

LINKÖPINGS UNIVERSITET

ETT PROJEKT OM

L^AT_EX

JOHANNES DAHLBERG, JOHDA058@STUDENT.LIU.SE
HENRIK RISBERG, HENRI407@STUDENT.LIU.SE
ANDRÉ TEINTANG, ANDTE949@STUDENT.LIU.SE
JOACIM WIELL, JOAVI869@STUDENT.LIU.SE

LINKÖPING

12 OKTOBER 2009

Sammanfattning

L^AT_EX är en vidare utveckling av typsättningspråket TeX och skapades av Leslie Lamport i början av 1980-talet då han ansåg att originalpråket saknade stöd för macron. L^AT_EX används för att skapa textdokument utan att behöva fokusera på typografi, index och referenser.

Till skillnad från andra ordbehandlare skriver man ut koder i L^AT_EX för att sätta typsnitten och som sedan med hjälp av en TeX konverterare översätter koden till ett färdigt PDF eller Postscript.

Innehåll

1	Introduktion	1
1.1	Syfte	1
1.2	Metod & Källor	1
1.3	Struktur	1
2	Vad är L^AT_EX?	2
2.1	Bakgrund	2
2.2	T _E X	2
2.3	Vad används L ^A T _E X till?	3
3	Skriva i L^AT_EX	4
3.1	Vad skriver man i?	4
3.2	Hur skriver man?	5
4	Övningar	6
4.1	Första kommandona	6
4.1.1	Prova på!	7
4.2	Enkel kodning i L ^A T _E X	7
4.2.1	Programmering	7
4.2.2	Källhänvisning	8
4.2.3	Innehållsförteckning och kapitel	8
4.3	Laborationer	9
	Referenser	10
	Bilaga Detta arbete	11

1 Introduktion

1.1 Syfte

Vårt syfte med rapporten är att ge en överblick över vad \LaTeX är, hur det uppstod och hur det kan användas i praktiken. Vi vill även bygga upp en kunskapsgrund inom \LaTeX i form av fakta och grundläggande kommandon.

1.2 Metod & Källor

Vi har valt att utforma vår rapport genom att först lära oss att hantera \LaTeX och sedan använda det vid skrivandet. På det viset så får vi möjlighet att beskriva språket ur en praktiskt användbar synvinkel. Av samma anledning valde vi att göra övningarna sist.

Våra huvudsakliga källor har varit dokumentation kring språket på nätet.

1.3 Struktur

Vi har valt att börja med \LaTeX bakgrund för att på så sätt få en förståelse för dess struktur. Vi har även valt att ta upp de kringliggande faktorerna som krävs för att utnyttja språket, så som redigerare och konverterare.

Sedan kommer vi att gå vidare till lite grundläggande kodning i \LaTeX innan vi summerar det hela i form av några enkla övningar.

2 Vad är L^AT_EX?

2.1 Bakgrund

L^AT_EX skapades av Leslie Lamport i början av 1980-talet. Han planerade att skriva en bok (Great American Concurrency Book).¹ Han arbetade i T_EX men tyckte att han behövde flermacron. De som redan fanns var i hans ögon otillräckliga.

När sedan Don Knuth släppte en tidig version av nuvarande T_EX (T_EX80) skapade Leslie ett paket med macron till TeX vilket blev början till L^AT_EX.² 1993 släpptes nuvarande versionen av L^AT_EX (L^AT_EX2e) och gruppen som har tagit över utvecklingen för L^AT_EX håller på att jobba med den 3:e versionen samtidigt som de underhåller den nuvarande.

2.2 T_EX

T_EX är det språk som hela L^AT_EX bygger på. Det började skrivas av en man vid namn Donald Knuth under sjuttioalet för sedan avstanna 1985. Orsaken till dess skapande var att han blev mer och mer missnöjd med formateringen av hans böcker, "The Art of Computer Programming", volym 1-3. Han ville skapa ett typsättningspråk som kunde bli det ultimata typsättningsverktyget och det anses än idag vara det generellt sett bästa sättet för att typsetta matematiska formler.

T_EX är avsett för att användas direkt av författaren. Den som skapar en text har oftast en förbestämd bild av hur det ska se ut som blir mer och mer otydlig ju fler led den måste passera. Genom att använda T_EX så kan man försäkra sig om att texten blir precis som man vill ha den och att den inte kommer se annorlunda ut bara för att den öppnas med något annat program eller på ett annat system.³

Till skillnad från många WYSIWYG-språk (What You See Is What You Get) så är tex inte alls avsett att visa hela resultatet redan i skrivningsfasen. Tvärt om så ser det väldigt kompakt och oformaterat ut. Det är först när man omvandlar texten till en kompilerad textfil eller pdf som man ser

¹ref: <http://research.microsoft.com/en-us/um/people/lamport/pubs/pubs.html#latex> [1]

²ref: <http://research.microsoft.com/en-us/um/people/lamport/pubs/lamport-latex-interview.pdf> [2]

³ref: <http://www.tug.org> (the T_EXUsers' Group) [6]

slutresultatet av det man har skapat.

