Typesetting Papers with LaTeX

Ulrich Germann
University of Toronto

Goal of Today's Tutorial

To enable you to use LaTeX to produce term papers, reports, conference papers, and journal articles in professional quality using existing layouts.

Not Covered

- Designing and defining layouts
- "Drawing" pictures in LaTeX
- Concurrent web/paper publishing
- Most lot of the gory details

Covered

- What is LaTeX and Why Use It?
- Basic LaTeX Syntax
- Tables
- Typesetting Mathematical Formulas
- "Floats" (Figures and Tables)
- Defining Commands and Environments
- Cross-References
- Footnotes
- Citations and Lists of References
- Organizing Large Documents
- Importing Graphics

What is LaTeX?

- *Not* a word processor
  ⇒ can't be compared directly with Word
- Technically, a macro package for D. Knuth's *typesetting system* TeX
- Conceptually, a *document processing system*
  - separates content from presentation
  - good layout is difficult ⇒ leave it to experts
- Best seen as a programming language: A source is compiled into a paper or printable online document.

LaTeX Myths: It's Unix/Linux only

"If LaTeX doesn't run on your machine, your machine is either incredibly old, incredibly new, or unbelievably obscure."

(From one of the many LaTeX FAQs)

LaTeX Myths: It's not WYSIWYG

- The LaTeX 'philosophy' separates content and presentation on purpose. Don't fiddle with layout while you write.
- LaTeX produces postscript and PDF. How much more WYSIWYG do you want?
- How much *synchronous* WYSIWYG you get depends on your choice of editor. There is at least one WYSIWYG frontend for LaTeX called Lyx (`http://www.lyx.org`).

LaTeXMyths: It's Difficult

Hopefully, this tutorial will convince you that it is not.

Why Use LaTeX? — Technical Reasons

- Portable across platforms, i.e., truly device-independent
- Transparent (once you've figured out the gory details)
- Mature, stable, well-maintained
- Able to handle very large documents (e.g., *your thesis*) reliably and gracefully
- Slow versioning cycle (fewer version incompatibilities)
- Huge support community

Why Use LaTeX? — Workflow Reasons

- Explicitly designed for scientific publishing (formulas, citations, references)
- Easy integration of material from other sources (figures, data, code)
- Facilitates modular document design
- De-facto an industry standard in our field; many conferences/publishers use/prefer/require LaTeX
- Facilitates uniform typographic conventions
  - within one document
  - across multiple documents (conference proceedings, journals, multi-authored books)

Why Use LaTeX? — Other Reasons

- Money: FREE!
- Politics/Religion: Open source
- Looks better:

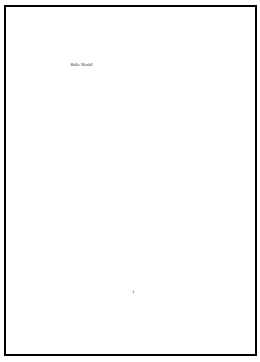
definitions vs. definitions

First Steps in LaTeX: 'Hello World'

```
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```

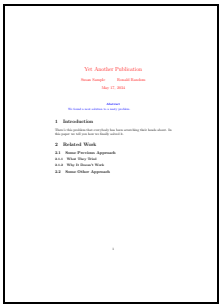
hello-world.tex

```
% latex hello-world
...
% dvips hello-world
...
% gv hello-world &
```



First Steps in LaTeX: Title and Sections

```
\documentclass{article}
\title{Yet Another Publication}
\author{Susan Sample \and Ronald Random}
\date{May 17, 2034} %default:\today
\begin{document}
\maketitle
\begin{abstract}
We found ...
\end{abstract}
\section{Introduction}
There's ...
\section{Related Work}
\subsection{Some Previous Approach}
\subsubsection{What They Tried}
...
```



First Steps in LaTeX: Title and Sections

