

Om L^AT_EX och prov 1 i momentet *D. Texter och L^AT_EX* i kursen D0015E

Håkan Jonsson

Institutionen för system- och rymdteknik
Luleå tekniska universitet

16 september 2020

Sammanfattning

Prov 1 går ut på att skapa ett dokument identiskt med ett annat med hjälp av L^AT_EX. Här ges en förklaring till innehållet i den startfil du får.

1 Allmänt

L^AT_EX erbjuder fantastiska möjligheter att producera professionellt typsatt text, men också lika stora möjligheter att göra fel. Du måste därför vara *mycket noggrann* då du skriver ditt L^AT_EX-manus. Minsta tecken har betydelse och även små fel, som att råka använda en stor bokstav där det ska vara en liten, gör att varje försök att generera ett dokument från manuset rasar samman som ett korthus.

Skriv manuset i mycket små steg och generera dokument mellan stegen, så får du eventuella fel i småportioner och kan enklare åtgärda dem. Jag gör alltid så här och jag råder även dig att göra det.

1.1 Layout

Skilj mellan *innehåll* och *utseende*. Koncentrera dig på innehållet, dvs att text och formler får korrekt innehåll, och överlåt sen åt L^AT_EX att bestämma

utseendet. Kanske detta känns ovant, för WYSIWYG¹-program som t ex Microsoft Word bygger ju på att skribenter också bestämmer layouten. Men L^AT_EX är inte något WYSIWYG-program. L^AT_EX är programmerat med hjälp av de bästa layoutreglerna och layoutar bättre än oss alla i 99 fall av 100. Om du trots det tycker att layouten inte är bra så vänta med uttryckliga ändringar av layouten tills du fått allt innehåll korrekt(!) Annars skapar du bara en massa problem för dig själv. Även små ändringar av innehållet kan orsaka mycket stora ändringar av hur L^AT_EX utformar dokumentets layout.

1.2 Kommentarer

L^AT_EX-systemet ignorerar allt på en rad efter ett procenttecken (%). Med % kan man således inkludera kommentar i sitt manus som inte påverkar det färdiga dokumentet men som kan vara viktiga för någon som läser manuset. Om du kikar på filen `koch-first.tex`, ett inledande men ofullständigt manus, ser du många sådana kommentarer som inkluderats som en hjälp till dig. Var och en av dem listar nämligen något som ska vara med i ditt färdiga manus och ungefär på den plats kommentaren förekommer.

1.3 Kommandon och omgivningar

Ett L^AT_EX-kommando startar med ett bakåt-snedstreck (\). Sen följer kommandots namn och eventuella argument². Vill man t ex ha **text med fet stil** använder man kommandot `\textbf{text med fet stil}`. Vill man istället att *text ska framhävas* så skriver man `\emph{text ska framhävas}`. Det finns väldigt många olika kommandon och ett råd är att snarare googla efter/slå upp detaljer då du behöver dem än att lära dig dem alla utan till.

Förutom kommandon finns det även L^AT_EX-omgivningar (*environments* på engelska). Dessa inleds med `\begin`, avslutas med `\end` och påverkar det som skriv mellan `begin-end`. Exempel: Om man vill

centrera text

använder man omgivningen `center` och skriver

```
\begin{center}
  centrera text
\end{center}
```

Om du läser manuset till detta dokument kommer du att se att jag använt omgivningen `verbatim` för att få in centreringsomgivningen ovan i min text.

¹What-You-See-Is-What-You-Get.

²Jämför med hur funktioner och metoder används i `python`.

1.4 Tomrum

En tomrad bland text markerar nytt stycke. Detta kan märkas genom att första raden dras in eller att \LaTeX lägger in lite tomrum vertikalt mellan styckena. Var noga med eventuella tomrader du väljer att ta med i ditt manus. Däremot spelar godtyckligt många mellanslag mellan ord i text ingen annan roll än att de byts ut mot ett lagom stort mellanrum oavsett hur många de är. Därför kan man i regel fritt *indentera rader*, dvs stoppa in mellanslag i början av rader för att visa på hur på varandra följande rader hör ihop, och därigenom dramatiskt öka manusets läsbarhet. För din egen skull - gör det!

