Essential LaTeX ++

Jon Warbrick
with additions by David Carlisle, Michel Goossens, Sebastian Rahtz, Adrian Clark

January 1994

Contents

List of Tables

1 Introduction

This document is an attempt to give you all the *essential* information that you will need in order to use the LATEX document preparation system[1].[1]. Only basic features are covered, and a vast amount of detail has been omitted. In a document of this size it is not possible to include everything that you might need to know, and if you intend to make extensive use of the program you should refer to a more complete reference. Attempting to produce complex documents using only the information found below will require much more work than it should, and will probably produce a less than satisfactory result.

The main reference for LATEX is *The LATEX User's guide and Reference Manual* by Leslie Lamport[1]. This contains all the information that you will ever need to know about the program, and you will need access to a copy if you are to use LATEX seriously.

Both the Manual and this document avoid mentioning anything that depends on the particular computer system that you are using. This is because LATEX is available on a number of systems and they all differ in one way or another. Instead, they both refer to a *local guide* for their particular system.

2 How Does LATEX Work?

In order to use LATEX, you generate a file containing both the text that you wish to typeset and instructions to tell LATEX how you want it to appear. You will normally create this file using your system's text editor. You can give the file any name you like, but it should end "`.tex`" to identify the file's contents. You then get LATEX to process the file, and it creates a new file of typesetting

[1] For those readers who have some familiarity with LATEX, this document assumes the LATEX 2_ε release of January 1994 or later

commands; this has the same name as your file but the "`.tex`" ending is replaced by "`.dvi`". This stands for '`Device Independent`' and, as the name implies, this file can be used to create output on a range of printing devices. Your *local guide* will go into more detail.

Rather than encourage you to dictate exactly how your document should be laid out, LaTeX instructions allow you to describe its *logical structure*. For example, you can think of a quotation embedded within your text as an element of this logical structure: you would normally expect a quotation to be displayed in a recognizable style to set it off from the rest of the text. A human typesetter would recognize the quotation and handle it accordingly, but since LaTeX is only a computer program it requires your help. There are therefore LaTeX commands that allow you to identify quotations and consequently allow LaTeX to typeset them correctly.

Fundamental to LaTeX is the idea of a *document class* that determines exactly how a document will be formatted. LaTeX provides standard document classes that describe how common logical structures (such as quotations) should be formatted. You may have to supplement these styles by specifying the formatting of logical structures peculiar to your document, such as mathematical formulae. You can also modify the standard document classes or even create an entirely new one, though you should know the basic principles of typographical design before creating a radically new style.

There are a number of good reasons for concentrating on the logical structure rather than on the appearance of a document. Doing so prevents you from making elementary typographical errors in the mistaken belief that they improve the æsthetics of a document—you should remember that the primary function of document design is to make documents easier to read, not prettier. It is also more flexible, since you need only alter the definition of the quotation style to change the appearance of *all* the quotations in a document. Most important of all, logical design encourages better writing. A visual system makes it easier to create visual effects rather than a coherent structure; logical design encourages you to concentrate on your writing and makes it harder to use formatting as a substitute for good writing.

3 A Sample LaTeX File

Have a look at the example LaTeX file in Figure 1. It is a slightly modified copy of the standard LaTeX example file `small.tex`. Your local guide will tell you how you can make your own copy of this file. The line numbers down the left-hand side are not part of the file, but have been added to make it easier to refer to various portions. Also have a look at Figure 2 which shows, more or less, the result of processing this file.

3.1 Running Text

Most documents consist almost entirely of running text—words formed into sentences, which are in turn formed into paragraphs—and the example file is no exception. Describing running text poses no problems, you just type it in naturally. In the output that it produces, LaTeX will fill lines and adjust the spacing between words to give tidy left and right margins. The spacing and distribution of the words in your input file will have no effect at all on the eventual output. Any number of spaces in your input file are treated as a single space by LaTeX. It also regards the end of each line as a space between words (see lines 15–17). A new paragraph is indicated by a blank line in your input file, so don't leave any blank lines unless you really wish to start a paragraph.

LaTeX reserves a number of the less common keyboard characters for its own use. The ten characters

```
# $ % & ~ _ ^ \ { }
```

should not appear as part of your text—if they do, LaTeX will get confused.

```
% SMALL.TEX -- Released 5 July 1985
% USE THIS FILE AS A MODEL FOR MAKING YOUR OWN LaTeX
% INPUT FILE. EVERYTHING TO THE RIGHT OF A  %  IS A
% REMARK TO  YOU AND IS IGNORED BY LaTeX.
%
% WARNING!  DO NOT TYPE ANY OF THE FOLLOWING 10 CHARACTERS
% EXCEPT AS DIRECTED:        &   $   #   %   _   {   }   ^   ~   \

\documentclass[11pt]{article}% YOUR INPUT FILE MUST CONTAIN
\begin{document}             % THESE TWO LINES PLUS
                             % THE \end COMMAND AT THE END

\section{Simple Text}    % THIS COMMAND MAKES A SECTION TITLE.

Words are separated by one or    more      spaces.  Paragraphs are
    separated by one or more blank lines.  The output is not affected
by adding extra spaces or extra blank lines to the input file.


Double quotes are typed like this: ``quoted text''.
Single quotes are typed like this: `single-quoted text'.

Long dashes are typed as three dash characters---like this.

Italic text is typed like this: \emph{this is italic text}.
Bold   text is typed like this: \textbf{this is  bold  text}.

\subsection{A Warning or Two}        % THIS MAKES A SUBSECTION TITLE.

If you get too much space after a mid-sentence period---abbreviations
like etc.\ are the common culprits)---then type a backslash followed by
a space after the period, as in this sentence.

Remember, don't type the 10 special characters (such as dollar sign and
backslash) except as directed!  The following seven are printed by
typing a backslash in front of them:  \$  \&  \#  \%  \_  \{  and  \}.
The manual tells how to make other symbols.

\end{document}                    % THE INPUT FILE ENDS LIKE THIS
```

Figure 1: A Sample LaTeX File

1 Simple Text

Words are separated by one or more spaces. Paragraphs are separated by one or more blank lines. The output is not affected by adding extra spaces or extra blank lines to the input file.