2.3 Vad används L^AT_EX till?

L^AT_EX är ett makropaket till TeX som är skapat av Leslie Lamport i början av 1980-talet för att förenkla TeX typografi, i början var de bara ett par makron som han ville att andra skulle kunna använda, med det lede senare till att bli L^AT_EX.⁴

Fokuset i L^AT_EX är att man separerar typografin och textinnehållet, även kallat WYSIWYM⁵. Genom detta, så är det betydligt lättare att skapa en snygg och konsekvent layout i L^AT_EX än i de flesta ordbehandlare. Detta för att man inte måste hålla på att ändra om i texten och finjustera på olika sätt, som att bestämma typsnitt och storlek på överskrifter, välja indentering och numrering etc. Istället skriver man bara, om det nu är det man vill, ”jag vill att den här texten ska påbörja ett nytt kapitel” (`\chapter{’ ’någon text’ ’}`) så kommer L^AT_EX automatiskt att förstora texten, skapa ny sida, numrera kapitlet vid behov etc. Med andra ord så låter T_EX dig fokusera på *vad* du ska skriva istället för *hur* du ska skriva det.⁶

Skalbarheten är hög på grund av just separationen mellan text och layout, så till skillnad ifrån i WYSIWYG⁷- alternativ såsom Microsoft Word och Open Office kan det bli komplicerat att hantera stora texter. Detta gör att mest bokförlag och akademiska⁸ kretsar områden använder L^AT_EX.⁹

För att se L^AT_EX extra kommandon utöver T_EX finns många av dem listade på <http://www.giss.nasa.gov/tools/latex/ltx-2.html>.

⁴ref: <http://research.microsoft.com/en-us/um/people/lamport/pubs/pubs.html#latex> [1]

⁵What You See Is What You Mean

⁶ref: <http://ctan.tug.org/tex-archive/info/lshort/english/lshort.pdf> [5]

⁷What You See Is What You Get

⁸Främst inom naturvetenskapliga för att det är enkelt att skriva formler och ekvationer.

⁹ref: <http://upload.wikimedia.org/wikibooks/en/2/2d/LaTeX.pdf> [7]

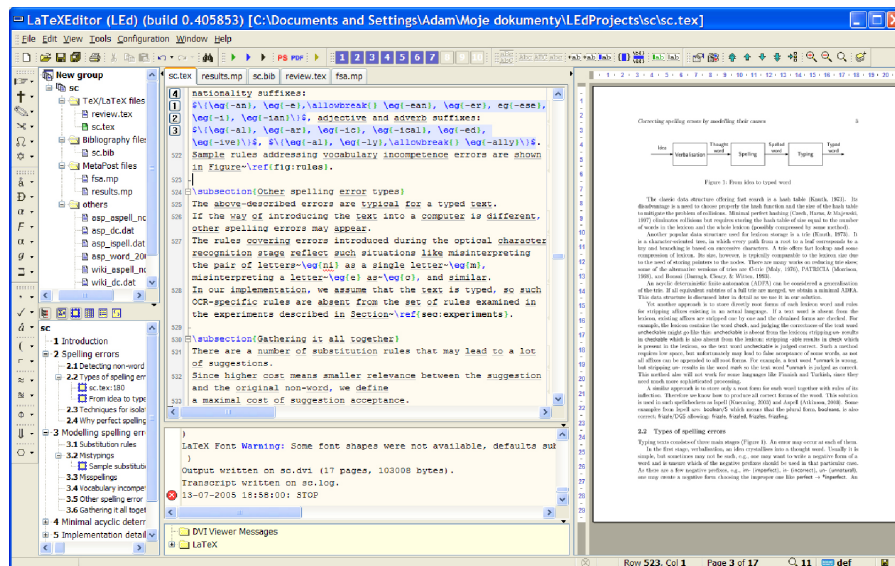
3 Skriva i L^AT_EX

3.1 Vad skriver man i?

Vanligtvis brukar man använda sig av en vanlig ordbehandlare, som Emacs eller Notepad, när man sammanställer råmaterialet av sitt dokument. När man skriver ett L^AT_EX-dokument är det viktigt att spara filen med endelsen .tex för att den sedan ska kunna köras igenom en konverterare som läser om källfilen till ett mer läsvänligt format så som Postscript eller PDF¹⁰.

För att kunna köra filen genom konverteraren så krävs det även att du har laddat ner L^AT_EX-paketen till datorn. Om du använder Linux så rekommenderar vi paketen *texlive-latex-base*, *texlive-latex-recommended* och *texlive-latex-extra*. För windows och OsX så kan allt hämtas som en installationsfil.

