LaTeX for Thesis and Large Documents

by

Stephen M. Carr and Wail Gueaieb

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the Ph.D. degree in
Electrical and Computer Engineering

School of Information Technology and Engineering
Faculty of Engineering
University of Ottawa

Abstract

These notes attempt to provide the beginner with the correct *approach* to using LaTeX for a thesis or large document, as well as with examples of how to format the main document structures. The notes themselves are written in LaTeX 2$_\varepsilon$ and the source documents can serve as a ready-made template for a thesis. However, please also consult the UW Thesis Regulations and Guide, available from the Graduate Studies Office, to make sure that your formatting complies with the current guidelines.

This is an example of a long abstract, formatted manually. It could be several paragraphs or even several pages long, since it may be difficult to summarize a thesis is a few words. Just keep track of how many pages it is and increment the counter correctly before beginning automatically formatted pages. Try to keep it short though!

Acknowledgements

Thanks to the previous authors of these notes, Colin Campbell and Ray White, who pioneered the on-line distribution of this course material. I've kept many examples, but have substantially re-organized, extended, and updated these notes for LaTeX 2_{ε}.

Contents

1 Introduction **1**
1.1 What is LaTeX? 1
1.2 Why Use LaTeX? 2
1.3 Possible Reasons Not To Use LaTeX 2
1.4 Where to Find LaTeX? 3

2 Creating Your Source File **4**
2.1 Description of Source File(s) 4
2.2 Organizing Your Source File 4
2.3 Editing Your Source File 5
2.4 Parts of the Master Source File 5
2.4.1 Preamble 5
2.4.2 The Logical Document 6

3 Example LaTeX Commands for Typical Large Documents **11**
3.1 Selected Text Structures 11
3.1.1 Basic In-Line Commands 11
3.1.2 Chapters, Sections and Subsections 12
3.1.3 Displayed Text Structures 13
3.2 Cross-References 15
3.3 Text Fonts 16
3.3.1 Font Selection 16
3.3.2 Font Family, Series, and Shape Selection 16
3.3.3 Text Size 17
3.4 Figures 17
3.5 Tables 18
3.6 Math Structures 20

3.6.1 In-Line Math 20
3.6.2 Displayed Formulae 20
3.6.3 Numbered Equations 21
3.6.4 Equation Arrays 21
3.6.5 Math Fonts 22
3.7 Appendices 22
3.8 Bibliography 22
3.8.1 “By-Hand” Method 22
3.8.2 Using BibTeX 23
3.9 Glossary 24
3.10 Index 24
3.10.1 “By-Hand” Method 25
3.10.2 Using MakeIndex 25

4 Processing Your LaTeX Source Files Under Unix 26

4.1 Running **latex** 26
4.2 Syntax Checking 26
4.3 Spell Checking 27
4.4 Previewing the Typeset Document 27
4.5 Printing 27
4.6 Creating Portable Document Format (PDF) 28

A Sources of Information and Help 29

B Glossary of Terms 30

List of Tables

3.1 Minimum Beam Radius Using Four Reliability Indices 19

List of Figures

3.1 Cantilever Beam 18

Chapter 1

Introduction

1.1 What is LaTeX?

LaTeX is a program for formatting or "typesetting" documents. Actually, when we talk about LaTeX we are usually referring not only to the **latex** program, but also to a suite of other utility programs which work together to format large, complicated documents or books.

LaTeX is unlike common word processing programs found on micro computers. It is more like a programming language. "Source" files are prepared with a text editor which are then processed by the **latex** program to produce the final document. Tags in the source file tell the **latex** program how to format the document. Also, like a programming language, you can (and should) create your own formatting tags, which are included at the beginning of your source file.

LaTeX is a set of macros written by Leslie Lamport [3] in the TeX language developed by Donald Knuth [2]. You may encounter two versions of LaTeX. The "old" version is 2.09. In 1993, a newer version, LaTeX 2_ε, was developed to standardize the many extensions which have been developed to improve functionality. To distinguish between versions, in LaTeX 2_ε the first command in a source document was changed from `\documentstyle` to `\documentclass` , and the extension "packages" are handled differently (with the new `\usepackage` command). The standardized packages in LaTeX 2_ε are described in a handy companion volume [1] to Lamport's "LaTeX Book"[3]. These last two books are required reading for serious LaTeXing, although often an example document (such as this one) is equally valuable.

1.2 Why Use LaTeX?

Why use LaTeX rather than a word processing package?

- The source document is plain (ASCII) text, so is very portable. Any editor or word processor on any computer can be used to write the source document.
- LaTeX is available on most computing platforms (the source code is free).
- It is used by many (especially scientific) publishers to format journals and books.
- LaTeX takes care of most of the formatting of the document with standard environments based on good typesetting form, letting the author concentrate on the *content*.
- It is flexible enough to be modified to suit any special needs.
- LaTeX *easily* keeps track of the numbering and organization of structures in large documents, allowing easy modifications.
- Large documents can be developed in modular form (chapters or sections in separate files).
- LaTeX has a superior mathematics formating language with every conceivable mathematical symbol.