Double quotes are typed like this: "quoted text". Single quotes are typed like this: 'single-quoted text'.

Long dashes are typed as three dash characters—like this.

Italic text is typed like this: *this is italic text*. Bold text is typed like this: **this is bold text**.

1.1 A Warning or Two

If you get too much space after a mid-sentence period—abbreviations like etc. are the common culprits)—then type a backslash followed by a space after the period, as in this sentence.

Remember, don't type the 10 special characters (such as dollar sign and backslash) except as directed! The following seven are printed by typing a backslash in front of them: $ & # % _ { and }. The manual tells how to make other symbols.

Figure 2: The Result of Processing the Sample File

3.2 LaTeX Commands

There are a number of words in the file that start with '\' (see lines 9, 10 and 13). These are LaTeX *commands* and they describe the structure of your document. There are several things that you should realize about these commands:

- All LaTeX commands consist of a '\' followed by one or more characters.
- LaTeX commands should be typed using the correct mixture of upper- and lower-case letters. `\BEGIN` and `\Begin` are *not* the same as `\begin`.
- Some commands are placed within your text. These are used to switch things, like different typestyles, on and off. The `\small` command is used like this to emphasize text, normally by changing to an *italic* typestyle (see line 25). The command and the text are always enclosed between '`{`' and '`}`'—the '`{\em`' turns the effect on and and the '`}`' turns it off. All the typeface-changing switches like this have a corresponding *command* with an argument — all the possibilities are shown on page 9.
- There are other commands that look like

  ```
  \command{text}
  ```

  In this case, `text` is called the "argument" of the command. The `\section` command is like this (see line 13). Sometimes you have to use curly brackets '`{}`' to enclose the argument, sometimes square brackets '`[]`', and sometimes both. There is method behind this apparent madness, but for the time being you should be sure to copy the commands exactly as given.
- When a command's name is made up entirely of letters, you must make sure that the end of the command is marked by something that isn't a letter. This is usually either the opening bracket around the command's argument, or a space. When it's a space, that space is always ignored by LaTeX. We shall see later that this can sometimes be a problem.

3.3 Overall Structure

There are some LaTeX commands that must appear in every document. The actual text of the document always starts with a `\begin{document}` command and ends with an `\end{document}` command (see lines 10 and 39). Anything that comes after the `\end{document}` command is ignored. Everything that comes before the `\begin{document}` command is called the *preamble*. The preamble should contain only LaTeX commands to describe the document's style.

One command that must appear in the preamble is the `\documentclass` command (see line 9). This command specifies the overall style for the document. Our example file is a simple technical document and uses the `article` style, modified to print in eleven-point fonts. There are other styles that you can use, as you will discover in Section 4.

3.4 Other Things to Look At

LaTeX can print both opening and closing quote characters, and can manage either of these either single or double. To do this, it uses the two quote characters from your keyboard: **`** (which sometimes resembles a grave accent or back-quote) and **'** (apostrophe). You type these characters once for single quote (see line 21), and twice for double quotes (see line 20). The double quote character itself **"** is almost never used.

LaTeX can produce three different kinds of dashes. A long dash, for use as a punctuation symbol, as is typed as three dash characters in a row, like this '`---`' (see line 23). A shorter dash, used between numbers as in '10–20', is typed as two dash characters in a row, while a single dash character is used as a hyphen.

From time to time, you will need to include one or more of the LaTeX special symbols in your text. Seven of them can be printed by making them into commands by proceeding them by backslash (see line 36). The remaining three symbols can be produced by more advanced commands, as can symbols that do not appear on your keyboard such as †, ‡, §, \$, ©, ♯ and ♣.

It is sometimes useful to include comments in a LaTeX file, to remind you of what you have done or why you did it. Everything to the right of a `%`-sign is ignored by LaTeX, and so it can be used to introduce a comment.

4 Document Classes and Options

There are four standard document classes available in LaTeX:

`article` intended for short documents and articles for publication. Articles do not have chapters, and when `\maketitle` is used to generate a title (see Section 9), the title appears at the top of the first page rather than on a page of its own.

`report` intended for longer technical documents. It is similar to `article`, except that it contains chapters and the title appears on a page of its own.

`book` intended as a basis for book publication. Page layout is adjusted assuming that the output will eventually be used to print on both sides of the paper.

`letter` intended for producing personal letters. This style will allow you to produce all the elements of a well laid out letter: addresses, date, signature, etc..

These standard styles can be modified by a number of *options*. They appear in square brackets after the `\documentclass` command. Only one class can be used in a document but you can have more than one option, in which case their names should be separated by commas. The standard options are:

`11pt` prints the document using eleven-point type for the running text rather that the ten-point type normally used. Eleven-point type is about ten percent larger than ten-point.

`12pt` prints the document using twelve-point type for the running text rather than the ten-point type normally used. Twelve-point type is about twenty percent larger than ten-point.

`twoside` causes documents in the article or report styles to be formatted for printing on both sides of the paper. This is the default for the book style.

`twocolumn` produces two columns of text on each page.

`titlepage` causes the `\maketitle` command to generate a title on a separate page for documents in the `article` style. A separate page is always used in both the `report` and `book` styles.

There is further option which is very useful in European countries. The `a4paper` option causes the output in all of the standard styles to be adjusted to fit correctly on A4 paper. (LaTeX was designed in America where the standard paper is shorter and slightly wider than A4; without this option you will find that your output looks a little strange.)

5 Environments

We mentioned earlier the idea of identifying a quotation to LaTeX so that it could arrange to typeset it correctly. To do this you enclose the quotation between the commands `\begin{quotation}` and `\end{quotation}`. This is an example of a LaTeX construction called an *environment*. A number of special effects are obtained by putting text into particular environments.