Det finns även program som är utvecklade specifikt för att hantera T_EX och L^AT_EX. Exempel på ett sådant program är L^ED (L^AT_EX Editor) som är utvecklat att köras på Windows 95/98/Me/ 2000/2003/Vista och är kompatibel med samtliga TeX distributioner. Fördelen med att använda L^ED framför en vanlig ordbehandlare är de inbyggda funktionerna som medföljer.



Projektbehandlaren underlättar arbetet om man jobbar med ett projekt som består utav flera olika filer och format (ex. T_EX, BibT_EX, MetaPost,

¹⁰ref: <http://scottmcppeak.com/latex/whatislalax.html> [8]

METAFONT, innehållsförteckning, listor, EPS, Tabeller och andra filtyper). Varje kategori grupperas i ett projektträd som gör det enkelt att navigera och växla mellan de olika filerna i projektet.

I själva editorn finns det även ett flertal smarta funktioner. Flikar för att kunna ha flera dokument öppna samtidigt, bokmärken för att underlätta navigation i dokumentet och rättstavningskontroll för att bara nämna några av funktionerna. LEd är också utrustat med en dokumentvisare som förhandsgranskar dokumentet i realtid, vilket ofta är en funktion man kan sakna när man skriver i L^AT_EX¹¹.

3.2 Hur skriver man?

L^AT_EX är, som tidigare nämnt, inte som vanliga ordbehandlade som t.ex. Word, utan i L^AT_EX använder man istället kommandon för att typsättadokument.

```
\documentclass[a4paper]{article}
```

Detta kommando är ett exempel på hur kommandona ser ut och just detta kommandot är för att ställa in vilken typ av dokument man ska skriva, alltså vilken dokumentklass. Nedan kan ni se ett längre exempel på kommando (i L^AT_EX används % för att kommentera bort text):

```
1 \documentclass[titlepage,a4paper,12pt]{article}
2 \usepackage[swedish]{babel}
3 \usepackage[T1]{fontenc}
4 \begin{document} % dokumentet börjar
5 \normalize{Detta
6 \large{är
7 \Large{ett
8 \LARGE{test
9 \huge{av
10 \HUGE{\LaTeX}\footnote{\LaTeX uttalas Laytech}}}}}}
11 \end{document} % dokumentet slutar
```

Denna koden kommer att producera detta (*observera fotnoten längst ner på sidan*) Detta är ett test av L^AT_EX¹²

¹¹ref: http://www.latexeditor.org/more_about.html [9]

¹²L^AT_EX uttalas Laytech

4 Övningar

4.1 Första kommandona

Som tidigare nämnt så krävs det ett par grundläggande kommandon för att skapa ett dokument i L^AT_EX. Det första är:

```
\documentclass[inställningar]{dokumentklass}
```

Det sätter standarden för layout, andra kommandon och funktioner i L^AT_EX. Till exempel så skulle dokumentklassen “book” skapa automatisk sidnumrering, vilket klassen “article” inte gör. Nästa kommando är:

```
\usepackage[paket]{...}
```

Där väljer man olika övergripande paket, tillbehör, som man vill inkludera. Till exempel:

```
\usepackage[latin1]{inputenc}
```

Ovanstående kommando skulle koda texten i latin1¹³. Sedan följer kommandot:

```
\begin{document}
```

Det markerar startpunkten för brödtexten i dokumentet. Slutligen:

```
\end{document}
```

Här slutar dokumentet.

¹³latin1 är samma som ISO-8859-1 (ISO_8859-1:1987), ref: <http://www.iana.org/assignments/character-sets> [10]

4.1.1 Prova på!

Prova att återanvända koden nedan i ett dokument vid namn “helloworld.tex” och se resultatet. Fundera över varför det blev som det blev och försök att få hela “Hello World!” att stå i fetskrift genom att ändra i koden. För att omvandla ett dokument till PDF skriv pdflatex i kommandotolken följt av tex-filens fulla namn. Om du har installerat L^AT_EX på datorn och har emacs så kan du hitta möjligheter att kompilera och förhandsgranska din fil inuti editorn.

```
1 \documentclass{article}
2 \usepackage[latin1]{inputenc}
3 \begin{document}
4 \huge{\textbf{Hello} World!}
5 \end{document}
```

```
<user>$pdflatex helloworld.tex
```

4.2 Enkel kodning i L^AT_EX

4.2.1 Programmering

I L^AT_EX går det att programmera, men det är inte ett helt programmeringsspråk. Det finns booliska värden, if-satser och while-satser. Testa följande kod i Emacs:

```
1 \LaTeX
2 \setboolean{isWord}{false}
3 \setboolean{isOpenOffice}{true}
4 \if{isOpenOffice}
5   this is not...
6 \else
7   \if{isWord}
8     this is not...
9   \else
10    is the \textbf{greatest!}
11  \fi
12 \fi
```

Försök förstå vad som händer i koden. Testa gärna själva att använda if-satser.