```
\documentclass{article}
\title{Yet Another Publication}
\author{Susan Sample \and Ronald Random}
\date{May 17, 2034} %default:\today
\begin{document}
\maketitle
\begin{abstract}
We found ...
\end{abstract}
\section{Introduction}
There's ...
\section{Related Work}
\subsection{Some Previous Approach}
\subsubsection{What They Tried}
...
```

Yet Another Publicatio

Susan Sample Ronald Rand

May 17, 2034

Abstract

We found a neat solution to a nasty problem.

1 Introduction

There's this problem that everybody has been scratching this paper we tell you how we finally solved it.

2 Related Work

2.1 Some Previous Approach

2.1.1 What They Tried

2.1.2 Why It Doesn't Work

2.2 Some Other Approach

LaTeX Syntax

- Multiple spaces are treated as one
  ⇒ `Hello␣␣␣World` same as `Hello␣World`
- Single EOL char's count as spaces
- `\\` and `\newline` force line break
- `\\*` forces line break but prohibits page break
- One or more empty lines mean 'new paragraph'

LaTeX Syntax

- **Quotes:** `` `single' `` → 'single'
  `` ``double'' `` → "double"
- **Dashes:** `foo-bar` → hyphen foo-bar
  `126\,--134` → "en-dash" 126–134
  `foo --- bar` → "em-dash" foo — bar

Special Characters

- `%` comment character: everything to the end of line and all leading spaces on the next line are ignored
- `\` commands and declarations start with \
- `{ }` scope delimiters:
  - scope of declarations
  - obligatory arg's of commands
- `~` non-breakable space: LaTeX won't break the line here
- `$` used to enter/exit text style math mode
- `_`, `^` produce $_{\text{subscript}}$ / $^{\text{superscript}}$ in math mode
- `&` 'tab' character in tabular and array environments
- `#` used to distinguish arguments in the definition of commands (macros) and environments

Declarations, Commands, Environments

Declarations

`\`⟨*name = string of letters*⟩

Commands

`\`⟨*special char or name*⟩

`\`⟨*special char or name*⟩ {*mandatory arg1*}{*mand. arg2*}...

`\`⟨*name*⟩ [*optional arg1*]{*mand. arg2*}...

Environments

```
\begin{name}[opt. arg]{mand. arg}{mand. arg}...
...
\end{name}
```

Declarations

- change the way things look until the end of scope, e.g.

```
{one {two \bfseries three
{\itshape four} \Huge five} six}
```

⇒ one two **three** *four* **five** six

- `\large`, `\Large`, `\huge`, `\Huge`, `\small`, `\footnotesize`, `\tiny`, `\bfseries`, `\itshape`, `\sffamily`, `\rmfamily`, `\centering`, `\raggedright`, `\sloppy`, ...

Commands

- may have optional (`[...]`) and mandatory (`{...}`) arguments
- numerous functions
  - insert stuff (`\LaTeX` → LaTeX)
  - declare new commands, environments, counters, lengths etc. (explained later)
  - set stuff in a certain way, e.g.

```
A\parbox{3mm}{B\\B}C
A\parbox[t]{3mm}{B\\B}C
```

A B C B

A B C B

Environments

```
\begin{quote}
...
\end{quote}
```

LaTeX runs practically everywhere. As one FAQ puts it,

> *"If LaTeX doesn't run on your machine, your machine is either incredibly old, incredibly new, or unbelievably obscure."*
>
> (From one of the many LaTeX FAQs)

Environments

```
\begin{equation}
...
\end{equation}
```

Entropy is defined as

$$H(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i). \quad (1)$$

The `tabular` and `array` Environments

- `tabular` is for text; `array` for math

```
\begin{tabular}[b]{p{1.5cm}|r@{.}lc}...
```

- **optional argument:** `b`, `t`, or `c`
  - `t` – align top row of table with line
  - `b` – bottom row of table with line
  - `c` – center (default)
- **mandatory argument:** column formatter
  - `l` / `r` / `c` – left/right-aligned/centered column
  - `p{`*width*`}` – fixed-width column of width *width*
  - `|` – vertical line *between* columns
  - `@{...}` – other stuff between col's (on every row!)
  - `*{3}{c|}` – = `c|c|c|`

The `tabular` and `array` Environments

in the table/array body:

- `&` – separates colums; ends formatting scope!
- `\multicolumn{`*num*`}{`*col*`}{`*stuff*`}` – cell spanning *num* columns, containing *stuff*, formatted according to *col*
- `\\` – marks end of row

**add'l specifiers *after* \ **

- `[`*length*`]` – extra vertical space *length*
- `\hline` – creates a horizontal line, two a double line
- `\vline` – creates a vertical line within a cell
- `\cline{`col_1–col_2`}` – creates horizontal line between cols col_1 and col_2

The `tabular` and `array` Environments