Jag brukar indentera sånt som är inne i en omgivning minst 2 mellanslag relativt närmast omgivande omgivning. Det gör att jag enklare ser vad som tillhör vad. Om jag inte hinner indentera när jag skriver så återvänder jag till texten och gör det senare. Jag råder dig att indentera på samma sätt som mig, och även då du programmerar datorer. I vissa språk, t ex `python`, har indenteringen även betydelse för slutresultatet.

2 Om prov 1

Här följer en genomgång av innehållet i `koch-first.tex` med kommentarer.

2.1 Inledningen

Varje manus inleds med lite allmänna deklARATIONER och importer.

- `\documentclass[12pt,a4paper]{article}` säger att detta är ett manus för en artikel som ska typsättas med 12 punkters typsnitt och passa på ett A4-ark.
- `\usepackage[utf8]{inputenc}` anger den teckenkodning som används. Detta är datorspecifikt så du kan behöva ändra `uft8` till något annat t ex `latin1`.
`\usepackage` är ett kommando som importerar andra kommandon och omgivningar än de som det grundläggande \LaTeX -systemet består av. Kommandot gör det importerade tillgängligt när man skriver manuset.
- `\usepackage{graphicx}` gör det möjligt att inkludera grafik, t ex JPEG-bilder eller PDF-filer. Detta gör man i praktiken sen med kommandot `\includegraphics`.

- `\usepackage{amsmath, amsthm, amssymb}` utökar stödet för matematik.
- `\newtheorem{theorem}{Theorem}` introducerar `theorem`, en ny omgivning för matematiska satser/teorem.

Inledningen kan även spalta upp titel, författarnamn, datum, rubrikstilar och annat. Så görs i manuset för det dokument du just nu läser men inte i `koch-first.tex`.

2.2 Själv dokumentet

Dokumentdelen av manuset består av omgivningen `document`. Allt som skrivs inom

```
\begin{document}
```

```
\end{document}
```

hamnar i det färdiga dokumentet. Text som skrivs in hamnar i dokumentet i den ordning den skrivs (uppfifrån och ned, vänster till höger). Kommandon och omgivningar säger om text är något speciellt som till exempel rubriker, fotnoter, matematiska formler mm. \LaTeX väljer ett sätt att typsätta på, och bestämma layouten på, som passar texten och den pappersstorlek man angett i inledningen.

- `\section` säger att här ska det in en avsnittsrubrik. Argumentet till kommandot, i detta fall `Headline...`, är rubriktexten. \LaTeX kommer att välja en lagom storlek på rubriken (och övrig text) som passar till det slutgiltiga dokumentets storlek.
- `\emph` har vi gått igenom ovan på sidan 2.
- `\cite` används för att referera till en referens, som beskrivs i omgivningen `thebibliography` i slutet av manuset. En referens är typiskt en artikel eller rapport. Här refererar vi referensen `koch`, vilket är varför vi skriver `\cite{koch}`. \LaTeX kommer att byta ut `\cite{koch}` mot ett löpnummer inom hakparenteser.

Har man flera referenser ger \LaTeX varje ett eget nummer och ser till att utbytena blir rätt. Detta är mycket smidigt eftersom man inte själv behöver hålla reda på löpnumren, utan kan helt överlåta detta åt \LaTeX -systemet. Däremot måste man (förstås) själv definiera referenserna och ge dem unika nyckelord (i detta fall `koch`).

- `figure` är en omgivning för figurer som man vill ha bildtext till, t ex bilder. `[h]` anger att vi vill ha figuren just **h**är, och inte någon annanstans(!) Beroende på \LaTeX :s layoutregler *kan* figuren ändå hamna på annan plats, vilket man då kan åtgärda. Men sådan åtgärd ska man alltså vidta först när allt annat innehåll är korrekt (inte omedelbart när man till sin fasa ser att figuren hamnar på fel ställe).
- `\label` är en etikett för något i dokumentet som man sen vill kunna referera till som avsnitt, ekvationer, sidor, bilder, tabeller mm. Till `\label` fogar man ett unikt namn på etiketten som man sen känner igen. En etikett för ett inledande avsnitt skulle kunna vara `sec:intro` medan man för en ekvation kanske använder `eq:pythagoras`. De här prefixen `sec:` och `eq:` är inget \LaTeX -systemet kräver, utan bara till för att man snabbt ska se vad för något en referens leder till. För en figur brukar jag skriva t ex `\label{fig:circle}`, där `fig:` gör det lätt att skilja etiketten från etiketter för avsnitt och ekvationer. Du får döpa dina etiketter som du vill men håll dig till vanliga bokstäver i etikettnamnen.