4.2.2 Källhänvisning

Det finns även kommandon som hanterar källhänvisningar. `thebibliography`, `bibitem` och `cite`. Dessa kommandon hanterar automatisk uppräknings för varje källa och länkar dem till referenslistan.

```
1 \begin{thebibliography}
2 \bibitem{nyckel_1}
3   Källa 1
4 \bibitem{nyckel_2}
5   Källa 2
6 \end{thebibliography}
```

`thebibliography` skapar referenslistan, och `bibitem` används för att skapa varje referens. För att använda sina referenser använder man kommandot `\cite{nyckel}`. Vill ange flera referenser gör man det genom att separera nycklarna med ett kommatecken `\cite{nyckel_1, nyckel_2}`.

4.2.3 Innehållsförteckning och kapitel

För att skapa och hantera innehållsförteckningen används:

```
\section{Kapitelnamn}
\subsection{Undre kapitelnamn}
\subsubsection{Under undre kapitelnamn} och
\tableofcontents.
```

Kommandot `\tableofcontents` används där man vill placera sin innehållsförteckning. Man kan även själv lägga till onummerade kapitel i innehållsförteckningen med kommandot `\addcontentsline`. Det brukar oftast användas när man vill lägga till t.ex. referenslistan. `\section*` är kommandot för att skapa ett onummerat kapitel. Notera stjärnan på slutet.

```
1 \tableofcontents
2 % Läger till ett onummerat kapitel, det kommer inte
3 % att läggas till innehållsförteckningen
4 \section*{Bilaga 1 källkod på rapport}
5 % Läger till ett kapitel med namet "Bilagor"
6 % toc = Table Of Content
7 \addcontentsline{toc}{section}{Bilagor}
```

4.3 Laborationer

För att lösa dessa uppgifterna använd allt vi tidigare nämnt samt se igenom koden till detta arbetet. Om du inte hittar svaret ändå så använd *The Not So Short Introduction to L^AT_EX₂e* (<http://ctan.tug.org/tex-archive/info/lshort/english/lshort.pdf>), eller NASA:s hemsida (<http://www.giss.nasa.gov/tools/latex/ltx-2.html>)

Testa att göra följande:

- Skapa en framsida med hjälp av de inbyggda kommandona.
- Skapa en innehållsförteckning.
- Använd kommandot `\section` för att skapa en rubrik.
- Använd kommandot `\footnote{}` för att skapa en fotnot i texten.

Avancerade uppgifter:

- Ta bort sidnumreringen på förstasidan och på innehållsförteckningen och se till att innehållsförteckningen visar rätt sidhänvisning.
- Skapa underrubriker till till den första skapade rubriken.
- Testa att använda kommandot `verbatim` och `verb`. Vad är det för skillnad mellan dessa två?

Referenser

- [1] <http://research.microsoft.com/en-us/um/people/lamport/pubs/pubs.html> <2009-09-07>
- [2] <http://research.microsoft.com/en-us/um/people/lamport/pubs/lamport-latex-interview.pdf> <2009-09-07>
- [3] <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=LaTeX3> <2009-10-05>
- [4] http://en.wikibooks.org/wiki/LaTeX/Introduction#What_is_LaTeX <2009-10-05>
- [5] <http://ctan.tug.org/tex-archive/info/lshort/english/lshort.pdf> <2009-10-05>
- [6] <http://www.tug.org> <2009-10-05>
- [7] <http://upload.wikimedia.org/wikibooks/en/2/2d/LaTeX.pdf> <2009-10-05>
- [8] <http://scottmcppeak.com/latex/whatislatex.html> <2009-10-05>
- [9] http://www.latexeditor.org/more_about.html <2009-10-05>
- [10] <http://www.iana.org/assignments/character-sets> <2009-10-05>