1.3 Possible Reasons Not To Use LaTeX

- The software is free — but you *need* the reference books.
- There is little on-line help — although the Web has some information.
- Error messages can be cryptic, and errors difficult to isolate.
- You need to constantly be **latex**-ing and viewing your source to see what you've done.
- Word processors, such as WordPerfect, now do a reasonable job of handling large documents (but they're still not as good as LaTeX).

1.4 Where to Find LaTeX?

LaTeX is available on all Unix hosts on campus. Since the source is freeware, implementations have been developed for just about all computing platforms. Some versions are commercial because they provide additional user interface features such as editors which process the document in the background, or graphical equation editors. One such commercial product, Scientific Workplace, is supported at UW. It provides a graphical front end to LaTeX in which the user is shielded from the language syntax. See `http://ist.uwaterloo.ca/ew/software/scientific/` for details.

LaTeX is not currently on the Waterloo Polaris network.

Also, the Comprehensive TeX Archive Network (CTAN) on the Internet, contains all source files and utilities for LaTeX if you wish to download a copy for your home computer.

Chapter 2

Creating Your Source File

2.1 Description of Source File(s)

LaTeX source files are just ASCII text files containing the text of your document and the LaTeX mark-up tags.

The mark-up tags are of two types, *commands* and *environments*.

Command A command is a single LaTeX tag which performs a specific action, such as inserting some blank space, or toggles a formatting mode on or off, such as a larger type face.

Environment An environment is a formatting mode which is delimited by `\begin` and `\end` statements. For example, the `description` environment formats the display of a description offset from the surrounding text. Most commands also have an equivalent environment form.

2.2 Organizing Your Source File

Some general notes about organization:

- Clearly denote and comment the sections of your source file.
- In the main body, put each sentence on a new line, and let the lines wrap. This makes it easier to edit and move sentences around.
- Plan your document ahead of time. Think about what nomenclature you will use. What *logical structures* are suggested by the nomenclature?

- Will you need a bibliography? Do you have a lot of references which you will use again sometime?
- Will you need an index? If so, plan concepts and words to index.

LaTeX can easily keep track of all of these things, but it's easier to create your document if your are thinking about index items, citations and cross-references as you go, rather than as an afterthought!

2.3 Editing Your Source File

Use your favourite text editor.

- Unix: **vi**, **emacs**, or the easy **pico**
- PC or Mac: use any editor or word processor (and save as text)

Break your document up into several source files, a "master" file, and "subdocuments" (*e.g.*, one for each chapter and appendix).

2.4 Parts of the Master Source File

2.4.1 Preamble

The "preamble" of the source file tells the **latex** program how to format the structures and commands it finds in your document.

The basic format of the document is provided by the first preamble command, the `\documentclass` command. Document classes built-in to LaTeX are **letter, article, report, book,** and **slides**.

Examples of the `\documentclass` Command

```
\documentclass[12pt,2col]{article}
\documentclass[10pt]{report}
```

Options to this command (and other LaTeX commands) are given in square brackets just before the curly braces. Options are used to adjust the default point size of text, or other characteristics defined by a class.

UW has added its own modifications to some classes, notably, the **thesis91e** class. However, this class uses a several “styles” dating from LaTeX2.09, which are not now available on many Unix hosts. Since all **thesis91e** does is prepare the front material for the thesis, which is easily done “by hand”, it is not necessary. This document was prepared with the standard **report** document class.

The formatting commands may be built-in, defined in the standard packages, or defined by the user in the preamble itself with `\newcommand`. The `\usepackage` command identifies standard LaTeX 2_ε packages as well as non-standard [1] packages (which you could write yourself). Don’t forget to provide non-standard packages along with the other LaTeX source documents if you give them to someone else!

2.4.2 The Logical Document

The logical document resides inside the `document` environment. The document consists, usually, of three parts:

1. Front material — title page, table of contents, list of figures, *etc.*
2. Main body of text — chapters, sections, subsections, *etc.*
3. End material — appendices, bibliography, index

The outline of the logical document should be clear in the master source file. Use the `\include` command, *e.g.*, `\include{chap1}`, to include source subdocument `chap1.tex`.

Example Master Source File

```
% LaTeX (version 2e) source for "LaTeX for Thesis and Large Documents"
% By Stephen Carr and Wail Gueaieb
% Updated 2005-04-07
%
% Parts to edit are marked by the following block:
%###################################
%## Edit this part
%###################################
```