5.1 Quotations

There are two environments for quotations: `quote` and `quotation`. `quote` is used either for a short quotation or for a sequence of short quotations separated by blank lines:

US presidents have been known for their pithy remarks:

> The buck stops here.
>
> I am not a crook.

```
US presidents have been known for
their pithy remarks:
\begin{quote}
The buck stops here.

I am not a crook.
\end{quote}
```

Use the `quotation` environment for quotations that consist of more than one paragraph. Paragraphs in the input are separated by blank lines as usual:

Here is some advice to remember:

> Environments for making quotations can be used for other things as well.
>
> Many problems can be solved by novel applications of existing environments.

```
Here is some advice to remember:
\begin{quotation}
Environments for making quotations
can be used for other things as well.

Many problems can be solved by
novel applications of existing
environments.
\end{quotation}
```

5.2 Centering and Flushing

Text can be centred on the page by putting it within the `center` environment, and it will appear flush against the left or right margins if it is placed within the `flushleft` or `flushright` environments. Notice the spelling of `center`—unfortunately LaTeX doesn't understand the British English spelling.

Text within these environments will be formatted in the normal way; in particular, the ends of the lines that you type are regarded as spaces. To indicate a new line, you type the \\ command. For example:

one two three
four
five

```
\begin{center}
one
two
three \\
four \\
five
\end{center}
```

5.3 Lists

There are three environments for constructing lists. In each one, each new item is begun with an \item command. In the itemize environment, the start of each item is given a marker, while, in the enumerate environment, each item is marked by a number. These environments can be nested within each other, in which case the amount of indentation used and the marker are adjusted accordingly:

- Itemized lists are handy.
- However, don't forget
    1. The 'item' command.
    2. The 'end' command.

```
\begin{itemize}
\item Itemized lists are handy.
\item However, don't forget
  \begin{enumerate}
  \item The 'item' command.
  \item The 'end' command.
  \end{enumerate}
\end{itemize}
```

The third list-making environment is description. In a description you specify the item labels inside square brackets after the \item command. For example:

Three animals that you should know about are:

gnat A small animal that causes no end of trouble.

gnu A large animal that causes no end of trouble.

armadillo A medium-sized animal.

```
Three animals that you should
know about are:
\begin{description}
  \item[gnat] A small animal that causes
    no end of trouble.
  \item[gnu] A large animal that causes
    no end of trouble.
  \item[armadillo] A medium-sized animal.
\end{description}
```

5.4 Verbatim Output

Sometimes you will want to include text exactly as it appears on a terminal screen. For example, you might want to include part of a computer program. Not only do you want LaTeX to stop playing around with the layout of your text, you also want to be able to type all the characters on your keyboard without confusing LaTeX. The `verbatim` environment has this effect:

The section of program in question is:

```
{ this finds %a & %b }

for i := 1 to 27 do
   begin
   table[i] := fn(i);
   process(i)
   end;
```

```
The section of program in
question is:
\begin{verbatim}
{ this finds %a & %b }

for i := 1 to 27 do
   begin
   table[i] := fn(i);
   process(i)
   end;

\end{verbatim}
```

6 Type Styles

We have already come across the `\emph` command for changing the typeface to emphasis. Here is a full list of the available typeface changing commands:

Command	*or*	*Effect*
`\textrm{...}`	`{\rmfamily...}`	Text is set in roman family
`\textsf{...}`	`{\sffamily...}`	Text is set in sans serif family
`\texttt{...}`	`{\ttfamily...}`	Text is set in `typewriter` family
`\textmd{...}`	`{\mdseries...}`	Text is set in medium series
`\textbf{...}`	`{\bfseries...}`	Text is set in **bold** series
`\textup{...}`	`{\upshape...}`	Text is set in upright shape
`\textit{...}`	`{\itshape...}`	Text is set in *italic* shape
`\textsl{...}`	`{\slshape...}`	Text is set in *slanted* shape
`\textsc{...}`	`{\scshape...}`	Text is set in SMALL CAPS shape
`\emph{...}`	`{\em...}`	Text is set *emphasized*
`\textnormal{..}`	`{\normalfont..}`	Text is set in the document font

Remember that the declaration versions (second column) are used *inside a pair of braces* to limit the amount of text that they affect. It is recommended that you use the commands (first column) with a parameter of the text to typeset differently. In addition to the typeface commands, there are a set of commands that alter the size of the type. These commands are:

tiny scriptsize footnotesize small normalsize large Large LARGE huge Huge

```
\tiny tiny
\scriptsize scriptsize
\footnotesize footnotesize
\small small
\normalsize normalsize
\large large
\Large Large
\LARGE LARGE
\huge huge
\Huge Huge
```

7 Sectioning Commands and Tables of Contents

Technical documents, like this one, are often divided into sections. Each section has a heading containing a title and a number for easy reference. LaTeX has a series of commands that will allow you to identify different sorts of sections. Once you have done this, LaTeX takes on the responsibility of laying out the title and of providing the numbers.

The commands that you can use are:

```
\chapter       \subsection       \paragraph
\section       \subsubsection    \subparagraph
```

The naming of these last two is rather unfortunate, since they do not really have anything to do with 'paragraphs' in the normal sense of the word: they are just lower levels of section. In most document styles, headings made with `\paragraph` and `\subparagraph` are not numbered. `\chapter` is not available in document style `article`. The commands should be used in the order given, since sections are numbered within chapters, subsections within sections, etc..

A seventh sectioning command, `\part`, is also available. Its use is always optional, and it is used to divide a large document into series of parts. It does not alter the numbering used for any of the other commands.

Including the command `\tableofcontents` in your document will cause a contents list to be included, containing information collected from the various sectioning commands. You will notice that each time your document is run through LaTeX the table of contents is always made up of the headings from the *previous* time you ran LaTeX on it. This is because LaTeX collects information for the table as it processes the document, and then includes it the next time it is run. This can sometimes mean that the document has to be processed through LaTeX *twice* to get a correct table of contents.

