A brief LaTeX tutorial.

Dr.Henri P. Gavin [*], Dr.George Easaw [‡]

Abstract

LaTeX is a very powerful text processing system available to all students through the university's network of Unix work stations. Versions of LaTeX which run on PC's and Mac's are available through the Internet. LaTeX is becoming the "standard" text processor for publications in science, engineering, and mathematics. LaTeX has many features, including:

- The printed output is *BEAUTIFUL*!
- Equations are relatively easy to enter, once you get the hang of it.
- LaTeX automatically numbers the sections, subsections, figures, tables, footnotes, equations, theorems, lemmas, and the references of your paper. When you add a new section or sub-section, or a new reference, every item is re-numbered automatically! It can create its own table of contents and indices automatically as well.
- It can easily manage very long documents, like theses and books.

The down-side of LaTeX is that it is somewhat cumbersome to use and can be difficult to learn. Many LaTeX manuals are not written for the rank beginner, and don't make it terribly clear how to make simple formatting changes such as margins, font size, or line spacing.

For many people, learning LaTeX through examples is the easiest way to go. The purpose of this document is to illustrate what LaTeX can do for you, and (without getting into too much detail) show you how to do it, so that you can learn to use LaTeX quickly and easily.

[*]Department of Civil and Environmental Engineering, Duke University, Durham, NC 27708–0287

[†]Department of Mechanical Engineering, College of Engineering, Farmagudi, Goa, 403 401

[‡]The original document by the first author has been reformatted by the second author for use in the Indian context. Value addition has been done by the Latex report template provided by the second author

Contents

1 How to use this tutorial **1**

2 The basic elements of a LaTeX file **1**

2.1 Font-size and other required formatting commands 1

2.2 Margins, line-spacing, and other optional formatting commands 2

2.3 Sections, sub-sections, paragraphs, and new-lines 2

2.4 Italics, bold-face, fontsizes, underlining, and centering 2

2.5 Examples of mathematical expressions 3

2.6 Lists 4

2.7 Tabbing 5

2.8 Arrays, Tables and Figures 5

2.8.1 Arrays 5

2.8.2 Tables 6

2.8.3 Figures 6

2.9 Appendix 7

2.10 Bibliographic references 7

3 Spelling correction **8**

4 Formatting and printing your .tex file **8**

5 Template for Reports **9**

6 Conclusion **11**

1 How to use this tutorial

The best way to use this tutorial is to read this (nice-looking) formatted version, and compare it to the LaTeX source file called `tutorial.tex` . By example, you will see how the title page is made, how PostScript figures can be included in your text, how to adjust the margins, how to double-space your text, how to organize your document into numbered sections, how equations are entered, how to make tables, and how to enter your list of bibliographic references.

This tutorial describes the new version of LaTeX: version *2ε*.

2 The basic elements of a LaTeX file

To use LaTeX, begin by creating an ASCII text file with a `.tex` suffix. You can use your favorite editor to do this (xedit, emacs, and vi are some examples). The file used to create this document is called `tutorial.tex` Blank lines in your `.tex` file indicate a new paragraph and a space character indicates a new word (*duh*). Otherwise all blank space is ignored. It is a good habit to start new sentences with a new line in your `.tex` file.

Comments in a `.tex` file start with a `%` character and go the the end of the line. The `%` character is therefore a *special character* in LaTeX. If you want to actually print a percent-sign, you must type `\%` . Three other special characters that you will use often are `\$` , `\` , and `&` (more about these later). The other six special characters are: `# _ ~ ^ { }` . To actually print any of these special character, simply precede it with a `\` character.

2.1 Font-size and other required formatting commands

The first command in your `.tex` file *must* be:

```
\documentclass[12pt]{article}
```

This declares that you will be writing an article, which is the right kind of document if you are not breaking up your work into chapters. Options are included between the square brackets. You can change the point-size of your font by specifying `12pt` for 12-point font, or `11pt` for 11-point font. If you don't specify `12pt` or `11pt` , LaTeX will default to a 10-point font. Other options are shown in the LaTeX source file `tutorial.tex` . The text of your document, **must** begin with the line:

```
\begin{document}
```

and end with the line:

```
\end{document}
```

These commands tell LaTeX to start and to stop looking for text to process, and **must** be included at the beginning and end of the document. The most simple `.tex` file has these three commands:

```
\documentclass{article}
\begin{document}
    This is the only text in my file.
\end{document}
```

and will use LaTeX's default font, margins, line spacing, page numbering and paragraph indentation for the pre-defined article document class.