[1] Note that all packages and options are plain (ASCII) text files written in LaTeX or TeX, which can be copied and modified by the user. If you create your own packages or classes, LaTeX will find them if you put them in the same directory as your document source files (or, under Unix, set **latex**’s path variables to find them).

```
%
% Do NOT edit anything else unless you know what you are doing.
%
%======================================================================
%   P R E A M B L E
%======================================================================
\documentclass[%
12pt,   % font size
oneside,  % final version of thesis has to be submitted single-sided
%twoside, % in case you want to print a draft on both sides of the page
]{report}
%----------------------------------------------------------------------
%###################################
%## Edit this part
%###################################
\newcommand{\thesisauthor}{Stephen M. Carr and Wail Gueaieb}
\newcommand{\thesistitlecoverpage}{%
  \LaTeX \\
  for Thesis \\
  and Large Documents
}
\newcommand{\thesistitleheadings}{\LaTeX for Thesis and Large Documents}
\newcommand{\degree}{Ph.D.} % possible values are:
                          % M.A. / M.A.Sc. / M.Sc. / MCS / Ph.D.
\newcommand{\nameofprogram}{Electrical and Computer Engineering}
\newcommand{\academicunit}{School of Information Technology and Engineering}
\newcommand{\faculty}{Faculty of Engineering}
\newcommand{\graduationyear}{2005}

\newcommand{\abstractfile}{abstract.tex}
\newcommand{\acknowledgementfile}{acknowledgement.tex}
%----------------------------------------------------------------------
% Reprocess only those files which have changed recently:
% \includeonly{intro,creating,commands}
%----------------------------------------------------------------------
% Create a listing in the log of all files needed to process this document
\listfiles
```

```
%----------------------------------------------------------------------
\makeindex % activate index-making
%----------------------------------------------------------------------
\input{private/thesis-preamble} % thesis-specific settings
%----------------------------------------------------------------------
% My own command and environment definitions:
\newcommand{\program}[1]{\textbf{#1}} % program names in bold text
\newcommand{\exten}[1]{\texttt{#1}} % file extensions in bold text (use caps)
\newcommand{\cmmd}[1]{\textbackslash\texttt{#1}} % command name in tt font
\newcommand{\enviro}[1]{\texttt{#1}} % environment name in tt font

\newcommand{\eg}{\textit{e.g.},} % some Latin abreviations in italic
\newcommand{\ie}{\textit{i.e.},}
\newcommand{\etc}{\textit{etc}.\@}

\newcommand{\mat}[1]{\ensuremath{\mathcal{#1}}}
% matrix names in uppercase caligraphic
\newcommand{\vect}[1]{\ensuremath{\mathit{#1}}}
% vector names in math italic
\newcommand{\rv}[1]{\ensuremath{\mathbf{#1}}}
% math bold for random variables
\newcommand{\degg}[1]{\mbox{\raisebox{3pt}{$\circ$}\hspace{-.5pt}#1}}
% command to produce a degree sign. Example: \degg[C] gives degrees Celcius

\newenvironment{definition}[1]{\begin{quote}\emph{#1}:}{\end{quote}}
  % Provides indented formal definition and emphasizes the word.
  % e.g. \begin{definition}{Reliability} ... \end{definition}

\newenvironment{where}[1]% Equation symbol lists
 {\begin{list}{}%
  {\renewcommand{\makelabel}[1]{\hfill\textnormal{##1 =}}%
   \settowidth{\labelwidth}{\textnormal{#1 =}}%
   \setlength{\leftmargin}{\labelwidth}%
   \addtolength{\leftmargin}{\labelsep}%
   \setlength{\itemsep}{-\parsep}}}%
 {\end{list}}
% Example:
```

```
% \begin{where}{where $E$}
%  \item[where $E$] least squares error term;
%  \item[$w$] weighting factor associated with each measured variable.
% \end{where}
%----------------------------------------------------------------------
% Standard LaTeX2e packages I am using (as seen in "The LaTeX Companion"):
\usepackage[dvips]{graphicx}
% ... if you want to include encapsulated postscript figures
\usepackage{makeidx} % ... if you want an index
\usepackage{amsmath} % ... if you need lots of math symbols
\usepackage[dvips=true,bookmarks=true]{hyperref}
% ... only needed for PDF generation
%----------------------------------------------------------------------
% Non-standard packages I am using (things I've written, borrowed, etc.):
%\usepackage{} % ... note that old .sty files can be included here

%======================================================================
%   L O G I C A L    D O C U M E N T
%======================================================================
\begin{document}
%----------------------------------------------------------------------
% FRONT MATERIAL
%----------------------------------------------------------------------
\include{private/thesis-frontpages}
% Title page, declaration, borrowers' page, abstract, acknowlegements,
% dedication, table of contents, list of tables, list of figures

%----------------------------------------------------------------------
% MAIN BODY
%----------------------------------------------------------------------
\input{private/thesis-headings} % Specify thesis headings
%----------------------------------------------------------------------
%###################################
%## Edit this part
%###################################
% Chapters
% Include your "sub" source files here (must have extension .tex)
```

```
\include{chapter1-intro} %"What is LaTeX, Why Use It, and Where to Find It?"
\include{chapter2-creating} %"Creating Your Source File"
\include{chapter3-commands} %"Example \LaTeX\ Commands for Large Documents"
\include{chapter4-processing} %"Processing the Source File"
%----------------------------------------------------------------------
% END MATERIAL
%----------------------------------------------------------------------
%###################################
%## Edit this part
%###################################
% Appendices
\appendix
% Designate with \appendix declaration which just changes numbering style
% from here on
%\include{useful_pkgs} %"Some Useful \LaTeX2e Packages"
\include{appendix1-help} %"Sources of Information and Help"

% Glossary
% You could use a \begin{description} ... \end{description} for this
\include{glossary}

% Bibliography
% If done using the BibTeX program, use
\bibliographystyle{plain} % sorted alphabetically, labeled with numbers
\bibliography{bibliography/keylatex} % names file keylatex.bib as my bibliography file
% OR, do it "by hand" inside a "thebibliography" enivironment

% Index
% Put a \makeindex command in the Preamble if you use MakeIndex program
% and put
\printindex % here
% OR, do it "by hand" inside \begin{theindex} ... \end{theindex}
%----------------------------------------------------------------------
\end{document}
%======================================================================
```

Chapter 3

Example LaTeX Commands for Typical Large Documents

For details of the many LaTeX commands, the reference books are indispensable. However, examples of some structures you will likely need are given here.