8 Producing Special Symbols

You can include in your LaTeX document a wide range of symbols that do not appear on your keyboard. For a start, you can add an accent to any letter:

ò	\`{o}	õ	\~{o}	ǒ	\v{o}
o̧	\c{o}	ó	\'{o}	ō	\={o}
ő	\H{o}	ọ	\d{o}	ô	\^{o}
ȯ	\.{o}	o͡o	\t{oo}	o̲	\b{o}
ö	\"{o}	ŏ	\u{o}		

Several other symbols are available and are generated using the following commands:

†	\dag	§	\S	©	\copyright
‡	\ddag	¶	\P	$	\pounds
œ	\oe	Œ	\OE	æ	\AE
Æ	\AE	å	\aa	Å	\AA
ø	\o	Ø	\O	ł	\l
Ł	\E	ß	\ss	¿	?`
¡	!`	...	\ldots	LaTeX	\LaTeX

There is also a `\today` command that prints the current date. When you use these commands remember that LaTeX will ignore any spaces that follow them, so that you can type '`\pounds 20`' to get '$20'. However, if you type '`\LaTeX is wonderful`' you will get 'LaTeXis wonderful'—notice the lack of space after LaTeX. To overcome this problem you can follow any of these commands by a pair of empty braces and then any spaces that you wish to include, and you will see that `\LaTeX{} really is wonderful!` (LATEX really is wonderful!).

Finally, LATEX 'math' mode, normally used to layout mathematical formulae, gives access to an even larger range of symbols, including the upper and lower case greek mathematical alphabet, calligraphic letters, mathematical operators and relations, arrows and a whole lot more. This will be discussed in Section 13.

9 Titles

Most documents have a title. To title a LATEX document, you include the following commands in your document, usually just after `\begin{document}`.

```
\title{required title}
\author{required author}
\date{required date}
\maketitle
```

If there are several authors, then their names should be separated by `\and` in the `\author` command; they can also be separated by `\\` if you want them to be centred on different lines. If the `\date` command is left out, then the current date will be printed.

```
\title{Essential \LaTeX}
\author{Jon Warbrick \and A N Other}
\date{14th February 1988}
\maketitle
```

Essential LATEX

J Warbrick A N Other

14th February 1988

The exact appearance of the title varies depending on the document style. In the `report` and `book` styles, the title appears on a page of its own. In the `article` style, it normally appears at the top of the first page; use the class option `titlepage` to alter this (see Section 4).

10 Tabular Material

Because LATEX will almost always convert a sequence of spaces into a single space, it can be rather difficult to lay out tables. See what happens in this example

Income Expenditure Result
20s 0d 19s 11d happiness
20s 0d 20s 1d misery

```
\begin{flushleft}
Income  Expenditure Result    \\
20s 0d  19s 11d     happiness \\
20s 0d  20s 1d      misery
\end{flushleft}
```

The `tabbing` environment overcomes this problem. Within it, you set tab-stops and tab to them much as you would do on a typewriter. Tab-stops are set with the `\=` command, and the `\>` command moves to the next stop. The `\\` command is used to separate each line. A line that ends in `\kill` produces no output, and can be used to set tab-stops:

Income	Expenditure	Result
20s 0d	19s 11d	Happiness
20s 0d	20s 1d	Misery

```
\begin{tabbing}
Income \=Expenditure \=       \kill
Income \>Expenditure \>Result \\
20s 0d \>19s 11d \>Happiness  \\
20s 0d \>20s 1d  \>Misery
\end{tabbing}
```

Unlike a typewriter's tab key, the \> command always moves to the next tab-stop in sequence, even if this means moving to the left. This can cause text to be overwritten if the gap between two tab-stops is too small.

If you have more sophisticated tabular work, you will need the `tabular` environment. This differs from `tabbing` in that you need not worry about widths of columns—LaTeX will look at the whole table and see how wide columns need to be to cope with the widest entries in them. You must, however, tell LaTeX how many columns there are and how you wish to lay them out. You do this by supplying a *template* after `\begin{tabular}` with a letter for each column:

`l`	means the column will be left justified
`r`	means the column will be right justified
`c`	means the column will be centred

Items within a row are separated by the & character (now you know why it is treated specially), and the row is ended, as before, with \\. A simple table with three columns and two rows looks like:

Name	*Age*	*Height*
Sebastian	45	195cm
Mathew	1	68cm

```
\begin{tabular}{lrr}
\em Name & \em Age &\em Height \\
Sebastian & 45 & 195cm         \\
Mathew    & 1 & 68cm
\end{tabular}
```

Various other bells and whistles can be used to make the layout nicer:

1. You can put `\hline` between rows to draw a line across the table.
2. Type a | in the template where you want a vertical bar down every row of the table.
3. You can specify that a column is of a fixed width with text flowing within it by using the symbol `p` in the template, followed by a width between { and }. The width can be given in any of 'cm', 'mm' or 'in' (but not 'inches'!).

Thus, a more sophisticated table might look like this:

Group	*Type*	Sherds
Groups 1–9	Grey wares	218
Groups 40–44	Black (mostly 'black burnished') wares	116
Groups 61–67	Buff-red-orange wares	46
Groups 81–85	Colour-coated fine wares	67
Groups 91–2, 93–4	Mortaria and miscellaneous	35
Group 96	Samian	56
		538

```
\begin{tabular}{|l|p{1in}|r|}  \hline
\em Group &\em Type &Sherds  \\\hline
Groups 1--9   &Grey wares&218\\
Groups 40--44 &Black (mostly
'black burnished')  wares&116\\
Groups 61--67 &Buff-red-orange
wares & 46                   \\
Groups 81--85 &Colour-coated
fine wares & 67              \\
Groups 91--2, 93--4&Mortaria
and miscellaneous & 35       \\
Group 96 &Samian & 56        \\\hline
 &&538                       \\\hline
\end{tabular}
```

Note that line-endings in what you type are not important; rows end only when a \\ is found.