2.2 Margins, line-spacing, and other optional formatting commands

The group of commands in your `.tex` file which come before `\begin{document}` is called the *preamble*. In the preamble, you can specify your margins and other options. Look in the preamble of `tutorial.tex` to see how to adjust the margins, line-spacing, paragraph spacing, paragraph indentation, equation indentation, and page numbering. If you leave these options out, then LaTeX will choose default values for you. Many LaTeX users do not like these defaults, and choose to modify the formatting style. Some aspects of the formatting style are easier to change than others. For example, changing the default font-size of your section headings is not trivial. With LaTeX you definitely do not have the same flexibility in formatting that you do with Microsoft Word. On the other hand, LaTeX has a standard format which is applied systematically and automatically to your entire document. You can spend your time working on the *content* of your paper, and leave the *style* to LaTeX!

2.3 Sections, sub-sections, paragraphs, and new-lines

A new section starts with a `\section{ "section title" }` command. A subsection starts with a `\subsection{ "sub-section title" }` command. You can also use the `\subsubsection{ "sub-sub-section title" }` command if you need to. Paragraphs are separated by a blank line. To force LaTeX to start a new line, put two back-slashes '\\' where you want the line-break to occur. To force LaTeX to start a new page, use the command: `\newpage` To insert vertical blank space in your document (5 millimeters in this example), use the command: `\vspace{5.mm}`

2.4 Italics, bold-face, fontsizes, underlining, and centering

Compared to Macintosh-based word processors, the fonts available in LaTeX can be quite limited. The default font is called Times-Roman `{\rm }`. Other type faces are: sans-serif `{\sf }`, *slanted* `{\sl }`, *italics* `{\it }`, **bold-face** `{\bf }`, SMALL CAPS `{\sc }`, and `typewriter {\tt }`. A broader range of fonts may be used by adding packages in the preable, for example `\usepackage{palatino}` , `\usepackage{utopia}` , and others are available. Text can be underlined as well using the `\underline{ }` command.

Fontsizes are changed by giving the commands `\normalsize, \large, \Large, \huge` etc. depending on the size one may need.

Formatted text is normally left- and right- justified.

Text between `\begin{center}` and `\end{center}` is centered horizontally.
New lines must be forced in centered text with the `\\` command.
To center a single line, use the command
`\centerline{ "centered text" }` .

In addition to specifying a font size in your `\documentclass` command, you can change font size within your text. These fonts are: tiny `{\tiny }`, scriptsize `{\scriptsize }`, footnotesize `{\footnotesize }`, small `{\small }`, normalsize `{\normalsize }`, large `{\large }`, Large `{\Large }`, and Huge `{\Huge }`.

2.5 Examples of mathematical expressions

Never start a paragraph with an equation!

One frequently needs to number equations and refer to them at different parts in the document.

Equation (1) says $\alpha = \beta\gamma\delta$.

$$\alpha = \beta\gamma\delta \tag{1}$$

Equations are automatically numbered by LaTeX. You can refer to an equation by its number if you label the equation in your `.tex` file.
Equation (1) has the label `\label{eq:abc}` in the `tutorial.tex` file, and is referred to with the command `\ref{eq:abc}` .
At any point in your document you can refer to this equation by typing `\ref{eq:abc}` in your `.tex` file. Labeling equations is optional.