LaTeX formatting tags include both *commands* and *environments*. A command is like a switch; it turns a certain mode on until it is changed or ended by the extent of its "scope", delimited by curly braces (or by default, such as the end of a paragraph). Examples of commands are "`\emph{really} big`" to emphasize a word and "`\large`" to turn on large text until it is changed again. An environment has an explicitly defined scope, indicated by `\begin` and `\end` commands, *e.g.*,`\begin{equation} ... \end{equation}`. All commands also have an environment form. See the Glossary of Terms on page 30 for more key concepts.

3.1 Selected Text Structures

3.1.1 Basic In-Line Commands

- LaTeX ignores extra spaces in the input. Add space with \␣, *i.e.*, \ followed by a space.

- Separate paragraphs with a blank line.

- LaTeX ignores anything after a % character (comment symbol).

- *Emphasize* text with the `\emph` command (italicizes).

- LaTeX recognizes three types of dashes
  - Punctuation (`---` , or `\textmdash`): He jumped — but not high enough!
  - Range (`--` , or `\textndash`): Read chapters 1–12 for tomorrow’s class.
  - Interword (`-`): What a thought-provoking essay!
- Use pairs of opening and closing *single* quotes rather than double quotes, to get “this” rather than ”this”.
- Ellipsis … is produced with the `\ldots` command.
- Produce some of LaTeX’s special symbols with a backslash preceding them *e.g.*, produce $, &, %, #, _, { and } with `\$`, *etc.*.
- Produce the remaining special characters `\`, `~`, and `^`, with the `\verb` command or inside a `verbatim` environment, or use text mode commands: `\textbackslash`, `\textasciitilde`, `\textasciicircum`, respectively.
- If you end a sentence with an upper case letter, let LaTeX know with a `\@` before the closing punctuation, *e.g.*, `Jane works for ABC\@?`
- Prevent inappropriate line breaks with a `~` *e.g.*, `Ms.~Wong`
- Special accented characters may also be produced, *e.g.*, `\"{o}` creates ö. Many other special symbols can be produced in math mode.

3.1.2 Chapters, Sections and Subsections

Logical structures of a document are preceded by commands.

```
\chapter{A Chapter Heading}
\section{A Section Heading}
A sentence.
%
\subsection{A Subsection Heading}
A sentence.
\subsubsection{A Subsubsection Heading}
A sentence.
```

LaTeX takes care of numbering these structures.

Long Headings and the Table of Contents

It is often the case that long chapter, section or caption names should be formatted differently in the Table of Contents than in the body of your thesis. The way to achieve this is to use an option field in the chapter, section, caption tag to contain the form of the heading that should appear in the Table of Contents, *e.g.*,:

```
\chapter[Introduction]{Introduction: Motivation for My Research}
```

Here, the Table of Contents will use the shorter chapter title, and the longer one will appear as the actual chapter title. This is also a handy method to use if you want the actual chapter title to contain a line break command (\\) while that's not necessary in the Table of Contents because of the smaller font used.

Unnumbered sections

There is a way to suppress the automatic numbering of several kinds of document structures. This is achieved by using the "star" form of the LaTeX command. For example to suppress the numbering of a paticular section:

```
\section*{A Section With No Number}
```

The starred form also exists for other automatically numbered document structures such as equations.

3.1.3 Displayed Text Structures

There are many structures for displaying text. Here are some examples:

Lists

For un-numbered lists use the `itemize` environment. For example,

```
  \begin{itemize}
  \item Birds
     \begin{itemize}
     \item ducks
     \item sparrows
     \end{itemize}
  \item Mammals
```

```
   \begin{itemize}
   \item dogs
   \item whales
   \end{itemize}
\end{itemize}
```

produces

- Birds
  - ducks
  - sparrows
- Mammals
  - dogs
  - whales

For numbered lists use the `enumerate` environment.

For definitions or descriptions use the `description` environment (see Appendix B).

```
\begin{description}
\item[Dog] Humankind’s best friend.
\end{description}
```

Dog Humankind’s best friend.

Quotations

Use the `quote` environment for short quotes.

```
\begin{quote}
If, at first, you don’t succeed, try, try again. \emph{Anonymous}
\end{quote}
```

> If, at first, you don’t succeed, try, try again. *Anonymous*

Use the `quotation` environment for quotations containing more than one paragraph.

Free-Form Text

The `verse` environment can be used to display free-form typeset text.

```
\begin{verse}
The verse environment lets you\\
break the lines wherever you like. If you go on too long ...
\end{verse}
```

> The verse environment lets you
> break the lines wherever you like. If you go on too long, though, the line will
>   eventually get broken for you.