11 Tables and Figures

When you lay out some text and numbers in a table, it is almost always intended that the whole unit stays together, not breaking over a page. What do you do when you are at the bottom of a page with not enough space left? Similarly, what do you do when you want to leave space for a photograph to be pasted in? When you are typing the text, you do not know where LATEX will start a new page, so it is often very difficult to leave space just where you want. LATEX solves this by a system of 'floats', objects that will be placed in a nice position on the page, but not necessarily exactly where you put it in your file. They will often be placed at the bottom of the current page, or at the top of the next, or on a page by themselves, if they need that much space. LATEX lets you specify two sorts of float, namely *tables* and *figures*; in each of them you can specify a caption. The following shows how you would set up space for a figure in which you wanted to leave 3 inches of space, followed by a caption:

```
\begin{figure}
\vspace{3in}
\caption{A Photograph of my Subject}
\end{figure}
```

Figures and tables are automatically numbered, like sections; just as `\tableofcontents` lists your sections, subsections, etc. with their page numbers, `\listoffigures` and `\listoftables` will list the figures and tables with their captions.

12 Cross-References and Citations

One of the most useful things LATEX can do for you is to to automatically generate a section, page, figure, table, or equation number in a cross-reference. For example, to refer to Section 4 of this report (on document styles) and Figure 2 (the output of processing `sample.tex`) in this sentence, the commands used were:

```
to refer to Section~\ref{sec:styles} of this report (on document
styles) and Figure~\ref{fig:result} (the output of processing
```

You will see that the `\ref` command inserted the cross-references number into the printed document. The ~ simply inserts a space while prohibiting LATEX from breaking a line.

Of course, to refer to a cross-reference, there must be an indication in the LATEX input file of the point being referred to. This is done via the `\label` command. The label for a section is most conveniently done immediately after the section heading, as in:

```
\section{Document Classes and Options}
\label{sec:styles}
```

and that for a figure or table in or immediately following the `\caption` command:

```
\caption{The Result of Processing the Sample File}
\label{fig:result}
```

Any text may be used in `\label` and `\ref` commands, though the convention impled above is a good one to follow. Page references use `\label` in the same way, but are referred to using `\pageref` instead of `\ref`.

In much the same way that you can refer to other parts of your document from the LATEX file symbolically, you may cite other documents. To do this, you include the `\cite` command in your text, as in

```
The book by Lamport \cite{Lamport-LaTeX} is the principal
reference work on \LaTeX.
```

The argument to the `\cite` command, known as the *citation key*, is used to uniquely identify a paper, book, or other document that has been described in a *bibliographic database*.

Having cited other work from your document, you can generate a list of references by including the commands

```
\bibliographystyle{plain}
\bibliography{mybib1,mybib2}
```

where you want the list to appear. The `\bibliographystyle` command tells LATEX how you want the citations to appear, the standard ones being:

`plain` Citations are sorted alphabetically in the list, with each entry being assigned a number; entries in the text are indicated in square brackets.

`unsrt` Citations are ordered in the reference list in the order of their first appearance and assigned a number; entries in the text are indicated in square brackets.

`alpha` Citations are sorted alphabetically in the list but have labels like "Lam86" instead of a number; the same label appears in the text in square brackets.

`abbrev` Citations are similar to `alpha` but are more compact (e.g., by abbreviating month and journal names).

The argument of the `\bibliography` command is a comma-separated list of filenames, which should end in `.bib` (i.e., `mybib1.bib` and `mybib2.bib` in the example above). These files contain the expanded references referred to by the `\cite` commands. A full description of the format of these `.bib` files and how the citations are converted to a LATEX-compatible form are beyond the scope of this (introductory) guide; details are given in Appendix B of Lamport's book.

13 Mathematical Typesetting

13.1 Math, Display-math and Equation

Mathematics is treated by TEX completely differently from ordinary text. There are two special *modes* for mathematics, known as *math mode* and *display math mode*.

Math mode commands are surrounded by `\(...\)` or by `$...$`.

Some mathematics set inline $2 \times 3 = 6$. Note that spaces in the input file are ignored in math mode.

```
Some mathematics set inline
\( 2\times 3 =   6 \).
Note that spaces in the input file
are ignored in math mode.
```

Display math mode commands are surrounded by `\[...\]`.

A larger equation to be displayed on a line by itself.

$$f(x) = \sum_{i=0}^{\infty}\frac{f^{(i)}(x)}{i!}$$

```
A larger equation to be displayed on a line by itself
\[ f(x) = \sum_{i=0}^{\infty}\frac{f^{(i)}(x)}{i!} \]
```

There is a variant of display math mode, the `equation` environment, which automatically generates an equation number.

$$\left(\begin{array}{cc} 1 & 2 \\ 0 & 1 \end{array}\right)\left(\begin{array}{cc} 2 & 0 \\ 1 & 3 \end{array}\right) = \left(\begin{array}{cc} 4 & 6 \\ 1 & 3 \end{array}\right) \qquad (1)$$

```
\begin{equation}
\left(\begin{array}{cc}
1 & 2 \\0 & 1
\end{array}\right)
\left(\begin{array}{cc}
2 & 0\\1 & 3
\end{array}\right)
=
\left(\begin{array}{cc}
4 & 6 \\1 & 3
\end{array}\right)
\end{equation}
```

These short examples show the main types of commands available in math mode. A few things to note are:

1. Subscripts and superscripts are produced with _ and ^, as in `x_{1} = p^{2}` for $x_1 = p^2$.
2. Fractions are produced by the `\frac` command: `\(\frac{a + b}{c}\)` gives $\frac{a+b}{c}$.
3. Various commands give names to mathematical symbols:
   `\infty \Rightarrow \surd \bigotimes` generate $\infty \Rightarrow \surd \bigotimes$
4. Arrays are produced by the `array` environment. This is identical to the `tabular` environment described in Section 10 except that the entries are typeset in math mode instead of LR mode. Note that the `array` environment does not put brackets arround the array, so it can also be used for setting determinants—or even sets of equations in which you want the columns to line up.
5. The commands `\left` and `\right` produce delimiters that grow as large as needed. They can be used with a variety of symbols, e.g., `\left( \left\{ \left|`. The full set of these delimiters is shown in tables 5 and 4 below.