The equation-formatting capabilities of LaTeX are highly touted! The following is an important equation in solid mechanics. It also shows how to do sub-scripts, super-scripts, and fractions.

$$I_{zz} = \int_{-b/2}^{b/2} \int_{-h/2}^{h/2} y^2 dy dx = \frac{bh^3}{12}. \tag{2}$$

These equations are important for statics:
$\sum F_x = 0$, $\sum F_y = 0$, and $\sum M_z = 0$.

and shows that in-line mathematical symbols can be inserted in your text by putting them between dollar signs, `\$` .
In general, all mathematical symbols are denoted by their Greek names, i.e., `\alpha` for α, `\Gamma` for Γ and `\epsilon` for ϵ. [1]

Other mathematical symbols are available, such as `\approx` for $\approx$, `\pm` for $\pm$, `\times` for $\times$, `\div` for $\div$, `\propto` for $\propto$, `\leq` for $\leq$, `\geq` for $\geq$, `\ll` for $\ll$, `\gg` for $\gg$, `\neq` for $\neq$, `\nabla` for ∇, `\Re` for $\Re$, `\Im` for $\Im$, `\flat` for $\flat$, `\sharp` for $\sharp$, `\partial` for ∂, `\infty` for ∞, `\sin` for sin, `\log` for log, `\arctan` for arctan, `\heartsuit` for $\heartsuit$, and many, many more.

Mathematical objects, like arrays, vectors, and matrices can be created as well.

See any text on LaTeX for more details regarding mathematical formulas.

2.6 Lists

Lists of items can be enumerated and itemized. An enumerated list begins with:

```
\begin{enumerate}
```

and ends with the command:

```
\end{enumerate} .
```

Individual items are denoted by the `\item` command. You can look at `tutorial.tex` to see how these commands work together.

1. This is the first item in an enumerated list.
2. This is the second. There is no limit to the length of items in a list. A single item can be several sentences long. You can have lists nested within other lists. You can put equations, tables, and figures in lists as well.
3. This is the third item.

That list was enumerated. The following list is itemized.

- Item number one in an itemized list.
- Itemized lists work just like enumerated list, except that `enumerate` is replaced by `itemize` in the `\begin` and `\end` commands.
- Item $\mathcal{C}$.

[1] Making Greek letters is a piece of π!

2.7 Tabbing

When preparing curriculum vitae and other carefully formatted documents, it is sometimes useful to define tabs. The command:
`\newcommand{\MyTabs}{ \hspace*{25.mm} \= ... \kill }`
in the preamble of `tutorial.tex` shows how to place tabs at every inch across the page starting at the left margin. The `\=` characters set the tab stops. The width of the text between the `\=` characters is the width of the tab. With `MyTabs` as defined in the preamble, you can start tabbing with the command:
`\begin{tabbing} \MyTabs`
Within the tabbing environment the `\>` characters advance to the next tab stop and lines must be terminated by the `\\` characters. Here is an example:

Column one Column two Column three etc.
Dates Job Title Employer Location

2.8 Arrays, Tables and Figures

Tables and figures are called *floating bodies* in LaTeX because their position on the page is seldom where you insert the table or figure commands in your `.tex` file. Rather, LaTeX allows tables and figures to float around on the page, and 'run a-ground' on the top of the page, the bottom of the page, or on a page by themselves. This is done to improve the readability of the text. On the printed page, tables and figures will never appear before they do in your `.tex` file. Commands in the preamble of `tutorial.tex` help LaTeX place tables and figures where you intend them to be.

2.8.1 Arrays

Arrays can be created like this.

`\begin{tabular}{c|l|r}` - col 1 is centered, 2 left and 3 right justified.

`\hline` - introduces a horizontal line

`field1 & field2 & field3` - each field is separated by an `&` sign. After all rows have been filled,

`\end{tabular}` - ends the entries.

If tables need to be filled, add `\begin{table}` and `\end{table}` before and after the begin and end tabular statements. `\label{ Label for the table}` adds a label for the table. See next section.

2.8.2 Tables

Tables are one of the more difficult items to create in LaTeX. Tables starts with the line:

```
\begin{table}[hbtp]
```

the options (in square brackets) tell LaTeX where to put the table on the page: **h**ere, **b**ottom, **t**op, or on a separate **p**age. A table can be centered on the page, and can be given a caption (with the `\caption{ }` command), which appears above the table. To format the table, the line:

```
\begin{tabular}{||c|l|r|r||} \hline \hline
```

tells LaTeX to make a table with two vertical lines on the left, then a centered column, then a vertical line, then a left justified column, another vertical line, then two right justified columns separated by a single vertical line, and two vertical lines on the right side. The table begins with two horizontal lines (`\hline`). Items in a table are separated by the ampersand character `&` . At the end of the (pre-defined) number of columns, a new line character '`\\`' is required. You may use one or more horizontal lines, (`\hline`) to separate the lines of text, at your choosing. The commands `\end{tabular}` and `\end{table}` close the table. Table 1 illustrates how a simple table appears.