The `verbatim` environment allows literal text to be printed. It uses a monospaced font to preserve spaces in the input and allows all special characters to be printed. Lines are not broken by LaTeX.

```
\begin{verbatim}
@#$%@#     $%%^!@#!
\en {verbatim}
```

```
@#$%@#     $%%^!@#!
```

Short pieces of literal text may be inserted in a line with the `\verb` command. This command is delimited by any two characters rather than by curly braces, *e.g.*, “`\verb+some literal text \\%^+`”.

3.2 Cross-References

Cross-references are produced with the `\label` and `\ref` commands.

```
\section{Discussion}
\label{sec.discussion}
.
.
.
As discussed in Section~\ref{sec.discussion}, ...
```

Page references may be generated with the `\pageref` command.

```
As discussed in Section~\ref{sec.discussion}
(page~\pageref{sec.discussion}) ...
```

3.3 Text Fonts

LaTeX has commands which allow the user to select the font family as well as the shape, weight, and size of the characters. It is important to know that there are different commands for text mode and math mode. Math fonts are discussed in Section 3.6.5.

3.3.1 Font Selection

The default font used by LaTeX is called Computer Modern, designed by Donald Knuth. Only recently — just pre-LaTeX 2$_\varepsilon$ — has it been (easily) possible to select other styles of fonts in LaTeX documents. The default font for the whole document may be selected with the `\usepackage` command.

3.3.2 Font Family, Series, and Shape Selection

Once the default font family is chosen, LaTeX commands may be employed to change the family, series, and shape characteristics. It should be emphasized that these changes should usually be defined in structure commands (defined in the preamble), rather than used directly in the text. Then, if you change your mind about how you want some object or concept identified, you just have to change the definition in the preamble for it to take place everywhere.

The font *family* is one of roman, sans serif, or typewriter (monospaced). The *series* is defined by weight (boldness) and width and may be either bold or medium. The *shape* may be upright (the default), italic, slanted, or small capitals.

Text-mode commands for controlling these attributes are:

Command	**Description**
`\textrm`	roman family
`\textsf`	sans serif family
`\texttt`	`typewriter family`
`\textbf`	**bold series**
`\textmd`	medium series
`\textit`	*italic shape*
`\textsl`	*slanted shape*
`\textup`	upright shape (default)

3.3.3 Text Size

Text size is controlled relative to the default size selected in the `\documentclass` command.

Size commands include `\tiny`, `\small`, `\large`, `\Large`, `\LARGE`, `\huge`, `\Huge`, `\normalsize`, `\footnotesize`, and produce the following effects:
tiny small large Large LARGE huge Huge normalsize footnotesize

3.4 Figures

LaTeX has its own built-in drawing capabilities in the `picture` environment, which allows the user to draw pictures composed of text, lines, arrows, simple curves, and geometric shapes. However, it is probably easier to use a drawing package to create your figures, then include them as Encapsulated PostScript (EPS) files. Whenever possible, save your drawings as EPS files (a form of PS designed for embedded figures), rather than as PostScript (PS), which includes full page formatting information. If necessary, however, PS files can be converted to EPS by adding the “bounding box” co-ordinates to the top of the PS file.

LaTeX allows inclusion of PostScript figures with the `graphicx` package. EPS files are simply embedded in the output device-independent (DVI) file for processing by a PostScript post-processor such as (dvips). The `graphicx` package allows the figure to be sized and rotated as desired. To achieve legibility, however, it is best to *draw* the figure the size you want it.

Here is an example of an included EPS figure and its LaTeX source code. Figure 3.1 shows a cantilever beam of circular cross-section subjected to a point load ($\mathbf{p}$) and a uniformly distributed load ($\mathbf{q}$), both of which are uncertain. The length of the beam, ($\mathbf{l}$), and its physical properties, Young’s modulus ($\mathbf{e}$) and ultimate bending stress ($\mathbf{b}$), are also fixed but uncertain. The random inputs of the design problem are then $\mathbf{V} = [\mathbf{p},\ \mathbf{q},\ \mathbf{l},\ \mathbf{e},\ \mathbf{b}]^T$.

```
\begin{figure}[!htbp]
 \begin{center}
  \includegraphics[clip=true]{beam.eps}
 \end{center}
\caption{Cantilever Beam}
\label{fig.beam}
```

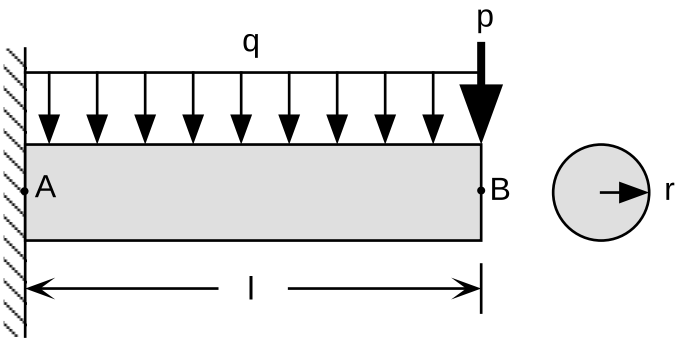

Figure 3.1: Cantilever Beam

```
\end{figure}
```

Note the `figure` environment, which provides for captioning, and `center` environment for positioning the figure.