Avsnitt, ekvationer, figurer och annat numreras separat inom respektive sort och stigande. Det går dock att ändra detta (men det behöver man sällan).

- `\centering` inne i en `figure` ger centrering av figuren. Omgivningen `center` fungerar likartat men ska inte användas just här.
- Kommandot `\includegraphics` säger att här ska grafik in från en fil vars namn ett argument anger. Vår fil heter `snowflake.jpg` och dessutom föreskriver vi att vi vill att bilden får en bredd på 9 cm. Detta behövs egentligen inte, men då får bilden sin naturliga storlek som kan vara direkt olämplig.

Bilder är något \LaTeX inte klarar av särskilt väl. Bildhantering i \LaTeX är krångligt, blir ofta fel, och därför har jag inkluderat hela det kommando du ska använda för att få in bilden på snöflingans första konstruktionssteg.

- `\caption` anger en bildtext.
- Omgivningen `quote` lägger in ett längre citat, och indenterar från både höger och vänster.
- `\textit` ger kursiv stil.

- Kommandot `\ref` används för att referera till något i dokumentet som märkts ut med en etikett. Om man t ex lagt in en (1) figur i sitt dokument, och i denna inkluderat etiketten `\label{fig:snowflakes}`, så kommer `\ref{fig:snowflakes}` i texten sen att bytas ut mot en 1:a. Skulle figuren vara den andra i ordning, sker utbytet istället till en 2:a.
- `\subsection` fungerar som `\section` men används för rubriknivån omedelbart under den översta. I \LaTeX finns även ytterligare en nivå, som man får med `\subsubsection`.
- Omgivningen `theorem` typsätter en typisk matematisk sats. Fyll i med påståendet i satsen mellan `begin` och `end`.
- `proof` används för bevis. I ditt manus ska du skriva kod för två bevis, som alltså läggs in i omgivningen `proof`. Notera att `proof` automatiskt lägger till en passande slutsymbol.
- Matematisk löptext märks i \LaTeX ut genom att omges av dollartecken (\$). Skriver man t ex `$f(x)+f(2x)/2+f(3x)/3$` typsätts det som matematisk text, dvs

$$f(x) + f(2x)/2 + f(3x)/3,$$

och inte som vanlig text, dvs `f(x)+f(2x)/2+f(3x)/3`. Lägg noga märke till stilskillnaden. Det är lika fel att typsätta matematiska formler som vanlig löptext som att säga att löptext ska se ut som matematisk text. I prov 1 finns flera tillfällen när det gäller att välja rätt typ av text.

(Matematiska formler kan även märkas ut på andra sätt.)

- `\Delta` ger den grekiska bokstaven stora D (dvs Δ).
- `N_i` ger i matematisk text N_i , dvs understrykningstecken ger (i matematiska formler) ett index.
- `displaymath` ger en formel som typsätts fritt i ett större vertikalt mellanrum. Man behöver inga dollartecken i omgivningen, utan \LaTeX förutsätter att det som skrivs där måste vara matematik.
Samma matematiska text kan i regel antingen typsättas i löptext, med dollartecken, eller fritt med bl a `displaymath`; se även `equation` nedan.
- `cases` används för ekvationer med många fall.

- Skriver man `\text{en mening}` i matematiska formler typsätts `en mening` ändå som vanlig löptext. Detta är användbart när man vill lägga till förklaringar.
- `N_{n-1}` ger ett mera komplicerat index på N . Klammerparenteser kan alltid användas för att gruppera ihop. Skriver man t ex

$$B_{N_{i+1}+M_{j+2i}}$$

får man

$$B_{N_{i+1}+M_{j+2i}}.$$

På samma sätt kan t ex omfattande exponenter och integralgränser skapas.