13.2 Spacing

All spaces in the input file are ignored in math mode. Sometimes you may want to adjust the spacing; use one of the following commands to achieve this.

`\,` thin space	`\:` medium space
`\!` negative thin space	`\;` thick space

A good example of where LATEX needs some help with spacing is

$\int\!\!\int z\, dx dy .. \int\int z dx dy$ `\(\int\!\!\int z\, dx dy .. \int\int z dx dy\)`

13.3 Changing Fonts in Math Mode

The default math mode font is $math\ italic$. This should not be confused with ordinary *text italic*. The font for 'ordinary' letters can be changed with the usual commands, `\emph`, `\textbf`, etc.. Note that lower case Greek letters (`\alpha`, etc.) are regarded as mathematical symbols (which means that they must be typed in math mode) and are not affected by these commands.

`\mathbf` produces **bold face roman** letters. If you wish to have $\boldsymbol{bold\ face\ math\ italic}$ letters, and bold face Greek letters and mathematical symbols, use the `\boldmath` command *before* going into math mode. This changes the default math fonts to bold.

$x = 2\pi \Rightarrow x \simeq 6.28$ $\mathbf{x} = 2\pi \Rightarrow x \simeq 6.28$ $\boldsymbol{x = 2\pi \Rightarrow x \simeq 6.28}$

```
\( x = 2\pi \Rightarrow x \simeq 6.28 \)
\(\mathbf{x} = 2\pi \Rightarrow x \simeq 6.28 \)
{\boldmath \(x = \mathbf{2}\pi \Rightarrow x
          \simeq{\mathbf{6.28}}             \)}
```

There is also a calligraphic font for upper case letters; these are produced by the `\mathcal` command:

$\mathcal{F}$

```
\( \mathcal{F} \)
```

14 What about $'s?

If you have converted to LaTeX from plain or AMS-TeX, you will probably be wondering why there has been no mention of $ and $$.

In these systems math mode is surrounded by `$`'s and display math mode is surrounded by `$$`. This has certain drawbacks over the LaTeX system as it is difficult for your text editor to match `$`'s as it is hard to tell which ones are starting math mode and which are ending it. TeX will also get confused if you miss a $ out.

The (incorrect) input

```
let (a,b,c)$ be a Pythagorean triple, i.e.\ three
integers such that $a^{2}+b^{2}=c^{2}$.
```

produces the slightly mysterious error message

```
! Missing $ inserted.
<inserted text>
                $
<to be read again>
                   ^
l.56 ...triple, i.e.\ three integers such that $a^
                                                  {2}+b^{2}=c^{2}$
?
```

Note that it reports the wrong error and in the wrong place, the use of the ^ command out of math mode. TeX has typeset 'be a … such that' in math mode and *exited* math mode at the `$` after 'such that'. If you had made the equivalent LaTeX error, LaTeX has a better idea of what you indended:

```
let (a,b,c)\) be a Pythagorean triple, i.e.\ three
integers such that \(a^{2}+b^{2}=c^{2}\)
```

The error message may still be unintelligable, but at least it reports the error in the right place, you have used `\)` to end math mode when you were not in math mode (as you omitted the `\(` which should have been before the `(a,b,c)`).

```
LaTeX error.  See LaTeX manual for explanation.
              Type  H <return>  for immediate help.
! Bad math environment delimiter.
\@latexerr ...for immediate help.}\errmessage {#1}
```

```
\)...ifinner $\else \@badmath \fi \else \@badmath
                                                   \fi
l.56 let (a,b,c)\)
                   be a Pythagorean triple, i.e.\ three integers such that \...

?
```

The single dollar is sometimes useful for small sections of math.

Let G be a p-group `Let $G$ be a $p$-group`

The double dollar is *not* always equivalent to `\[ ... \]`, and so should not be used if you want your LaTeX file to be compatible with different styles and options (try the *fleqn* class option).

14.1 Symbols

The following tables show most of the symbols available from the standard LaTeX symbol fonts. Negations of the relational symbols can be made with the `\not` command:

$G \not\equiv H$ `$G \not\equiv H$`

In the appendix we have brought together a set of tables regrouping all standard LaTeX symbols (tables 1 to 11. Tables 13 to 12 show supplementary $\mathcal{AMS}$-TEX symbols, which are available when you specify the `amssymb` package.

15 Letters

Producing letters is simple with LaTeX. To do this you use the document style `letter`. You can make any number of letters with a single input file. Your name and address, which are likely to be the same for all letters, are given once at the top of the file. Each letter is produced by a `letter` environment, having the name and address of the recipient as its argument. The letter itself begins with an `\opening` command to generate the salutation.

The letter ends with a `\closing` command. You can use the commands `\encl` and `\cc` to generate lists of enclosures and people to whom you are sending copies. Any text that follows the `\closing` must be proceeded by a `\ps` command. This command produces no text—you'll have to type "P.S." yourself—but is needed to format the additional text correctly.

An example will make this clearer:

```
\documentstyle{letter}
\begin{document}

\address{1234 Avenue of the Armadillos \\
         Gnu York, G.Y. 56789}
\signature{R. (Ma) Dillo \\ Director of Cuisine}

\begin{letter}{G. Natheniel Picking \\
               Acme Exterminators \\
               Illinois}

\opening{Dear Nat,}

I'm afraid that the armadillo problem is still with us.
I did everything ...

... and I hope that you can get rid of the nasty
```

```
beasts this time.

\closing{Best Regards,}
\cc{Jimmy Carter\\Richard M. Nixon}
\end{letter}

\end{document}
```

16 Errors

When you create a new input file for LaTeX, you will probably make mistakes. Everybody does, and it's nothing to be worried about. As with most computer programs, there are two sorts of mistake that you can make: those that LaTeX notices and those that it doesn't. To take a rather silly example, since LaTeX doesn't understand what you are typing, it isn't going to be worried if you mis-spell some of the words in your text; you will just have to accurately proof-read your printed output. On the other hand, if you mis-spell one of the environment names in your file then LaTeX won't know what you want it to do.

