scala har vi skapat klassen \texttt{Square}.  
I sällsynta fall kan man vilja skriva något \textbf{i fetstil}.
```

Här skriver jag något *viktigt*. Och i Scala har vi skapat klassen `Square`. I sällsynta fall kan man vilja skriva något **i fetstil**.

Använd sparsamt: fetstil, lutande text, storleksändringar

Specialtecken

- % – Kommentarer till slutet av raden
- Tecken som kräver speciell hantering:

`\$ \% \_ \# \& \{ \} \textbackslash`

Streck, mellanrum, och punkter:

`\enquote{pgk-kursen}` skrivs med bindestreck.

Dod-delen pågår under vecka 1--4.

Tyvärr är den inte längre`\ldots`

Ibland vill man göra en lång paus i en mening---för att skapa spänning.

`\quad` Telefon: 046--222~80~38.`\`

Dagens datum: `\today`.

Fotnoter

Fotnoter är lätta att skriva:

```
Om man använder \LaTeX  
\footnote{uttalas  
  \enquote{lah-tekh}} så  
blir det bra. Alla rapporter  
blir automatiskt  
% snyggt  
professionellt  
utformade.
```

- Fotnoter numreras automatiskt: 1, 2, ...
- Men kan även modifieras till t.ex. bokstäver eller specialtecken: *, †, ‡, §, ¶, ...

Citationstecken (1/2)

Använd \enquote i "löpande text".

% Svenska citat:

Hon sa \enquote{hej där} i löpande text.

% Citat i citat:

Kompisen repeterade: \enquote{Hon sa \enquote{hej där} i löpande text}.

% Raka dubbla tecken (endast kod/ASCII):

Rakt dubbelt citattecken i kod: \texttt{\textquotedbl}

Citationstecken (2/2)

Använd \enquote i löpande text.

% Apostrof:

Typografisk apostrof: 0\textquoteright{}Neill, rock\textquoteright
n\textquoteright{}roll.

ASCII-apostrof (visa tecknet): \textquotesingle

% Svensk genitiv har ingen apostrof:

Annas bok; EU:s beslut (inte EU\textquoteright s).

% Engelska citat manuellt:

Engelska: \enquote{hi there} (skrivs med \texttt{\enquote{}} och
\texttt{}}).

Listor

Punktlistor är enkla:

```
\begin{itemize}
  \item första punkten
  \item här kommer den andra
    punkten i listan
\end{itemize}
```

- första punkten
- här kommer den andra punkten i listan

Numrerade listor är lika enkla:

```
\begin{enumerate}
  \item första punkten
  \item här kommer den andra
    punkten i listan
\end{enumerate}
```

- 1 första punkten
- 2 här kommer den andra punkten i listan

Definitioner

Man kan skapa fina definitioner:

Några klasser som vi använder:

```
\begin{description}  
  \item[Scanner] Inläsning från tangentbordet  
  \item[Random] Slumptal  
  \item[File] För att läsa och skriva filer  
\end{description}
```

Scanner Inläsning från tangentbordet

Random Slumptal

File För att läsa och skriva filer

Tabeller

Tabeller är enkla att skapa:

- Använd tabular-omgivningen
- Specificera antal kolumner och hur texten ska justeras
- Använd & för att skilja kolumner och \\ för att skilja rader

```
\begin{tabular}{lcr}
  Produkt & Typ      & Pris      \\
  \hline
  Skruvar & stora    & 0.18~kr \\
  Muttrar & M16      & 0.38~kr \\
  Spikar  & 12~tum   & 0.12~kr
\end{tabular}
```

Produkt	Typ	Pris
Skruvar	stora	0.18 kr
Muttrar	M16	0.38 kr
Spikar	12 tum	0.12 kr

Flytande tabeller

Med en `table`-omgivning skapar man en flytande tabell, som $\LaTeX$ själv placerar där det är lämpligt.

Man kan ange sin preferens för placering med `h` (here), `t` (top), `b` (bottom), `p` (page).

```
\begin{table}[t]
  \begin{tabular}{|l|c|r|}
    Produkt & Typ      & Pris      \\
    \hline
    Skruvar & stora    & 0.18~kr \\
    Muttrar & M16      & 0.38~kr \\
    Spikar  & 12~tum   & 0.12~kr
  \end{tabular}
  \caption{Våra produkter}
\end{table}
```

Att referera till etiketter

*Med etiketter (label) kan man referera till tabeller från texten:
(Figurer hanteras likadant som tabeller, i en figure-omgivning.)*

```
\begin{table}
  \begin{tabular}{lcr}
    Produkt & Typ      & Pris      \\
    \hline
    Skruvar & stora  & 0.18~kr \\
    Muttrar & M16    & 0.38~kr \\
    Spikar  & 12~tum & 0.12~kr
  \end{tabular}
  \caption{Våra produkter}
  \label{produkter}
\end{table}
```

Senare i texten kan vi referera till vår tabell `\ref{produkter}`, som visar våra produkter.

Programlistor

- För programkod kan man använda verbatim-omgivningen.
- Den visar texten precis som den är skriven.
- Med paketet fancyvrb kan man också använda `\VerbatimInput`-kommandot för att inkludera kod från en fil.
- **Notera** att verbatim inte hanterar tabulatortecken korrekt!

```
\usepackage{fancyvrb}  
\fvset{tabsize=4, fontsize=\small}  
\VerbatimInput{Point.scala}
```

```
{\small  
\begin{verbatim}  
class Point(val x: Int, val y: Int):  
  def move(dx: Int, dy: Int): Point =  
    Point(x + dx, y + dy)  
  
\end{verbatim}  
}
```

Programlister med listings

- Paketet `listings` är ett bättre alternativ.
- Stöd för syntaxfärgning och flera programmeringsspråk.
- Många anpassningsmöjligheter: teckensnitt, färger, tabstorlek, ...