Jacobsson, Per (2004) *Introduktion till L^AT_EX*. Studentlitteratur

Bilaga Detta arbete

```
\documentclass[pdftex,12pt]{article}
\usepackage[swedish]{babel}
\usepackage[T1]{fontenc}
\usepackage{graphicx}
\usepackage{appendix}
\usepackage[pdftitle={LaTeX}, pdfborder={0 0 0}]{hyperref}
\usepackage{listings}
\usepackage{verbatim}
\newcommand {\HRule}{\rule{\linewidth}{0,5mm}}
\title{\LaTeX}
\author{Johannes Dahlberg, johda058@student.liu.se\\
Henrik Risberg, henri407@student.liu.se\\
André Teintang, andte949@student.liu.se\\
Joacim Wiell, joavi869@student.liu.se\\
Linköpings Universtitet\\
Linköping\\
2009-09-23}
\begin{document}
\begin{titlepage}
\begin{center}

\textsc{\LARGE Linköpings Universtitet}\\[1,5cm]
\textsc{\Large Ett projekt om}\\[0,5cm]

\HRule\\[0,4cm]
{ \Huge \bfseries \LaTeX}\\[0,4cm]
\HRule\\[1,5cm]

\textsc{Johannes Dahlberg, johda058@student.liu.se\\
Henrik Risberg, henri407@student.liu.se\\
André Teintang, andte949@student.liu.se\\
Joacim Wiell, joavi869@student.liu.se}\\[0,5cm]
Linköping}\\[0,5cm]
\today}

\end{center}
\end{titlepage}
\end{document}
```

```
\end{titlepage}
\section*{Sammanfattning}
\LaTeX{} är en vidare utveckling av typsättningsspråket TeX och
skapades av Leslie Lamport i början av 1980-talet då han
ansåg att originalspråket saknade stöd för macron. \LaTeX{}
används för att skapa textdokument utan att behöva fokusera
på typografi, index och referenser.
```

Till skillnad från andra ordbehandlare skriver man ut koder i \LaTeX{} för att sätta typsnitten och som sedan med hjälp av en TeX konverterare översätter koden till ett färdigt PDF eller Postscript.

```
\thispagestyle{empty}
```

```
\newpage
\setcounter{page}{0}
\thispagestyle{empty}
\lstset{numbers=left, language=TeX, frame=single}
\tableofcontents
\vfill
\newpage
\section{Introduktion}
```

```
\subsection{Syfte}
```

Vårt syfte med rapporten är att ge en överblick över vad \LaTeX{} är, hur det uppstod och hur det kan användas i praktiken.

Vi vill även bygga upp en kunskapsgrund inom \LaTeX{} i form av fakta och grundläggande kommandon.

```
\subsection{Metod & Källor}
```

Vi har valt att utforma vår rapport genom att först lära oss att hantera \LaTeX{} och sedan använda det vid skrivandet.

På det viset så får vi möjlighet att beskriva språket ur en praktiskt användbar synvinkel. Av samma anledning valde vi att göra övningarna sist.

Våra huvudsakliga källor har varit dokumentation kring språket på nätet.

```
\subsection{Struktur}
```

Vi har valt att börja med \LaTeX{} bakgrund för att på så sätt få en förståelse för dess struktur. Vi har även valt att ta upp de

kringliggande faktorerna som krävs för att utnyttja språket, så som redigerare och konverterare.

Sedan kommer vi att gå vidare till lite grundläggande kodning i `\LaTeX{}` innan vi summerar det hela i form av några enkla övningar.
`\newpage`

`\section{Vad är \LaTeX{}}?`

`\subsection{Bakgrund}`

`\LaTeX{}` skapades av Leslie Lamport i början av 1980-talet. Han planerade att skriva en bok (Great American Concurrency Book).
`\footnote{ref: http://research.microsoft.com/en-us/um/people/lamport/pubs/pubs.html\#latex \cite{msoft_pubs}}` Han arbetade i `\TeX{}` men tyckte att han behövde flermacron. De som redan fanns var i hans ögon otillräckliga.

När sedan Don Knuth släppte en tidig version av nuvarande `\TeX{}` (`\TeX{}`80) skapade Leslie ett paket med macron till TeX vilket blev början till `\LaTeX{}`.
`\footnote{ref: http://research.microsoft.com/en-us/um/people/lamport/pubs/lamport-latex-interview.pdf \cite{msoft_interview}}` 1993 släpptes nuvarande versionen av `\LaTeX{}` (`\LaTeX2e`) och gruppen som har tagit över utvecklingen för `\LaTeX{}` håller på att jobba med den 3:e versionens samtidigt som de underhåller den nuvarande.

`\subsection{\TeX{}}`

`\TeX{}` är det språk som hela `\LaTeX{}` bygger på. Det började skrivas av en man vid namn Donald Knuth under sjuttio-talet för sedan avstanna 1985. Orsaken till dess skapande var att han blev mer och mer missnöjd med formateringen av hans böcker, ''The Art of Computer Programming'', volym 1-3. Han ville skapa ett typsättningspråk som kunde bli det ultimata typsättningsverktyget och det anses än idag vara det generellt sett bästa sättet för att typsetta matematiska formler.

`\TeX{}` är avsett för att användas direkt av författaren. Den som skapar en text har oftast en förbestämd bild av hur det ska se ut som blir mer och mer otydlig ju fler led den måste passera. Genom att använda `\TeX{}` så kan man försäkra sig om att texten blir precis som man vill ha den och att den inte

kommer se annorlunda ut bara för att den öppnas med något annat program eller på ett annat system. `\footnote{ref: http://www.tug.org (the \TeX Users' Group) \cite{tug}}`

Till skillnad från många WYSIWYG-språk (What You See Is What You Get) så är tex inte alls avsett att visa hela resultatet redan i skrivningsfasen. Tvärt om så ser det väldigt kompakt och oformaterat ut. Det är först när man omvandlar texten till en kompilerad textfil eller pdf som man ser slutresultatet av det man har skapat.