Figures can also be pasted in by hand if you leave room for them. The following commands will leave a vertical space of 2.5 inches with a caption.

```
\begin{figure}
\vspace{2.5in}
\caption{Caption for Figure and for List of Figures - up to 300 chars}
\end{figure}
```

3.5 Tables

Tables are easy to generate in LaTeX. Simple tables can be produced with the `tabbing` environment, as seen in Section 3.3.2. The source for that table is as follows:

```
\begin{tabbing}
mmmmm\=mmmmmmmmmmmmm\= \kill
\>\textbf{Command} \>\textbf{Description} \\
\>\cmmd{textrm} \>roman family \\
\>\cmmd{textsf} \>sans serif family \\
\>\cmmd{texttt} \>typewriter family \\
\>\cmmd{textbf} \>bold series \\
\>\cmmd{textmd} \>medium series \\
\>\cmmd{textit} \>italic shape \\
\>\cmmd{textsl} \>slanted shape \\
\>\cmmd{textup} \>upright shape (default)
\end{tabbing}
```

Tabs are set with `\=`. The `\kill` command cancels the line after the tabs are set. The `\>` moves to the next set tab stop.

The `table` and `tabular` environments can be used to produce captioned and bordered tables. The following Table 3.1 provides an example.

	CCP Rel. Indices		**F**ORM Rel. Indices	
	$\beta_{MVFO}$	$\beta_{MVMS}$	$\beta_{FOSM}$	$\beta_{FO}$
$\{r^\star \,\vert\, \beta_1 \geq 3.719\}$	1.347†	1.348†	1.325	1.175
Rel. Error	+14.6%	+14.7%	+12.8%	—
$\{r^\star \,\vert\, \beta_2 \geq 3.092\}$	1.213	1.220	1.337†	1.290†
Rel. Error	-6.0%	-5.4%	+3.6%	—

†Active constraint when both are considered in the design optimization.

Table 3.1: Minimum Beam Radius Using Four Reliability Indices

```
\begin{table}[!htbp]
\begin{center}
\begin{tabular}{|c||cc|cc|}
\hline
& \multicolumn{2}{c|}{ {\textbf CCP Rel. Indices} }
& \multicolumn{2}{c|}{ {\textbf FORM Rel. Indices} } \\
\cline{2-5}
& $\beta_{MVFO}$ & $\beta_{MVMS}$ & $\beta_{FOSM}$ & $\beta_{FO}$ \\
\hline \hline
$\left\{ r^\star \, | \, \beta_1 \geq 3.719 \right\}$
& 1.347\dag & 1.348\dag & 1.325 & 1.175 \\
Rel. Error & +14.6\% & +14.7\% & +12.8\% & --- \\
%($\beta_2$) & (6.670) & (6.540) & (3.022) & (1.763) \\
\hline
$\left\{ r^\star \, | \, \beta_2 \geq 3.092 \right\}$
```

```
& 1.213 & 1.220 & 1.337\dag & 1.290\dag \\
Rel. Error & -6.0\% & -5.4\% & +3.6\% & --- \\
%($\beta_1$) & (3.222) & (3.251) & (3.755) & (4.742) \\
\hline
\end{tabular}
\begin{tabular}{l}
\footnotesize % small text inside this environment only
\dag Active constraint when both are considered in the design
optimization.
\end{tabular}
\end{center}
\caption{Minimum Beam Radius Using Four Reliability Indices}
\label{tab.results}
\end{table}
```

Note that a second `tabular` environment was used to produce a footnote under the table. A `minipage` environment around the table could also have been used.

3.6 Math Structures

LaTeX uses an intuitive language for formatting mathematics. Some examples of the main math environments will give the idea.

3.6.1 In-Line Math

In-line math may be inserted between `\(` and `\)`, or `$` and `$` (the latter form being preferred since it is not fragile). For example,

```
\rv{A} is a random matrix of size $m \times n$ ($m < n$).
```

produces "**A** is a random matrix of size $m \times n$ ($m < n$)".

3.6.2 Displayed Formulae

Math may also be displayed (set apart from the text) with the `displaymath`, or between the short-cut commands `\[` and `\]`. For example,

```
\begin{displaymath}
\beta \equiv \Phi^{-1}( \gamma )
\end{displaymath}
```

produces

$$\beta \equiv \Phi^{-1}(\gamma)$$

3.6.3 Numbered Equations

Numbered equations are produced inside the `equation` environment. For example,

```
\begin{equation}
\label{eqn.gradbetaFO}
\nabla_D \beta_{FO} = \nabla_D \beta_{FOSM} =
\frac{ \nabla_{D} g_{i} (U^{\star} }{ \| \nabla_{U} g_{i}
(U^{\star}) \| }
\end{equation}
```

produces

$$\nabla_D \beta_{FO} = \nabla_D \beta_{FOSM} = \frac{\nabla_D g_i(U^\star)}{\|\nabla_U g_i(U^\star)\|} \tag{3.1}$$

3.6.4 Equation Arrays

The `eqnarray` produces equation arrays, which are structures for multi-line formulae. Each line is given a number, unless this is suppressed with `\nonumber` command. Numbering may be suppressed for the whole environment if the `eqnarray*` form of the environment command is used. For example,

$$\mathit{optimize} \quad \mathbf{C}^T X \tag{3.2}$$

$$\mathit{subject\ to} \quad \mathbf{A}X \leq \mathbf{B} \tag{3.3}$$

is produced by

```
\begin{eqnarray}
  {\mathrm optimize} & \rv{C}^{T} X  \label{eqn.lpobjective} \\
  {\mathrm subject \: to} & \rv{A} X \leq  \rv{B}
  \label{eqn.lpconstraints}
\end{eqnarray}
```

Note that since each line is numbered, each may be given a label. The `eqnarray` environment allows three columns, aligned on the & symbols.