When this sort of thing happens, LaTeX prints an error message on your terminal screen and then stops and waits for you to take some action. The error messages that it produces may seem user-unfriendly and not a little frightening at first. Nevertheless, if you know where to look they will probably tell you where the error is and what went wrong.

Consider what would happen if you mistyped `\begin{itemize}` as `\begin{itemie}`. When LaTeX processes this instruction, it displays the following on your terminal:

```
LaTeX error.  See LaTeX manual for explanation.
              Type  H <return>  for immediate help.
! Environment itemie undefined.
\@latexerr ...for immediate help.}\errmessage {#1}
                                                    \endgroup
l.140 \begin{itemie}

?
```

After typing the '?' LaTeX stops and waits for you to tell it what to do.

The first two lines of the message just tell you that the error was detected by LaTeX. The third line, the one that starts '!' is the *error indicator*. It tells you what the problem is, though until you have had some experience of LaTeX this may not mean a lot to you. In this case it is just telling you that it doesn't recognise an environment called `itemie`. The next two lines tell you what LaTeX was doing when it found the error; they are irrelevant at the moment and can be ignored. The final line is called the *error locator*, and is a copy of the line in your file that caused the problem. It starts with a line number to help you to find it in your file. If the error was in the middle of a line, it will be shown broken at the point where LaTeX realized that there was an error. As with all computer programs, LaTeX can sometimes pass the point where the real error was before discovering that something is wrong, but it doesn't usually get very far.

At this point you could do several things. If you knew enough about LaTeX you might be able to fix the problem, or you could type '`X`' and press the return key to stop LaTeX running while you go and correct the error. The best thing to do, however, is just to press the return key. This will allow LaTeX to go on running as if nothing had happened. If you have made one mistake, then you have probably made several and you may as well try to find them all in one go. It's much more efficient to do it this way than to run LaTeX over and over again fixing one error at a time. Don't worry about remembering what the errors were—a copy of all the error messages is being saved in a *log* file so that you can look at them afterwards. See your *local guide* to find out what this file is called.

If you look at the line that caused the error, it's normally obvious what the problem was. If you can't work out what your problem is, look at the hints below and, if they don't help, consult

Chapter 6 of Lamport's book: it contains a list of all of the error messages that you are likely to encounter, together with some suggestions as to what may have caused them.

Some of the most common mistakes that cause errors are:

- A mis-spelt command or environment name.
- Improperly matched '`{`' and '`}`'—remember that they should always come in pairs.
- Trying to use one of the ten special characters `# $ % & _ { } ~ ^` and \ as an ordinary printing symbol.
- A missing `\end` command.
- A missing command argument (that's the bit enclosed in '`{`' and '`}`').

A single error can get LATEX so confused that it reports a series of spurious errors as a result. If you have an error that you understand, followed by a series that you don't, then try correcting the first error—the rest may vanish as if by magic.

Sometimes LATEX may write a `*` and stop without an error message. This is normally caused by a missing `\end{document}` command, though other errors can cause it. If this happens type `\stop` and press the return key.

Finally, LATEX will sometimes print *warning* messages. They report problems that were not bad enough to cause LATEX to stop processing, but nevertheless may require investigation. The most common problems are 'overfull' and 'underfull' lines of text. A message like:

```
Overfull \hbox (10.58649pt too wide) in paragraph at lines 172--175
[]\tenrm Mathematical for-mu-las may be dis-played. A dis-played
```

indicates that LATEX could not find a good place to break a line when laying out a paragraph. As a result, it was forced to let the line stick out into the right-hand margin, in this case by 10.6 points. Since a point is $1/72.27$ inches, this may be rather hard to see, but it will be there nonetheless.

This particular problem happens because LATEX is rather fussy about line breaking, and it would rather generate a line that is too long than generate a paragraph that doesn't meet its high standards. The simplest way around the problem is to enclose the entire offending paragraph between `\begin{sloppypar}` and `\end{sloppypar}` commands. This tells LATEX that you are happy for it to break its own rules while it is working on that particular bit of text.

Alternatively, messages about "Underfull `\hboxes`" may appear. These are lines that had to have more space inserted between words than LATEX would have liked. In general there is not much that you can do about these. Your output will look fine, even if the line looks a bit stretched. About the only thing you could do is to re-write the offending paragraph!

17 A Final Reminder

You now know enough LATEX to produce a wide range of documents. But this document has only scratched the surface of the things that LATEX can do. This entire document was itself produced with LATEX (with no sticking things in or clever use of a photocopier) and even it hasn't used all the features that it could. From this you may get some feeling for the power that LATEX puts at your disposal.

Please remember what was said in the introduction: if you *do* have a complex document to produce then *go and read Lamport's book*. You will be wasting your time if you rely only on what you have read here. Once you have used LATEX for some time, you might want to know how to use the many style file extensions, which have been developed to aid LATEX users to typeset their documents. This and much else (like how to use *MakeIndex*, for generating indices, or BIBTEX, for managing your bibliographies) is covered in *The LATEX Companion*[3]. Finally the final word about TEX is Knuth's *TEX book*[2].

And one other warning: having dabbled with LATEX your documents will never be the same again...