`\subsection{Vad används \LaTeX{} till?}`
`\LaTeX` är ett makropaket till TeX som är skapat av Leslie Lamport i början av 1980-talet för att förenkla TeX typografi, i början var de bara ett par makron som han ville att andra skulle kunna använda, med det lede senare till att bli `\LaTeX`. `\footnote{ref: http://research.microsoft.com/en-us/um/people/lamport/pubs/pubs.html\#latex \cite{msoft_pubs}}`

Fokuset i `\LaTeX{}` är att man separerar typografin och textinnehållet, även kallat WYSIWYM`\footnote{What You See Is What You Mean}`. Genom detta, så är det betydligt lättare att skapa en snygg och konsekvent layout i `\LaTeX{}` än i de flesta ordbehandlare. Detta för att man inte måste hålla på att ändra om i texten och finjustera på olika sätt, som att bestämma typsnitt och storlek på överskrifter, välja indentering och numrering etc. Istället skriver man bara, om det nu är det man vill, ''jag vill att den här texten ska påbörja ett nytt kapitel'' (`\verb|\chapter{''någontext''|}`) så kommer `\LaTeX{}` automatiskt att förstora texten, skapa ny sida, numrera kapitlet vid behov etc. Med andra ord så låter `\TeX{} dig fokusera på \emph{vad} du ska skriva istället för \emph{hur} du ska skriva det.`
`\footnote{ref: http://ctan.tug.org/tex-archive/info/lshort/english/lshort.pdf \cite{tex_archive}}`

Skalbarheten är hög på grund av just separationen mellan text och layout, så till skillnad ifrån i WYSIWYG`\footnote{What You Se Is What You Get}`-alternativ såsom Microsoft Word och Open Office kan det bli komplicerat att hantera stora texter. Detta gör att mest bokförlag och akademiska `\footnote{Främst inom naturvetenskapliga för att det är enkelt att skriva formler och ekvationer.}` kretsar områden använder `\LaTeX{}.`
`\footnote{ref: http://upload.wikimedia.org/wikibooks/en/2/2d/LaTeX.pdf}`

`\cite{wiki_books_2}}`

För att se `\LaTeX{}` extra kommandon utöver `\TeX{}` finns många av dem listade på <http://www.giss.nasa.gov/tools/latex/ltx-2.html>.

`\newpage`

`\section{Skriva i \LaTeX{}}`

`\subsection{Vad skriver man i?}`

Vanligtvis brukar man använda sig av en vanlig ordbehandlare, som Emacs eller Notepad, när man sammanställer råmaterialet av sitt dokument. När man skriver ett `\LaTeX{}`-dokument är det viktigt att spara filen med endelsen `.tex` för att den sedan ska kunna köras igenom en konverterare som läser om källfilen till ett mer läsvänligt format så som Postscript eller PDF `\footnote{ref: http://scottmcppeak.com/latex/whatislatex.html}`

`\cite{scott_peak}}`.

För att kunna köra filen genom konverteraren så krävs det även att du har laddat ner `\LaTeX`-paketen till datorn. Om du använder Linux så rekommenderar vi paketen `\textit{texlive-latex-base, texlive-latex-recommended}` och `\textit{texlive-latex-extra.}` För windows och OsX så kan allt hämtas som en installationsfil. \\

Det finns även program som är utvecklade specifikt för att hantera `\TeX{}` och `\LaTeX{}`. Exempel på ett sådant program är LEd (`\LaTeX{}` Editor) som är utvecklat att köras på Windows 95/98/Me/2000/2003/Vista och är kompatibel med samtliga TeX distributioner. Fördelen med att använda LEd framför en vanlig ordbehandlare är de inbyggda funktionerna som medföljer. \\

`\includegraphics[scale=0.2]{screenshot.jpg}`

Projektbehandlaren underlättar arbetet om man jobbar med ett projekt som består utav flera olika filer och format (ex. `\TeX{}`, BibTeX, MetaPost, METAFONT, innehållsförteckning, listor, EPS, Tabeller och andra filtyper). Varje kategori grupperas i ett projektträd som gör det enkelt att navigera och växla mellan de olika filerna i projektet.