```
\lstset{
  language=Scala,
  basicstyle=\ttfamily\small,
  % fler inställningar ...
}

\begin{lstlisting}
class Point(val x: Int, val y: Int):
  def move(dx: Int, dy: Int): Point =
    Point(x + dx, y + dy)
\end{lstlisting}
```

Resultat:

```
class Point(val x: Int, val y: Int):
  def move(dx: Int, dy: Int): Point =
    Point(x + dx, y + dy)
```

Öka eller minska avstånd

- Vertikalt avstånd: `\vspace{längd}`
- Längder kan anges i cm, mm, in, pt, ...
- Negativ längd minskar avståndet.
- Snabbkommandon: `\smallskip`, `\medskip`, `\bigskip`
- Horisontellt avstånd: `\hspace{längd}`

Matematiska formler

$\LaTeX$ är mycket bra på att formatera matematisk text.

- I löpande text: `$ ... $` eller `\( ... \)` (mer robust i vissa sammanhang.)
- På egen rad: `\begin{displaymath} ... \end{displaymath}`
eller `\[ ... \]`
- Numrerad formel: `\begin{equation} ... \end{equation}`
- Etikett och referens: `\label` och `\ref`

Enkla formler

Formeln $x = 3y - 2$ står inne i texten. Däremot står

```
\begin{displaymath}
```

$$x_1 = 3y - 2$$

```
\end{displaymath}
```

för sig själv, precis som $x_2 = -0.6e^{2-y} + \frac{9.3}{\pi}$ och detta är en numrerad ekvation:

```
\begin{equation}
```

$$x_3 = 2y^2 + 7y - 2a + 3$$

```
\label{xochy}
```

```
\end{equation}
```

I ekvation~\ref{xochy} fann vi att \ldots

Formeln $x = 3y - 2$ står inne i texten. Däremot står

$$x_1 = 3y - 2$$

för sig själv, precis som

$$x_2 = -0.6e^{2-y} + \frac{9.3}{\pi}$$

och detta är en numrerad ekvation:

$$x_3 = 2y^2 + 7y - 2a + 3 \quad (1)$$

I ekvation 1 fann vi att ...

Symboler, index

```
\begin{displaymath}
  \alpha \leq \pi \approx 3.141592654
\end{displaymath}
```

$$\alpha \leq \pi \approx 3.141592654$$

```
\[
  x_{k+1} =
  x_k -
  \frac{f(x_k)}{f'(x_k)}
\]
```

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Exponenter, rötter

```
\[
e^x = 1+x+x^2/2!+x^3/3!+\cdots
\]
```

$$e^x = 1 + x + x^2/2! + x^3/3! + \dots$$

```
\[
x_{1,2}=\frac{p}{2}\pm
\sqrt{\frac{p^2}{4}-q}
\]
```

$$x_{1,2} = \frac{p}{2} \pm \sqrt{\frac{p^2}{4} - q}$$

Integraler, summor

```
\[  
  \int_{-\infty}^{\infty}  
    e^{-x^2} dx  
\]
```

$$\int_{-\infty}^{\infty} e^{-x^2} dx$$

```
\[  
  \sum_{k=1}^n \frac{1}{a_k}  
\]
```

$$\sum_{k=1}^n \frac{1}{a_k}$$

Funktioner

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

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

Matriser, parenteser

```
\[
A=\left(
\begin{array}{cccc}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{n1} & a_{n2} & \cdots & a_{nn}
\end{array}
\right)
\]
```

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

Bilder

Inkludera bilder i format: pdf, jpeg, png
Använd paketet `graphicx` (eller `graphics`)

```
\includegraphics[height=30mm]  
  {images/bild.pdf}
```

Bilder med figurmiljö

Slå in bilder i en figure-omgivningen, för att lägga till en bildtext och en etikett.

```
\begin{figure}  
  \includegraphics[height=30mm]  
    {images/enkelmodell.pdf}  
  \caption{Exempelbild}  
  \label{fig:exempelbild}  
\end{figure}
```

Från text kan vi nu referera till figur~\ref{fig:exempelbild}.

Egna kommandon

Man kan lätt definiera egna kommandon, till exempel ett kortare namn för en text som man använder ofta. Kommandon kan ha parametrar.

```
\newcommand{\scala}[1]  
  {\texttt{>{#1}<}}
```

Klasser: `\scala{Random}`, `\scala{Scanner}`, `\scala{FileUtils}` och `\scala{PrintStream}`.

`>Random<`, `>Scanner<`, `>FileUtils<` och `>PrintStream<`

Man kan definiera om existerande kommandon med `\renewcommand`. Det kan ställa till förvirring, så gör inte det.

LaTeX på egen dator

En sammanfattning av LaTeX-installationer finns på www.latex-project.org, sidan Getting LaTeX.

- Linux: Installera LaTeX via den vanliga pakethanteraren (t.ex. apt).
- Mac: Använd MacTeX (baserad på TeXLive, uppdateras årligen).
- Windows: Rekommenderat: TeXLive (www.tug.org/texlive/) för en komplett LaTeX-distribution.
- För redigering, använd moderna verktyg som t.ex. TeXstudio, TeXShop (för Mac), eller VS Code med LaTeX Workshop-tillägget.
- **Notera**, ni får lov att använda Overleaf om ni föredrar det!
- Lunds Universitet erbjuder nu t.o.m. gratis premiumkonto på Overleaf för studenter och anställda vid LU.