

# Mal til en slags artikkel

Fredrik Meyer

25. januar 2016

## Sammendrag

Sammendraget er en kort beskrivelse av hva resten av dokumentet handler om. Her skriver du hva temaet er, og hva du har tenkt å bevise. Det er ikke en roman du skriver, så spoilers skal ikke unngås. I resten av denne artikkelen skal jeg prøve å si noen nyttige ting for dem som ikke har brukt L<sup>A</sup>T<sub>E</sub>X så veldig mye. Ordentlige introduksjoner kan en finne andre steder. Artikkelen skal samtidig være en slags mal på hvordan en kan sette opp en artikkel eller en prosjektoppgave. Samtidig skal kildekoden være nyttig slik at den late leseren kan bruke den som en mal på sine produksjoner.

## 1 Introduksjon

I introduksjonen skal du motivere resten av artikkelen. Du forteller gjerne litt om hvorfor problemet er interessant, litt om historien bak, og hva andre folk har klart å gjøre før.

Deretter beskriver du i mer detalj hva du skriver om i de resterende seksjonene. I denne artikkelen vil Avsnitt 2 fortelle kort om hvordan du skaffer L<sup>A</sup>T<sub>E</sub>X og kommer i gang. Avsnitt 3 forteller litt om matematiske symboler og L<sup>A</sup>T<sub>E</sub>X. I Avsnitt 4 snakker jeg litt om noen nyttige vaner og uheldige uvaner folk ofte har. I Avsnitt 5 nevner jeg et par nyttige pakker som det er verdt å bli kjent med. I Avsnitt 6 lister jeg opp noen nyttige lenker, og til slutt avslutter jeg i Avsnitt 7 med å fortelle om hvordan det er vanlig å avslutte en artikkel.

## 2 Hvordan komme i gang med L<sup>A</sup>T<sub>E</sub>X

Det første du må gjøre er å skaffe L<sup>A</sup>T<sub>E</sub>X. Er du Linux-bruker, kan du følge oppskriften på <http://goo.gl/kIYY1C> (en stackoverflow-tråd) for å installere på egen maskin. Kjører du Windows, kan du følge oppskriften som ligger ute på IFI sine nettsider: <http://goo.gl/XsF5kf>. Bruker du Mac, kan du laste ned herfra: <https://tug.org/mactex/>.

Du vil også trenge en editor. Selv bruker jeg **TexShop** på Mac, og **Emacs** på Linux. Her er en fin liste over editorer: <http://goo.gl/PQi2nK>. Er du på team **Vim**, finnes det også L<sup>A</sup>T<sub>E</sub>X-pakker til denne. Det viktigste er at du har en hurtigtast for å kompilere (som du vil komme til å gjøre ofte).

Det er også mulig å unngå å laste ned noe i det hele tatt, om du er glad i online-editorer. Jeg vet om tre, men har bare prøvd én av dem. **Overleaf** (<https://www.overleaf.com/>), er en gratis editor som ved første øyekast ser veldig fin. Den kompilerer automatisk i bakgrunnen, slik at du ser forandringene i

dokumentet samtidig som du skriver. En annen som ser veldig fin ut, og som passer bedre for litt større prosjekter, er **ShareLaTeX** (<https://no.sharelatex.com/>), men som ikke er helt gratis.

En jeg har brukt er **SageMath Cloud** (<https://cloud.sagemath.com>). Dette er skyversjonen av **SageMath**, som er et slags gratis-substitutt for treenigheten **MATLAB**, **Mathematica**, og **Maple**.

Deretter må du lære deg å skrive dokumenter i  $\text{\LaTeX}$ . Dette gjøres best ved å hoppe ut i det. Skriv obliger i  $\text{\LaTeX}$ . Slutt med **Word** og **LibreOffice** og hold deg til et ordentlig tekstsysteem. Det hjelper ikke å lese masse introduksjoner med mindre du ikke koder selv.

### 3 Litt om å matematikk

Kanskje skriver du om analyse, og da snakker man gjerne om en  $\epsilon > 0$ . Matematikkmodus startes og slutes med dollartegn: `\epsilon > 0`.

Noen ganger har du lengre uttrykk:

$$f(z) = \sum_{i=0}^{\infty} \left( \frac{\int_{\gamma} g(w) dw}{i!} \right) + \left\langle \mathbf{v}, \sum_{i=1}^{\infty} \mathbf{v}_i \right\rangle. \quad (1)$$

For at parenteser skal få riktig størrelse, skriv `\left(` og `\right)` i stedet for kun `(` og `)`.