```
\begin{tabular}[b]{p{1.5cm}|r@{.}lc}
foo & 2321 & 23 \\ \cline{1-1}
bar & 3 & 23423 \\ [3ex] \hline
foobar & 2324 & 46423
\end{tabular}
```

foo	2321.23
bar	3.23423
foobar	2324.46423

The `tabular` and `array` Environments

- **line breaks within cells:**
  - Possible only in `p{}` columns
  - Forced linebreak can be tricky if respective column is the last. `\\` is ambiguous; use `\newline` instead.
  - `parbox` / `minipage` allowed only in `p{}` columns
  - nested `tabular` environments allowed everywhere; can be used as a hack to force linebreaks

```
... & \begin{tabular}{@{}l@{}}
      foo\\bar
      \end{tabular} & ...
```


The `tabular` and `array` Environments

line spacing between rows:

- controlled by `arraystretch` command, adjust with `\renewcommand{\arraystretch}{⟨number⟩}`
- valid within scope, e.g.

```
{\renewcommand{\arraystretch}{2}
\begin{tabular}{ll}
...
\end{tabular}}
```


The `tabular` and `array` Environments

more parameters:

`\tabcolsep`	–	half the space between two tabular columns; margin at the edges of the ta
`\arraycolsep`	–	same thing for arrays
`\arrayrulewidth`	–	thickness of lines created by `\hline` e
`\doublerulesep`	–	space between two adjacent lines

- can be adjusted with `\setlength{parameter}{value}`, e.g. `\setlength{\tabcolsep}{2pt}`
- such changes are valid within the current scope; use `{...}` to delimit scope of changes

The `tabular*` Environment

`\begin{tabular*}{width}[pos]{cols}`
produces a table of fixed total width *width*

```
\begin{tabular*}{\linewidth}%
{c@{\extracolsep{\fill}}lr@{\extracolsep{0pt}::}l}
\hline\hline
bla & bla & bla & bla\\
blabla & blabla & blabla & blabla\\\hline
\end{tabular*}
```

bla	bla	bla::bla
blabla	blabla	blabla::blabla

Typesetting Math: Inline/Text Style

- $...$
- \(...\)
- \begin{math}...\end{math}

```
Entropy is defined as
$H(x) = -\sum_{i=1}^n p(x_i) \log_2 p(x_i)$.
```

Entropy is defined as $H(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i)$.

Typesetting Math: Display Style

- `\begin{displaymath}...\end{displaymath}`
- `\[...\]`

```
Entropy is defined as
\[H(x)=-\sum_{i=1}^n p(x_i) \log_2 p(x_i).\]
```

Entropy is defined as

$$H(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i).$$

Math Mode: `\textstyle`

```
Entropy is defined as
\[\textstyle
H(x)=-\sum_{i=1}^n p(x_i) \log_2 p(x_i).
\]
```

Entropy is defined as

$$H(x) = -\textstyle\sum_{i=1}^{n} p(x_i) \log_2 p(x_i).$$

The reciprocal command for use between $ ... $ is `\displaystyle`.

Math Mode: `equation` environment

```
Entropy is defined as
\begin{equation}
H(x)=-\sum_{i=1}^n p(x_i) \log_2 p(x_i).
\label{eq-entropy}
\end{equation}
```

Entropy is defined as

$$H(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i). \quad (2)$$

We can refer to Equation 2 in the text as `Equation~\ref{eq-entropy}`.

Math Mode: Caveats

- no spaces between 'words'; all letters set in italics
- don't use `$...$` as poor man's italics

`$function$` → $function$

`\textit{function}` → *function*

- you can insert plain text in math mode via `\textrm`, `\textsf`, or `\mbox`.

```
$p(n) = \frac{1}{m} for n \leq m$
```

$$p(n) = \frac{1}{m} for n \leq m$$

```
$p(n) = \frac{1}{m} \mbox{ for } n \leq m$
```

$$p(n) = \frac{1}{m} \text{ for } n \leq m$$

Bold Math

- In order to entire formulas in **bold**, declare `\boldmath` *outside* the respective math environment

```
{\boldmath$\sqrt[4]{x^\alpha}$}
```

$\boldsymbol{\sqrt[4]{x^\alpha}}$

```
$\sqrt[4]{x^\alpha}$
```

$\sqrt[4]{x^\alpha}$

- individual words can be set in bold, italics (default anyway), plain text with `\mathbf`, `\mathit`, `\mathrm` / `\mathsf`

Math Mode — Braces, Brackets, etc.