3.6.5 Math Fonts

Just as there are different type styles available in text mode, there are also math mode type styles, summarized below.

Command	**Description**
`\mathrm`	roman family
`\mathsf`	sans serif family
`\mathtt`	`typewriter` family
`\mathbf`	**bold** series
`\mathit`	*italic* shape
`\mathcal`	$\mathcal{CALIGRAPHIC}$ shape (upper case only)

These commands only change the style of letters, numbers, and uppercase Greek letters, not other math symbols.

3.7 Appendices

Appendices may be included by simply adding an `\appendix` command in your master document. All logical structures (chapters, sections, *etc.*) which appear below that point will then be "numbered" with letters.

3.8 Bibliography

A bibliography can be created two ways in LaTeX, "by hand" or by using the **bibtex** program. Once the bibliography items have been created by one of these methods, they are referred to in the LaTeX source files with the `\cite` command. For example,

```
\LaTeX\ is a set of macros by Leslie Lamport \cite{lamport.book}.
```

3.8.1 "By-Hand" Method

In the "End Material" section of your master source file include your bibliography list, which you must format yourself. For example,

```
\begin{thebibliography}{9}
\bibitem{jungle.book} Jungle, George
```

```
        {\em Watch Out for That Tree}
        Toronto: University of Toronto Press, 1986.
\bibitem{cliches.book} Smith, J.\ Henry.
        {\em Cliches and Platitudes for Every Occasion}
        Ottawa: Political Press, 1996
\end{thebibliography}
```

The second argument in the `\begin{thebibliography}{9}` is a number greater than the number of references to be formatted. The label parameter in the `\bibitem` command is used in the `\cite` command.

3.8.2 Using BibTeX

It is often easier to keep your bibliographic data together in bibliographic databases. The **bibtex** program works with LaTeX by gathering cited references from specified databases (text files with extension `bib`, called `BIB` files) and formatting the bibliography according to a desired style. This method is much easier than doing the formatting "by hand". If you create your own `BIB` files, put them in the same directory as your LaTeX source files, or set **latex**'s BIBINPUTS environment variable (under Unix) to the correct path.

The **bibtex** program is run after the **latex** program since it uses the `AUX` files produced by **latex**. The style and position of the bibliography in the final document is indicated by inserting the `\bibstyle` and `\bibliography` commands in the master LaTeX source file. The bibliography section by default does not appear in the Table of Contents. Adding it just requires manual editing of the file with extension `TOC`, produced by **latex**, followed by a final run of **latex**.

The bibliography is formated according to the specified `\bibstyle` for each type of reference given: books, journals, collections, *etc.*. The `bib` file used for this document is given below.

```
% Bibliography of key references for "LaTeX for Thesis
% and Large Documents"
% For use with BibTeX

@book{goossens.book,
        author = "Michel Goossens and Frank Mittelbach and
          Alexander Samarin",
        title = "The \LaTeX\ Companion",
        year =  "1994",
        publisher = "Addison-Wesley",
        address =  "Reading, Massachusetts"
}

@book{knuth.book,
        author =        "Donald Knuth",
        title =         "The \TeX book",
        year =          "1986",
        publisher =     "Addison-Wesley",
        address =       "Reading, Massachusetts"
}

@book{lamport.book,
        author =        "Leslie Lamport",
        title =         "\LaTeX\ --- A Document Preparation System",
        edition =       "Second",
        year =  "1994",
        publisher =  "Addison-Wesley",
        address =       "Reading, Massachusetts"
}
```

3.9 Glossary

There is no specific LaTeX command for creating a glossary. However, one can easily be produced with the `description` environment, as shown in Appendix B.

The **makeindex** program can also be used to create a Glossary automatically, if terms are defined in specific places in your document.

3.10 Index

An index can be created two ways in LaTeX, “by hand” or by using the **makeindex** program.

3.10.1 "By-Hand" Method

In the "End Material" section of your master source file include your index with the `theindex` environment, then format it yourself. For example,

```
\begin{theindex}
\item \LaTeX 3
   \subitem program, 4
      \subsubitem running, 5
\indexspace
\item \program{bibtex} 12
   \subitem running, 13
\end{theindex}
```

3.10.2 Using MakeIndex

Use of the **makeindex** progam is recommended as a significant improvement over producing an index by hand. If using **makeindex**, indexed items are referred to in the LaTeX source files with the `\index` command. For example,

```
\LaTeX\index{\LaTeX} is a set of macros ...
```

There are several other variations of the `\index` command which assist in formatting and ordering the index items.

To use **makeindex**, put a `\usepackage{makeidx}` command in the preamble of your master source file. If the command `\makeindex` also exists in the preamble of your master source file, LaTeX will produce an `IDX` file which is processed by **makeindex** to produce the final `IND` file. The `\printindex` command is placed in the master source document where you want the index to go in the end material. Running **latex** again will place the index in the output `DVI` file. Note also that the index does not by default appear in the Table of Contents. It must be added by hand, as was done for the Bibliography.