Noen ganger ønsker du ikke nummerering. Den enkleste måten er da å bruke `\[...\]`. Det går også an å bruke `\nonumber`.

$$\text{Tr} \begin{bmatrix} x_1^1 & \dots & x_n^1 \\ \vdots & \ddots & \vdots \\ x_1^n & \dots & x_n^n \end{bmatrix} = \sum_{i=1}^n x_i^i.$$

Andre ganger ønsker du å skrive opp lemmaer, teoremer og bevis. Jeg bruker de selvdefinerte kommandoene `\begin{prop}...``\end{prop}`, og så videre. Se kildekode til dette dokumentet for flere detaljer.

**Proposisjon 3.1.** *Mor Åse er en stein.*

*Bevis.* En stein kan ikke fly, og det kan Mor Åse heller ikke. Derfor er Mor Åse en stein.  $\square$

Vi kan også skrive korollarer:

**Korollar 3.2.** *Det finnes ingen idioter.*

*Bevis.* Anta at det finnes idioter. Men én idiot er en idiot for mye, derfor kan ikke idioter finnes.  $\square$

Her er en definisjon:

**Definisjon 3.3.** La  $(A, d_A)$  og  $(B, d_B)$  være metriske rom. Vi sier at en følge av funksjoner  $\{f_i: A \rightarrow B\}_{i \in \mathbb{N}}$  konvergerer uniformt til  $f: A \rightarrow B$  om det for hver  $\epsilon > 0$  finnes et naturlig tall  $N > 0$  slik at  $d_B(f_n(x), f(x)) \leq \epsilon$  for  $n \geq N$  og for alle  $x \in A$ .

Det finnes også teoremer. Legg merke til at både proposisjoner, korollarer, lemmaer og teoremer følger samme nummerering. Dette er en smakssak og er mulig å endre.

**Teorem 3.4.** *La  $\{f_i\}_{i \in \mathbb{N}}$  være en uniformt konvergerende følge av uniformt kontinuerte funksjoner på intervallet  $[0, 1]$ . Da er grensefunksjonen*

$$f = \lim_{i \rightarrow \infty} f_i$$

*kontinuerlig.*

*Bevis.* Vi har at  $f$  per definisjon gitt ved  $f(x) = \lim_{n \rightarrow \infty} f_n(x)$ . La  $\epsilon > 0$  og la  $x \in [0, 1]$ . Siden følgen konvergerer uniformt, finnes det en  $n \geq N$  slik at  $|f(x) - f_n(x)| < \epsilon/3$  for alle  $x \in [0, 1]$  og alle  $n \geq N$ .

Siden hver funksjon i følgen er uniformt kontinuert, finnes det en  $\delta > 0$  slik at  $|f_n(x) - f_n(y)| < \epsilon/3$  for alle  $n$  og alle  $|x - y| < \delta$ . Dermed har vi at

$$\begin{aligned} |f(x) - f(y)| &\leq |f(x) - f_n(x)| + |f_n(x) - f(y)| \\ &\leq |f(x) - f_n(x)| + |f_n(x) - f_n(y)| + |f_n(y) - f(y)| \\ &\leq \frac{\epsilon}{3} + \frac{\epsilon}{3} + \frac{\epsilon}{3} \\ &= \epsilon. \end{aligned}$$

Dermed har vi bevist påstanden. Dette er det berømte  $\epsilon/3$ -beviset. □

Her har jeg brukt kommandoen `\begin{align}...\end{align}` for å få  $\leq$ -tegnene under hverandre. Litt eldre introduksjoner til L<sup>A</sup>T<sub>E</sub>X vil foreslå å bruke `eqnarray` for dette, men sistnevnte er av forskjellige grunner mye dårligere.

## 4 Noen vanlige vaner og uvaner

Her demonstrerer jeg samtidig kommandoen `\subsection`.

### 4.1 Lag egne kommandoer

Lag dine egne kommandoer for symboler du bruker ofte. Eksempelvis: det er slitsomt å skrive `\mathbb{R}` hver gang du ønsker å skrive  $\mathbb{R}$ . Med kommandoen `\newcommand` kan du gjøre dette slik: `\newcommand{\R}{\mathbb{R}}`. Den plasserer du på en linje øverst i dokumentet ditt<sup>1</sup>, før `\begin{document}`.