Table 1: An example Table

Title one	Title Two	Title Three	Title Four
equations ok	α^3	G_1^{3+q}	$\int_0^T x^2$
text ok	hello,	world	
numbers ok too	123.45	567.89	123456

2.8.3 Figures

Figures are easier to include than tables. The easiest way to put figures in your document is to simply staple them to the end. For many technical publications, this is actually preferred. The next easiest method is to create a captioned blank space in your document and cut and paste your figure with scissors and glue. To do this you need to create space in the text using the LaTeX figure commands. For example, the lines:

```
\begin{figure}[hbtp]

        \centerline{\psfig{figure=name_of_figure.eps, height=xcm,width=ycm,angle=zzz}}
```

the .eps figure fits into a box x × y size and can be set as portrait or landscape depending on the angle.

```
        \caption{ my figure caption }

        \label{blank-fig}

        \end{figure}
```

will create a box x × y centimeters with a numbered figure caption below it. The figure will be given a label that you can use to refer to it in your text.

To automatically insert PostScript figures, you can use the `graphicx` package. First, you will need to put the line:

```
\usepackage{graphicx}
```

after the `\documentclass` line in your `.tex` file preamble. Your figures can be centered by following the example in `tutorial.tex` .

The .eps figures are usually created using the xfig package, saved in default .fig format (from file menu) and then exported to .eps format (from the file menu). xfig is a fine vector graphics package for all ordinary low end graphics applications.

You can have more than one `\includegraphics{ }` or `\tabular` command in a `\figure` or `\table` environment. The following figures are side by side and are not numbered.

If you leave out the `height=` or `width=` parameters in the `\includegraphics` command, then the aspect ratio of the original figure will be preserved.

2.9 Appendix

`\begin{appendix}` and `\end{appendix}` are used to enter the text which comes as appendix. This appears in the final document after compiling as Appendix A, Appendix B and so on. All sectioning, subsection and subsubsection commands work here too.

2.10 Bibliographic references

The easiest way to cite references in your document is with the author's name and year of publication in parenthesis (Lamport, 1994). This is actually the preferred method in many technical publications. You can make a numbered list of references with the `enumerate` commands. If you choose this method you should use the Harvard system for formatting your list of references:

1. Last1, First1 Middle1, and First2 Middle2 Last2 (year). *Book Title.* ed. Publisher, City.
2. Last1, First1 Middle1, and First2 Middle2 Last2 (year). Article Title. *Journal Name,* vol. X, no. Y, page–page.

Alternatively, you may use the automatic citation features of LaTeX [**?**, **?**]. It is convenient to put all your references together in a separate file.

The bibliography file begins with the line:

```
\begin{thebibliography}{99}
```

where the `99` has the same width as the longest number of your reference. Each item in the bibliography begins with a

`\bibitem{ label }`

where the `label` is used to cite to the reference in your text.

The text following the `\bibitem` line is the text of your reference. A suggested reference style is shown in this section. The line:

`\end{thebibliography}`

ends the bibliography. To actually include the references in your document, put the command:

`\input{biblio}`

where you want your references to appear (usually at the end of the report) provided you have the file named biblio.tex containing all the references.

References are automatically numbered (and re-numbered) by the LaTeX bibliography manager.

The list of references is not alphabetized by LaTeX. This is one thing you must do yourself.

3 Spelling correction

You can check the spelling in your ASCII `.tex` file by issuing the `ispell myfile.tex` command. The `ispell` program automatically ignores the special LaTeX formatting commands when checking files that end in `.tex` .

4 Formatting and printing your .tex file

To create a properly formatted PostScript file, first run the LaTeX program at the unix prompt with the command:

`latex myfile.tex`

If you have a bibliography, a table of contents, or other labeled items like equations, or figures, you will usually need to run `latex` more than once to get the cross-references right.