```
\[
f(x) = \left\{%
\begin{array}{ll}
1 & \mbox{for } x\geq 3 \\
0 & \mbox{otherwise}
\end{array}\right.
\]
```

`\right.` matches any single open `\left(`, `\left[`, etc.

$$f(x) = \begin{cases} 1 & \text{for } x \geq 3 \\ 0 & \text{otherwise} \end{cases}$$

Math Mode — eqnarray(*)

```
\begin{eqnarray}
3x^2 + 7 x & = & 0 \nonumber\\
3x + 7 &=& 0 \\
x &=& \frac{7}{3} \label{james}
\end{eqnarray}
```

$$3x^2 + 7x = 0$$
$$3x + 7 = 0 \quad (3)$$
$$x = \frac{7}{3} \quad (4)$$

Math Mode — eqnarray(*)

```
\begin{eqnarray}
3x^2 + 7 x & = & 0 \nonumber\\
3x + 7 &=& 0 \\
x &=& \frac{7}{3} \label{james}
\end{eqnarray}
```

- `\nonumber` suppresses equation number
- We can refer to the last row via `\ref{james}`
- `\eqnarray*` environment never prints equation numbers; `\label{...}` makes no sense in that case.

Figures and Tables

```
\begin{figure}[t]
put your figure stuff here
\caption[text for list of figures]{caption text here\label{...}}
\end{figure}
```

```
\begin{table}[t]
\caption[text for list of tables]{caption text here \label{...}}
put your table stuff here
\end{table}
```


Figures and Tables

- optional location parameter specifies where to put the figure/table:
  `\begin{figure}[loc]`
  - `h` – right here (does not work for `figure*` / `table*` environments in two-column mode!)
  - `!` – really right here; try harder!
  - `t` – top of page
  - `b` – bottom of page
  - `p` – separate page with figures and tables only
- default is `[tbp]`
- create lists of tables/figures with `\listoffigures` / `\listoftables` where you want them

Figures and Tables

- The `figure*` and `table*` environments create full-width figures/tables in articles with two-column format.
- `\renewcommand{\topfraction}{val}`, $0 \leq val \leq 1$ in the *document preamble* sets the fraction of the page that can be used for figures and tables; `\bottomfraction`, `\textfraction` work the same way

Defining Commands

`\newcommand{name}[args][default]{definition}`

- defines new command

`\renewcommand{name}[args][default]{definition}`

- re-defines existing command

`\providecommand{name}[args][default]{definition}`

- defines existing command unless it exists

Defining Commands

```
\newcommand{\seq}[3][x]{%
\ensuremath{#1_{#2},\dots,#1_{#3}}}
```

```
...  the sequence \seq[k]{1}{n} is...
```

...the sequence $k_1, \dots, k_n$ is ...

Defining Environments

\newenvironment{*name*}[*args*][*default*]{*begdef*}{*enddef*}

- defines new environment

\renewenvironment{*name*}[*args*][*default*]{*begdef*}{*enddef*}

- redefines existing environment

```
\newenvironment{itquote}%
{\begin{quote}\itshape}%
{\end{quote}
```


Theorem-like Environments

```
\newtheorem{lemma}{Lemma}[chapter]
⋮
\begin{lemma}
$\mathcal{K}$ is transitive.
\label{K-transitive}
\end{lemma}
```

Lemma 3.1 *$\mathcal{K}$ is transitive.*

Theorem-like Environments

\newtheorem{*name*}{*text*}[*within*]

\newtheorem{*name*}[*like*]{*text*}

- *name* – name of the environment
- *text* – text displayed before the counter
- *within* – counts within a certain section such as chapter (theses), section, etc.
- *like* – use the same counter as some other theorem-like environment

Theorem-like Environments