- `equation` används som `displaymath` men ger automatiskt ett ekvationsnummer som skrivs ut i marginalen. Kodan

```
\begin{equation}
  \label{eq:first}
  f(x) = x^2 + 2x + 9
\end{equation}
```

ger t ex

$$f(x) = x^2 + 2x + 9 \tag{1}$$

och refererar man sen till etiketten `eq:first` med `Eq.~\ref{eq:first}` gör \LaTeX om detta till

Eq. 1

Den ”våg” (`~`) som syns i koden kallas *tilde* och är ett fast mellanslag, ett slags klister som beordrar \LaTeX att behandla det ihopklistrade som ett ord. Detta används för att hindra radbrytning mellan Eq. och siffran 1. I just detta sammanhang får radbrytning inte ske, och \LaTeX måste då upplysas om det.

- `\cdot` ger en centrerad punkt i matematik. En sådan brukar användas för att visa multiplikation mellan två tal eller där man annars vill trycka på att det verkligen är multiplikation. Annars ska man aldrig använda en punkt för multiplikation. Alltså, man skriver $f(x)g(x)$ och inte $f(x) \cdot g(x)$. Däremot måste man skriva $3 \cdot 4$ för annars blir det 34, dvs ett tal och inte en multiplikation.

- Tecknet `^` ger ”upphöjt i”, dvs en exponent i matematisk text. Exempel: Att skriva `$x^{f(x)}$` ger $x^{f(x)}$. Notera användningen av klammerparenteser för att gruppera ihop hela exponenten på samma sätt som de tidigare höll samman större index (se sidan 2.2). Har man en enkel exponent behövs inga klammerparenteser. För att få x^2 räcker det att skriva `x^2`.

- `\frac` används för bråk. Exempel: `\frac{L_{n-1}}{3}` ger, om det placeras i en `displaymath`,

$$\frac{L_{n-1}}{3}$$

`\frac` passar ofta mindre bra i löptext, då allt kan se högst ihoptryckt ut: $\frac{L_{n-1}}{3}$. Då kan man ofta få finare text genom att använda ett helt vanligt snedstreck: $L_{n-1}/3$

- `\left` och `\right` ger storleksanpassade parenteser. Skriver man `(\frac{4}{3})` får man

$$\left(\frac{4}{3}\right)$$

medan `\left(\frac{4}{3} \right)` ger det betydligt snyggare

$$\left(\frac{4}{3}\right)$$

Istället för parenteser kan man använda annat t ex klammerparenteser eller hakparenteser. Observera att `\left` och `\right` alltid förekommer i par.

- `\infty` ger tecknet ∞ , oändligheten.
- Förkortningen i.e., dvs ”that is”, skrivs `i.e.\` med ett mellanslag direkt efter bakåtsnedstreck.
- `\sqrt` ger kvadratroten ur något. `\sqrt{x}` blir \sqrt{x} .
- `\ldots` ger tre punkter på textens baslinje och används för att markera att något utelämnats.
- Omgivningen `equation*` fungerar som `displaymath`.
- `align*` är en omgivning för ekvationer som går över många led och rader.

- `\sum` ger ett summatecken i matematisk text. Skriver man `\sum_{k=1}^n b_k` får man t ex $\sum_{k=1}^n b_k$ i löptext och

$$\sum_{k=1}^n b_k$$

i fri text.

- `\lim` används för gränsvärden. Skriver man t ex

$$\lim_{x \rightarrow \infty} \frac{1}{x} = 0$$

blir det $\lim_{x \rightarrow \infty} \frac{1}{x} = 0$ i löptext och

$$\lim_{x \rightarrow \infty} \frac{1}{x} = 0$$

i en `displaymath`.

- `\to` ger en högerpil.
- `thebibliography` är en omgivning som bygger upp en referenslista där varje referens anges med hjälp av kommandot `\bibitem`.
- `\textbf` gick vi slutligen igenom på sidan 2 i början av detta dokument.