If you have an error in your `.tex` file, the LaTeX pre-processor will catch it, display an error message with the line number of the error, and give you a `?` prompt. If you type `e` at this prompt, you will enter an editor at this location in the text. If you type an `x` , you will exit the `latex` program.

You can then fix the error and re-process your `.tex` file. Running the `latex` program creates a file called `myfile.dvi` .

After you have run `latex` with no errors, you can create a pdf (portable document

format) file with the command:

`dvipdf myfile.dvi myfile.pdf`

This will create a .pdf file called `myfile.pdf` .

Before your print the .pdf file, you should check your document on the screen of an X-windows work-station with the command:

`ghostview myfile.pdf (or) kghostview myfile.pdf` (if you are using KDE desktop).

If everything is o.k. you can finally send the pdf file to a laser printer: `lpr -P`*printer_name* `myfile.pdf`

Or you can invoke the printer from within ghostview itself.

5 Template for Reports

Students frequently require LaTeX for compiling their project reports. A simple template which allows figures, equations etc to be inserted along with the text is given in this section.

Whenever a new chapter is written in a separate tex file, the first line of the new tex file should be

```
\chapter{title of chapter}
```

The rest of the chapter is written by including the

```
\section{title of section}, \subsection{title of subsection} and \subsubsection{
```

at the appropriate places in the document.

The report template for most students is given below.

```
\documentstyle[12pt,fullpage]{report} % font is 12 point
\renewcommand{\baselinestretch}{1.25} % this puts the spacing to 1.5
\textwidth 160mm
\textheight 230mm
\topmargin 5mm
\oddsidemargin -0.1in
\evensidemargin -0.1in

\input{epsf} % this lets you input .eps files from say xfig (useful in figures)
```

```
\input{psfig.sty}

% keep a file named title.tex ready with the right format
% then the following command is needed to input it into latex
\begin{document}  % starts the document formally
\pagenumbering{roman}
\input{title}   % you can give any other name to the file
\thispagestyle{empty}

\begin{abstract}
\input{abstr} % abstract file (abstr.tex}
\end{abstract}

\input{acknow} % file names acknow.tex to acknowledge others
\tableofcontents % create table of contents
\listoffigures   % creates list of figures
\listoftables  % creates list of tables
\newpage
\input{nomen} % nomenclature file
\newpage
\pagenumbering{arabic} % From here onwards numbering of the pages start
\input{intro}
\input{litt}
\input{chap3} % description of the work
\input{chap4} % experimental setup
\input{results}
\input{conclu}

\begin{appendix}
\input{appendix}
\end{appendix}
```

```
\begin{thebibliography}{99}
\input{biblio} % input the biblio.tex file
\end{thebibliography}

\end{document}
```

The biblio.tex or tutref.tex file will have many entries each line of which is thus. (Harvard style)

```
\bibitem{lamport00} Lamport Leslie (2000){\it Latex, the document preparation sy
```

lamport00 within braces is the keyword by which the book by Lamport is quoted anywhere in the report.

For example, The text written by Lamport `\cite{lamport00}` is the most authoritative one on Latex, would appear as

The text written by Lamport [1] is the most authoritative one on Latex.

This citation would put the proper reference number of the document by Lamport as given in the bibliography section, in the original report document.

6 Conclusion

As you can see, using LaTeX successfully is not completely trivial. There are, however, easy ways to get around the more difficult parts. For many science and engineering students and professionals the results are worth the extra effort.

This tutorial is far more detailed than necessary for the beginner. The quickest way to get started is to simply type in part of the preamble of `tutorial.tex` , and enter your own text. You can incorporate the examples found in this tutorial as you need, and as you become more experienced.

There are LaTeX manuals in the reference section of the Library. Reference [1] by Leslie Lamport is the standard reference on LaTeX.There are pages on the world wide web describing LaTeX as well. For tricky problems that need to be resolved quickly, you may post your question to the newsgroup `comp.text.tex` , or see `http://www-h.eng.cam.ac.uk/help/tpl/text`

References

[1] Lamport Leslie (2000) *Latex, the document preparation system*, Pearson Education Asia.