Etterhvert som man får ganske mange egendefinerte kommandoer, er det smart å ha dem i en egen fil. Denne filen kan så inkluderes ved å skrive `\usepackage{mystyle}` i preamblet til dokumentet. Navnet “mystyle” referer her til en fil med navn `mystyle.sty`.

Det er to måter å lage kommandoer på som du trenger å vite: `\newcommand` og `\renewcommand`. Sistnevnte brukes om du vil overskrive en annen kommando. For eksempel ønsker du å bruke  $\mathbb{P}$  for  $\mathbb{P}$ , og da skriver du

$$\renewcommand{\P}{\mathbb{P}}.$$


---

<sup>1</sup>Alt som er over `\begin{document}` kalles gjerne for “preamblet”. Nå demonstrerte jeg samtidig hvordan en lager fotnoter. Det gjøres med `\footnote{...}`.

Kommando	Beskrivelse	Eksempel
<code>\ref</code>	Nummer	3.4
<code>\eqref</code>	Nummer i parentes	(3.4)
<code>\pageref</code>	Sidetall	3
<code>\cref</code>	Type og nummer	Teorem 3.4
<code>\vref</code>	Type, plassering og nummer	Teorem 3.4 på forrige side

Tabell 1: Kryssreferansekommandoer.

En annen god grunn til å bruke egendefinerte kommandoer er at om du ved en senere anledning ønsker å endre symbolbruken, så kan dette gjøres enkelt i preamblet til dokumentet – i stedet for å gå gjennom koden og endre hvert tilfelle av symbolet.

## 4.2 Ikke bruk tvang

L<sup>A</sup>T<sub>E</sub>X er laget for at du skal slippe å tenke på annet enn skrivingen. Plasserer du for eksempel bilder og figurer i dokumentet, kan disse ende opp på plasser du ikke forventet. Det er mulig å tvinge L<sup>A</sup>T<sub>E</sub>X til å sette bildene hvor du selv vil ha dem, men det er stort sett alltid bedre heller å referere til bildene som “Figur x.xx” enn å bruke masse tid på tvinge dem akkurat der du tror de vil passe akkurat nå.

Mange lar seg også ofte irritere av de store margene i L<sup>A</sup>T<sub>E</sub>X. Ikke gjør det. De er satt slik av hensyn til leselighet.

## 4.3 Semantisk korrekte kommandoer

Ønsker du å tegne implikasjonspiler, bruk  $\implies$  (`\implies`) i stedet for  $\Rightarrow$  (`\Rightarrow`). Vi har også  $\impliedby$  (`\impliedby`), og  $\iff$  (`\iff`). For funksjoner  $f: X \rightarrow Y$  brukes `\to` (og om du er veldig pedantisk, skal kolonet skrives med `\colon`).

Om du ønsker å *utheve noe*, bruk `\emph{uthevet}`, og ikke `\textit{skråstilt}`. Om du ønsker å skrive ..., bruk `\dots`.

## 4.4 Aldri godta feilmeldinger

Ofte produseres det en PDF-fil som ser helt fin ut til tross for at du får opptil flere feilmeldinger under kompileringen. Det er fristende å ignorere disse feilmeldingene, men ikke gjør det. Får du faktiske feil som hindrer kompilering, er disse vanskeligere å finne. I tillegg vil dokumentet ditt ofte se bedre ut, da mange av feilmeldingene ofte er av kosmetisk art.

## 4.5 Referanser

Vi kan referere til teoremer, figurer, og seksjoner andre steder i dokumentet. Se for eksempel Proposisjon 3.1. Eller kanskje du vil være mer presis: se på Proposisjon 3.1 på side 2.

Se på Tabell 1 for de mulige kommandoene. Tabellen er kopiert fra Martin Helsøs L<sup>A</sup>T<sub>E</sub>X-mal, som det er en lenke til i Avsnitt 6 på neste side.

For å referere må du merke ligningen, teoremet (eller tabellen, eller figuren, ...) som skal refereres. Dette gjøres ved å sette en `\label{navn}` rett etter det som skal referes. Deretter kan du referere til dem med en av kommandoene i tabellen<sup>2</sup>

## 5 Noen eksempler på nyttige pakker

Kanskje ønsker du å vise kode eller algoritmer i dokumentet ditt. Dette kan gjøres med pakken `listings`. Se kildekoden til dette dokumentet for et eksempel.