```
\begin{lemma}[\citealp{key}]
$\mathcal{K}$ is transitive.
\label{K-transitive}
\end{lemma}
```

Lemma 3.2 (Author, Year) *$\mathcal{K}$ is transitive.*

Cross-References

- refer to figures, tables etc. with \ref{*label*}
- works for all labeled and numbered thingies: figures, tables, footnotes, equations, sections (put label in the argument of the \section{...} command), etc.
- you can put a \label{...} everywhere and refer to the page with \pageref{...}

Footnotes

- insert `\footnote{...}` command where you want the footnote in the text

```
... when used sparingly.%
%
\footnote{Like this one.}
%
In \LaTeX, ...
```

While excessive use of footnotes should be avoided, they can be useful when used sparingly.[1] In LaTeX, you can just insert them into the source text wherever you want them to appear.

[1] Like this one.

Citations and Bibliographies: BibTeX

- `.bst` file specifies the *bibliography style*
  - how citations appear in the text ([1] , Author (2004) , [AUT04] , etc.)
  - what the entries in the list of references look like
- `.bst` file usually provided by someone else (e.g., publisher)
- `.bib` files contain the bibliographic data

.bib files

```
@String{PoPM = {Publish or Perish Magazine}}

@Book{sample00,
  author =  {Susan Sample},
  title =   {A General Solution to All Problems},
  publisher =   {Random University Press},
  year =   2000,
  address =  {Springfield, Idaho}
}
@Article{user97,
  author =   {Joe User},
  title =   {Yet Another Publication},
  journal =   PoPM,
  year =   1997,
  volume =  23,
  number =  4,
  pages =  {23--35}
}
```


.bib Files: Caveats

- Titles of journal articles, conference papers, etc. are often automatically lowercased, regardless of what's in the `.bib` file. Enclosing stuff in `{...}` protects against that:

```
title = {Reinventing {C}anada}
```

- Accented characters should also be enclosed in `{...}`

```
title = {G{\"{o}}del's Theorem}
```


Citations

- standard latex provides the command `\cite{`*key*`}`
- default citation style is bla bla bla [1] bla bla
- list of references is created like this:

```
\bibliography{DataFile1,DataFile2}
\bibliographystyle{StyleFile}
```

- *DataFile** are the names of the `.bib` files (without the extension)
- *StyleFile* is the name of the `.bst` file
- I highly recommend using the `natbib` package for citations

This Just In: Loading Packages

- packages provide additional functionality and can change the layout
- many conferences provide packages rather than classes
- packages are loaded with `\usepackage` in the preamble

```
\documentclass[options]{article}
\usepackage[options]{natbib}
\bibliographystyle{plainnat}
\begin{document}
...
\bibliography{biblio.bib}
```

- package options are separated by commas

The Natbib Package

- accomodates all major citations styles
  ([1] , Author (Year) , [CITEKEY])
- provides several citation commands, most importantly

`\citet{`*key*`}`	Author (Year)
`\citep[see also][:24]{`*key*`}`	(see also Author, Year:24)
`\citealt{`*key*`}`	Author Year
`\citealp{`*key*`}`	Author, Year

`\begin{lemma}[\citealp{key}]`

Lemma 3.3 (Author, Year) *$\mathcal{K}$ is transitive.*

The Natbib Package

- is well documented and offers many customization options; `\usepackage[`*options*`]{natbib}`

default	→	[Autor1, 1997, Author2, 2000]
`[round,colon]`	→	(Autor1, 1997; Author2, 2000)
`[numbers]`	→	[23,21]

Including Other LaTeX Files

`\input{`*file*`}` – Include *file.tex* at this point

`\include{`*file1*, *file2*, ...`}`

- Include contents of *file1.tex*, *file2.tex*, ... with page-breaks at the beginning and end, and between files
- `\include` commands cannot be nested

`\includeonly{`*file1*, *file2*, ...`}` in preamble!

- Only those specified with `\includeonly` are processed and produce output.
- Allows you to print a single chapter with correct cross-references (provided the excluded files have not changed since last processed)

Sample Thesis Master File

```
\documentclass{ut-thesis}
\usepackage[round,colon]{natbib}
...
\input{preamble} % all the preamble definitions like title, author, etc.
\includeonly{Intro,Conclusions} % print only Intro and Conclusions
\begin{document}
\include{Title-Stuff} % title page, dedication, table of content,
                      % lists of figures and tables, preface, etc.
\include{Intro}
\include{Theory}
\include{RelWork}
\include{Implementation}
\include{Evaluation}
\include{Conclusions}
\include{Appendices}
\include{Bibliography}
\end{document}
```


Sample Article Master File