I själva editorn finns det även ett flertal smarta funktioner. Flikar för att

kunna ha flera dokument öppna samtidigt, bokmärken för att underlätta navigation i dokumentet och rättstavningskontroll för att bara nämna några av funktionerna. LEd är också utrustat med en dokumentvisare som förhandsgranskar dokumentet i realtid, vilket ofta är en funktion man kan sakna när man skriver i `\LaTeX\footnote{ref: http://www.latexeditor.org/more/_about.html \cite{latex_editor}}`.

```
\subsection{Hur skriver man?}
```

`\LaTeX{}` är, som tidigare nämnt, inte som vanliga ordbehandlade som t.ex. Word, utan i `\LaTeX{}` använder man istället kommandon för att typsättadokument.

```
\begin{verbatim}
```

```
\documentclass[a4paper]{article}
```

```
\end{verbatim}
```

Detta kommando är ett exempel på hur kommandona ser ut och just detta kommandot är för att ställa in vilken typ av dokument man ska skriva, alltså vilken dokumentklass. Nedan kan ni se ett längre exempel på kommando (i `\LaTeX{}` används `\%` för att kommentera bort text):

```
\begin{lstlisting}
```

```
\documentclass[titlepage,a4paper,12pt]{article}
```

```
\usepackage[swedish]{babel}
```

```
\usepackage[T1]{fontenc}
```

```
\begin{document} % dokumentet börjar
```

```
\normalsize{Detta
```

```
\large{är
```

```
\Large{ett
```

```
\LARGE{test
```

```
\huge{av
```

```
\HUGE{\LaTeX{\footnote{\LaTeX{} uttalas Laytech}}}}}
```

```
\end{document} % dokumentet slutar
```

```
\end{lstlisting}
```

Denna koden kommer att producera detta (`\emph{observera fotnoten längst ner på sidan}`)

```
\normalsize{Detta
```

```
\large{är
```

```
\Large{ett
```

```
\LARGE{test
```

```
\huge{av
```

```
\LaTeX{\footnote{\LaTeX{} uttalas Laytech}}}}}  
\newpage
```

```
\section{Övningar}  
\subsection{Första kommandona}  
Som tidigare nämnt så krävs det ett par grundläggande kommandon  
för att skapa ett dokument i \LaTeX{}. Det första är:\\  
\\verb|\documentclass[inställningar]{dokumentklass}|\\  
\\Det sätter standarden för layout, andra kommandon och funktioner i  
\LaTeX{}. Tillexempel så skulle dokumentklassen ‘‘book’’ skapa automatisk  
sidnumrering, vilket klassen ‘‘article’’ inte gör. Nästa kommando är:\\  
\\verb|\usepackage[paket]{...}|\\  
\\Där väljer man olika övergripande paket, tillbehör, som man vill inkludera.  
Till exempel:\\  
\\verb|\usepackage[latin1]{inputenc}|\\  
\\Ovanstående kommando skulle koda texten i latin1\footnote{latin1 är  
samma som ISO-8859-1 (ISO\_8859-1:1987), ref: http://www.iana.org/  
assignments/character-sets \cite {iana}}.  
Sedan följer kommandot:\\  
\\verb|\begin{document}|\\  
\\Det markerar startpunkten för brödtexten i dokumentet. Slutligen:\\  
\\verb|\end{document}|\\  
\\Här slutar dokumentet.\\  
\newpage
```

```
\subsubsection{Prova på!}  
Prova att återanvända koden nedan i ett dokument vid namn  
‘‘helloworld.tex’’ och se resultatet. Fundera över varför det blev som  
det blev och försök att få hela ‘‘Hello World!’’ att stå i fetskrift  
genom att ändra i koden. För att omvandla ett dokument till PDF skriv  
pdflatex i kommandotolken följt av tex-filens fulla namn.  
Om du har installerat \LaTeX{} på datorn och har emacs  
så kan du hitta möjligheter att kompilera och förhandsgranska  
din fil inuti editorn.  
\begin{lstlisting}  
\documentclass{article}  
\usepackage[latin1]{inputenc}  
\begin{document}
```

```
\huge{\textbf{Hello} World!}  
\end{document}  
\end{lstlisting}  
\verb|<user>$pdflatex helloworld.tex|
```

```
\subsection{Enkel kodning i \LaTeX{}}  
\subsubsection{Programmering}  
I \LaTeX{} går det att programmera, men det är inte ett helt  
programmeringsspråk.
```

Det finns booliska värden, if-satser och while-satser.

Testa följande kod i Emacs:

```
\begin{lstlisting}  
\LaTeX  
\setboolean{isWord}{false}  
\setboolean{isOpenOffice}{true}  
\if{isOpenOffice}  
  this is not...  
\else  
  \if{isWord}  
    this is not...  
  \else  
    is the \textbf{greatest!}  
  \fi  
\fi  
\end{lstlisting}
```

Försök förstå vad som händer i koden. Testa gärna själva att använda if-satser.

```
\newpage
```

```
\subsubsection{Källhänvisning}
```

Det finns även kommandon som hanterar källhänvisningar.

{\tt thebibliography}, {\tt bibitem} och {\tt cite}.