Chapter 4

Processing Your LaTeX Source Files Under Unix

4.1 Running latex

The **latex** program is executed on your master source file:

```
% latex master  (for a source file called master.tex)
```

If you are using **bibtex** or **makeindex**, run these next:

```
% bibtex master  (for source file master.tex)
% makeindex master
```

Run **latex** at least twice more to process all the labels and references. The resulting typeset document is in a file called `master.dvi`.

4.2 Syntax Checking

You will notice that LaTeX doesn't always give the most instructive error messages. It is useful to use a utility called **lacheck** to check for syntax errors before using the **latex** program.

```
% lacheck *.tex  (check all your .tex files)
```

4.3 Spell Checking

The Unix **correct** program is useful for checking spelling.

```
% correct -l master.tex  (-l option filters LaTeX commands)
```

4.4 Previewing the Typeset Document

The typeset document is in a `DVI` file, *e.g.*, `master.dvi`. This file may be previewed using either the **xdvi** or **ghostview** programs. It is a good idea to view your document frequently as you develop it. It's easier to debug your LaTeX source files as you work, rather than all at once.

To use **xdvi**, run it on the `DVI` file.

```
% xdvi master.dvi  (for source file master.tex)
```

To use **ghostview**, process the `DVI` file with **dvips** to produce a PostScript file then use **ghostview** to view it.

```
% dvips master.dvi > master.ps   (for source file master.tex)
% ghostview master.ps
```

4.5 Printing

Once you are satisfied with the final document you may print it using the **lpr** command.

```
% lpr -Plw -Fd master.dvi  (for source file master.tex)
OR ...
% lpr -Plw master.ps
```

You may also print directly from **ghostview**.

If you wish to print only some of the pages of your document, **ghostview** will let you do that. Alternatively, you can use the **dviselect** to print selected pages from a `DVI` file, for example

```
% dviselect -i master.dvi -o somepages.dvi 5-12 13 17
```

will place pages 5–12, 13, and 17 in a new `DVI` file called "somepages", which then must be sent to the printer as outlined above.

4.6 Creating Portable Document Format (PDF)

If you wish to create an electronic version of your document, rather than a paper version, a good choice is to create a PDF file, and LaTeX provides utilities for doing so.

PDF documents allow enriched features, such as hyperlinks. The **hyperref** package can be included in your master source file to automate internal hyperlinking in your document, as well as other features.

One important point to consider when creating a PDF file is the use of scalable fonts and graphics, to a avoid a “jaggy” appearance (loss of resolution) when you magnify a page.

This course demonstrates the inclusion of Encapsulated Postscript (EPS) files as figures. These are created by “printing to file” through a Postscript printer driver from a drawing or plotting application. EPS files are scalable graphics, so work well when creating PDF.

By default, LaTeX uses its own bit-mapped fonts (the Computer Modern set) when a printed document is produced. These bit-mapped fonts create blurry, “jaggy” results when converted to PDF. The way around this is to use the Postscript fonts made available under LaTeX 2$_\varepsilon$. Alternative Postscript fonts can be specified in the master document preamble by including font packages. There are also Postscript versions of the default Computer Modern font sets. To make sure the PS versions of the default Computer Modern fonts are used when creating a PS file via **dvips**, use the `-Ppdf` option (see below).

There are two methods to create a PDF file.

1. Create PDF directly from your `.tex` source filesby using the **pdflatex** program instead of **latex**.

2. Create a Postscript file then use **Adobe Acrobat** “distiller” or **Ghostview/GSView** to create the PDF file.

```
% dvips -Ppdf master.dvi > master.ps   (note use of -Ppdf option)
```

Appendix A

Sources of Information and Help

The best source of information about LaTeX is the two books mentioned in this course [3, 1]. Another excellent resource is the UseNet newsgroup `comp.text.tex`. A frequently-asked-questions (FAQ) list is also maintained by this news group. You might also search the World Wide Web for “LaTeX” for other sources of help.

Appendix B

Glossary of Terms

left-to-right (LR) mode Like paragraph mode, but no line breaks are inserted. The `\mbox` command is processed in LR mode.

math mode The mode for typesetting mathematics. Math mode is indicated in inline text between `\(` `\)` commands or pairs of $ symbols. The `equation` and similar environments also work in math mode. (Note that matched $ s within these environments toggles back to paragraph mode.)

mode One of three typesetting modes of LaTeX: paragraph mode, math mode, left-to-right (LR) mode.

paragraph mode The mode for processing ordinary text.

preamble The material in a LaTeX source file before the `\begin{document}` command. Used to set the `\documentclass`, packages (`usepackage`), and other formatting commands.

Bibliography

[1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1994.

[2] Donald Knuth. *The TeXbook*. Addison-Wesley, Reading, Massachusetts, 1986.

[3] Leslie Lamport. *LaTeX — A Document Preparation System*. Addison-Wesley, Reading, Massachusetts, second edition, 1994.

Index

LaTeX, 1

- availability, 3
- previewing, 27
- running, 26
- syntax checking, 26