```
\documentclass{article}
\usepackage[numbers]{natbib}
...
\title{...}
\author{...}
\begin{document}
\maketitle
\input{abstract}
\input{intro}
...
\input{conclusions}
\bibliographystyle{plainnat}
\bibliography{biblio}
\end{document}
```


Including `.eps` Graphics

```
\usepackage{graphicx}
⋮
\begin{document}
⋮
\includegraphics[opt1,opt2,...]{file}
```

Includes encapsulated postscript file *file.eps*.

\includegraphics Options

`scale=`*scale*	scale by factor *scale*
`width=`*w*	scale picture to have width *w*
`height=`*h*	scale picture to have height *h*
`angle=`*angle*	rotation angle
`origin=`*org*	origin for rotation, default is center (c)
`keepaspectratio`	If both `width` and `height` are specified, keep aspect ratio and scale to *at most* *w* and *at most* *h*

\includegraphics Options

`viewport=` *llx lly urx ury*	override bounding box (BB)
`trim =` *l b r t*	reduce BB by *l,b,r,t* on the left, bottom, right, top; *l,b,r,t* are length measures
`clip`	clip picture at bounding box/viewport; `\includegraphics*` has the same effect

List-making Environments

```
\begin{itemize}
\item ...
  \begin{itemize}
  \item ...
  \end{itemize}
\item ...
\end{itemize}
```

- Gnus
  - Green Gnus
- Gnats

List-making Environments

```
\begin{enumerate}
\item ...
  \begin{enumerate}
  \item ...
  \end{enumerate}
\item ...
\end{enumerate}
```

1. Gnus
   (a) Green Gnus
2. Gnats

List-making Environments

```
\begin{description}
\item[Gnus] ...
  \begin{description}
  \item[Green Gnus] ...
  \end{description}
\item[Gnats] ...
\end{description}
```

Gnus bla bla bla bla bla bla bla ...

Green Gnus bla bla bla ...

Gnats bla bla bla ...

Fine-tuning The Layout: Spaces

`\noindent`	insert at beginning of paragraph; blocks paragraph indentation
`\vspace{...}`	insert vertical space; no effect at top/bottom of page, use `\vspace*{...}` to force extra space at beginning/bottom of page
`\hspace{...}`	same thing for horizontal space
`~`	non-breakable space
`\␣`	force space (e.g., after commands)
`\,` , `\!` , `\:` , `\;`	thin / negative thin / medium / thick space

Fine-tuning The Layout: Breaks

`\newline`, `\\`, `\\*`	line break
`\newpage`	column break (in two-column mode); page break otherwise
`\clearpage`	page break plus print all floats declared so far
`\cleardoublepage`	like `\clearpage`; start new page on a new sheet of paper

Fine-tuning The Layout: Alignment

`\begin{`*alignment style*`}` . . . `\end{`*alignment stype*`}`

`center`	centering
`flushleft`	flush to the left (like a type writer)
`flushright`	flush to the right

Fine-tuning The Layout: Hyphenation

`conju\-gation`

- tells LaTeXthat where this current word can be hyphenated if needed

`\hyphenation{hy-phe-na-tion con-ju-ga-tion}`

- declares hypenation of words in general

Summary

- LaTeX syntax
- Tabular and Array Environments
- Typesetting Math
- Floats
- Defining Commands and Environments
- Theorems
- Cross-References
- Citations and Bibliographies
- Working with Multiple Files
- Integrating Postscript Figures
- List Environments
- Tweaking the Layout

Document Compilation Revisited

- run `latex` once → writes `.aux` file with citation keys and label locations
- run `bibtex` to produce `.bbl` file
- run `latex` twice more to insert citations and get cross-references right
- use `dvips` to produce postscript; `dvipdf` to produce PDF
- `pslatex` and `pdflatex` produce these files directly; doesn't always work

```
% latex thesis
% bibtex thesis
% latex thesis
% latex thesis
```


LaTeX Resources

- Leslie Lamport's LaTeXBook
- TeX User Group (`http://www.tug.org`)
  - TexLive CD
  - TeX Catalogue
- Class file for DCS theses available at `http://www.cs.toronto.edu/~fpitt/Latex`