Dessa kommandon hanterar automatisk uppräknings för varje källa och länkar dem till referenslistan.

```
\begin{lstlisting}  
\begin{thebibliography}  
\bibitem{nyckel_1}
```

```

Källa 1
\bibitem{nyckel_2}
Källa 2
\end{thebibliography}
\end{lstlisting}

```

\noindent{\tt thebibliography} skapar referenslistan, och {\tt bibitem} används för att skapa varje referens.

För att använda sina referenser använder man kommandot {\tt \$\backslash\$cite\{\{\it nyckel\}\}}. Vill ange flera referenser gör man det genom att separera nycklarna med ett kommatecken {\tt \$\backslash\$cite\{\{\it nyckel\$_\$1}, {\it nyckel\$_\$2}\}}.

```
\subsubsection{Innehållsförteckning och kapitel}
```

För att skapa och hantera innehållsförteckningen används:\\\
{\tt \$\backslash\$section\{\{\it Kapitelnamn\}\}}\\
{\tt \$\backslash\$subsection\{\{\it Undre kapitelnamn\}\}}\\
{\tt \$\backslash\$subsubsection\{\{\it Under undre kapitelnamn\}\}} och\\
{\tt \$\backslash\$tableofcontents}.\

\noindent Kommandot {\tt \$\backslash\$tableofcontents} används där man vill placera sin innehållsförteckning.

Man kan även själv lägga till onummerade kapitel i innehållsförteckningen med kommandot {\tt \$\backslash\$addcontentsline}.

Det brukar oftast användas när man vill lägga till t.ex. referenslistan. {\tt \$\backslash\$section*} är kommandot för att skapa ett onummerat kapitel. Notera stjärnan på slutet.

```

\begin{lstlisting}
\tableofcontents
% Lägger till ett onummerat kapitel, det kommer inte
% att läggas till innehållsförteckningen
\section*{Bilaga 1 källkod på rapport}
% Lägger till ett kapitel med namet "Bilagor"
% toc = Table Of Content
\addcontentsline{toc}{section}{Bilagor}
\end{lstlisting}

```

```

\subsection{Laborationer}
För att lösa dessa uppgifterna använd allt vi tidigare nämnt samt
se igenom koden till detta arbetet. Om du inte hittar svaret ändå så använd
The Not So Short Introduction to \LaTeX2e
(http://ctan.tug.org/tex-archive/info/lshort/english/lshort.pdf),
eller NASA:s hemsida (http://www.giss.nasa.gov/tools/latex/ltx-2.html)\\
Testa att göra följande:
\begin{itemize}
\item Skapa en framsida med hjälp av de inbyggda kommandona.
\item Skapa en innehållsförteckning.
\item Använd kommandot \verb|\section| för att skapa en rubrik.
\item Använd kommandot \verb|\footnote{| för att skapa en fotnot i texten.
\end{itemize}
Avancerade uppgifter:
\begin{itemize}
\item Ta bort sidnumreringen på förstasidan och på innehållsförteckningen och se till
att innehållsförteckningen visar rätt sidhänvisning.
\item Skapa underrubriker till till den första skapade rubriken.
\item Testa att använda kommandot verbatim och verb. Vad är det för skillnad mella
\end{itemize}
\newpage

\begin{thebibliography}{widest-label}
\addcontentsline{toc}{section}{Referenser}
\small{
\bibitem{msoft_pubs}
http://research.microsoft.com/en-us/um/people/lamport/pubs/pubs.html <2009-09-07>
\bibitem{msoft_interview}
http://research.microsoft.com/en-us/um/people/lamport/pubs/lamport-latex-interview
\bibitem{latex}
http://www.tex.ac.uk/cgi-bin/texfaq2html?label=LaTeX3 <2009-10-05>
\bibitem{wiki_books}
http://en.wikibooks.org/wiki/LaTeX/Introduction\#What\\_is\\_LaTeX <2009-10-05>
\bibitem{tex_archive}
http://ctan.tug.org/tex-archive/info/lshort/english/lshort.pdf <2009-10-05>
\bibitem{tug}
http://www.tug.org <2009-10-05>
\bibitem{wiki_books_2}

```

```

http://upload.wikimedia.org/wikibooks/en/2/2d/LaTeX.pdf <2009-10-05>
\bibitem{scott_peak}
http://scottmcpeak.com/latex/whatislatex.html <2009-10-05>
\bibitem{latex_editor}
http://www.latexeditor.org/more\_about.html <2009-10-05>
\bibitem{iana}
http://www.iana.org/assignments/character-sets} <2009-10-05>
\\
Jacobsson, Per (2004) \emph{Introduktion till \LaTeX{}}. Studentlitteratur
\end{thebibliography}
\newpage

\appendix

\noappendicestocpagenum
\section*{Bilaga Detta arbete}
\addcontentsline{toc}{section}{Bilaga Detta arbete}
\verbatiminput{LaTeX.tex}
\end{document}

```