Her er et eksempel på hvordan Python-kode vises.

---

```
def approximateFraction(a, n):
    '''
    Input: a real number a, a natural number n.
    Output: a pair [k, m], such that the fraction k/m is close
    to a and 1 <= m < n.
    '''
    L = [i*a - floor(i*a) for i in range(0, n + 1)]
    mindiff = [1, 0, 0]
    for i in range(len(L)):
        for j in range(i + 1, len(L)):
            if abs(L[i] - L[j]) < abs(mindiff[0]):
                mindiff = [(L[i] - L[j]), i, j]
    i, j = mindiff[1], mindiff[2]
    k = -int(round(L[i] - L[j] - (i - j)*a))
    m = i - j
    return [k, m]
```

---

Andre ganger ønsker du kanskje å tegne kommutative diagrammer. Her anbefales `TikZ` eller `xypic`. Førstnevnte er på (nesten) alle måter bedre, så lær den først. Her er et eksempel på sistnevnte:

$$\begin{array}{ccccccc}
 0 & \longrightarrow & A & \xrightarrow{\iota} & B & \xrightarrow{\pi} & C \longrightarrow 0 \\
 & & \downarrow \varphi & & \downarrow \gamma & & \downarrow \psi \\
 0 & \longrightarrow & D & \xrightarrow{\iota'} & E & \xrightarrow{\pi'} & F \longrightarrow 0
 \end{array} \tag{2}$$

Dette er altså første halvdel av slangelemmaet. Det er et lite prosjekt å tegne inn selve slangen. Det kan være en oppgave til de vågale.

## 6 Nyttige ressurser

Under er en liste over nyttige ressurser (merk at lenkene er klikkbare om du leser dette på en skjerm).

- [goo.gl/yxFaEy](http://goo.gl/yxFaEy). Dette er en PDF med liste over forskjellige symboler  $\text{\LaTeX}$  har å by på.
- [tobi.oetiker.ch/lshort/lshort.pdf](http://tobi.oetiker.ch/lshort/lshort.pdf) “*The Not So Short Guide to  $\text{\LaTeX}$* ”. Denne er veldig fin. Om du har en ettermiddag, anbefaler jeg at du leser hele.

---

<sup>2</sup>Dette avsnittet har en dobbel funksjon. Den demonstrerer også hvordan man lager tabeller.

- [en.wikibooks.org/wiki/LaTeX](http://en.wikibooks.org/wiki/LaTeX) Wikibooks er også en bra ressurs, med mange eksempler.
- [www.ctan.org/](http://www.ctan.org/) Lurer du på hvordan pakkene du bruker egentlig fungerer, slå opp på CTAN for glimrende dokumentasjon.
- [folk.uio.no/martibhe/LaTeX/](http://folk.uio.no/martibhe/LaTeX/) Martin Helsø har stor peil på  $\text{\LaTeX}$ , og han har en del nyttige ressurser på sin hjemmeside.
- [detexify.kirelabs.org/classify.html](http://detexify.kirelabs.org/classify.html) Detexify er glimrende. Tegn noe osm ligner på et mattesymbol, og du får forslag til hvilken  $\text{\LaTeX}$ -kode du skal bruke for å skrive dette symbolet.
- [tex.stackexchange.com](http://tex.stackexchange.com) Her kan du stille spørsmål om du har prøvd alle andre muligheter. Det er også interessant tidsbruk å lese svar på de høyest rangerte spørsmålene.

I tillegg: spør folk du kjenner om hjelp. Og du kjenner alltid Google.

## 7 Videre spørsmål

Det er ofte vanlig å avslutte en artikkel med en diskusjon av hvilke problemer det gjenstår å løse. Om artikkelen din var mer en introduksjon eller en oversiktsartikkel, er det vanlig å snakke litt om hva du *ikke* snakket om, og om hvor leseren kan finne mer informasjon.

For eksempel, i dette dokumentet, har jeg unngått å snakke om en del avanserte deler av  $\text{\LaTeX}$ , og jeg har unngått å liste mange symboler. Formålet har vært at dokumentet skal fungere både som en slags inngangsport, og at kildekoden kan brukes som en mal for leseren (bare fjern min tekst først).

God  $\text{\TeX}$ -ing!