A Mathematical symbols

α	`\alpha`	β	`\beta`	γ	`\gamma`	δ	`\delta`
ϵ	`\epsilon`	ε	`\varepsilon`	ζ	`\zeta`	η	`\eta`
θ	`\theta`	ϑ	`\vartheta`	ι	`\iota`	κ	`\kappa`
λ	`\lambda`	μ	`\mu`	ν	`\nu`	ξ	`\xi`
o	`o`	π	`\pi`	ϖ	`\varpi`	ρ	`\rho`
ϱ	`\varrho`	σ	`\sigma`	ς	`\varsigma`	τ	`\tau`
υ	`\upsilon`	ϕ	`\phi`	φ	`\varphi`	χ	`\chi`
ψ	`\psi`	ω	`\omega`				
Γ	`\Gamma`	Δ	`\Delta`	Θ	`\Theta`	Λ	`\Lambda`
Ξ	`\Xi`	Π	`\Pi`	Σ	`\Sigma`	Υ	`\Upsilon`
Φ	`\Phi`	Ψ	`\Psi`	Ω	`\Omega`		

Table 1: Greek letters

$\pm$	`\pm`	$\cap$	`\cap`	$\diamond$	`\diamond`	$\oplus$	`\oplus`
$\mp$	`\mp`	$\cup$	`\cup`	$\bigtriangleup$	`\bigtriangleup`	$\ominus$	`\ominus`
$\times$	`\times`	$\uplus$	`\uplus`	$\bigtriangledown$	`\bigtriangledown`	$\otimes$	`\otimes`
$\div$	`\div`	$\sqcap$	`\sqcap`	$\triangleleft$	`\triangleleft`	$\oslash$	`\oslash`
$\ast$	`\ast`	$\sqcup$	`\sqcup`	$\triangleright$	`\triangleright`	$\odot$	`\odot`
$\star$	`\star`	$\vee$	`\vee`	$\lhd$	`\lhd`[a]	$\bigcirc$	`\bigcirc`
$\circ$	`\circ`	$\wedge$	`\wedge`	$\rhd$	`\rhd`[a]	$\dagger$	`\dagger`
$\bullet$	`\bullet`	$\setminus$	`\setminus`	$\unlhd$	`\unlhd`[a]	$\ddagger$	`\ddagger`
$\cdot$	`\cdot`	$\wr$	`\wr`	$\unrhd$	`\unrhd`[a]	$\amalg$	`\amalg`

[a] Not predefined in LaTeX 2_ε. Use the packages `latexsym` or `amssymb`

Table 2: Binary operation symbols

$\leq$	`\leq`	$\geq$	`\geq`	$\equiv$	`\equiv`	$\models$	`\models`	$\prec$	`\prec`
$\succ$	`\succ`	$\sim$	`\sim`	$\perp$	`\perp`	$\preceq$	`\preceq`	$\succeq$	`\succeq`
$\simeq$	`\simeq`	$\mid$	`\mid`	$\ll$	`\ll`	$\gg$	`\gg`	$\asymp$	`\asymp`
$\parallel$	`\parallel`	$\subset$	`\subset`	$\supset$	`\supset`	$\approx$	`\approx`	$\bowtie$	`\bowtie`
$\subseteq$	`\subseteq`	$\supseteq$	`\supseteq`	$\cong$	`\cong`	$\Join$	`\Join`	$\sqsubset$	`\sqsubset`
$\sqsupset$	`\sqsupset`	$\neq$	`\neq`	$\smile$	`\smile`	$\sqsubseteq$	`\sqsubseteq`	$\sqsupseteq$	`\sqsupseteq`
$\doteq$	`\doteq`	$\frown$	`\frown`	$\in$	`\in`	$\ni$	`\ni`	$\propto$	`\propto`
$=$	`=`	$\vdash$	`\vdash`	$\dashv$	`\dashv`	$<$	`<`	$>$	`>`

Table 3: Relation symbols

$\rmoustache$	`\rmoustache`	$\lmoustache$	`\lmoustache`	$\rgroup$	`\rgroup`	$\lgroup$	`\lgroup`
$\arrowvert$	`\arrowvert`	$\Arrowvert$	`\Arrowvert`	$\bracevert$	`\bracevert`		

Table 4: Large delimiters

<table>
<tr><td>$\uparrow$</td><td><code>\uparrow</code></td><td>$\Uparrow$</td><td><code>\Uparrow</code></td><td>$\downarrow$</td><td><code>\downarrow</code></td><td>$\Downarrow$</td><td><code>\Downarrow</code></td></tr>
<tr><td>$\{$</td><td><code>\{</code></td><td>$\}$</td><td><code>\}</code></td><td>$\updownarrow$</td><td><code>\updownarrow</code></td><td>$\Updownarrow$</td><td><code>\Updownarrow</code></td></tr>
<tr><td>$\lfloor$</td><td><code>\lfloor</code></td><td>$\rfloor$</td><td><code>\rfloor</code></td><td>$\lceil$</td><td><code>\lceil</code></td><td>$\rceil$</td><td><code>\rceil</code></td></tr>
<tr><td>$\langle$</td><td><code>\langle</code></td><td>$\rangle$</td><td><code>\rangle</code></td><td>$/$</td><td><code>/</code></td><td>$\backslash$</td><td><code>\backslash</code></td></tr>
<tr><td>$\vert$</td><td><code>|</code></td><td>$\Vert$</td><td><code>\|</code></td><td></td><td></td><td></td><td></td></tr>
</table>

Table 5: Delimiters

$\leftarrow$	`\leftarrow`	$\longleftarrow$	`\longleftarrow`	$\uparrow$	`\uparrow`
$\Leftarrow$	`\Leftarrow`	$\Longleftarrow$	`\Longleftarrow`	$\Uparrow$	`\Uparrow`
$\rightarrow$	`\rightarrow`	$\longrightarrow$	`\longrightarrow`	$\downarrow$	`\downarrow`
$\Rightarrow$	`\Rightarrow`	$\Longrightarrow$	`\Longrightarrow`	$\Downarrow$	`\Downarrow`
$\leftrightarrow$	`\leftrightarrow`	$\longleftrightarrow$	`\longleftrightarrow`	$\updownarrow$	`\updownarrow`
$\Leftrightarrow$	`\Leftrightarrow`	$\Longleftrightarrow$	`\Longleftrightarrow`	$\Updownarrow$	`\Updownarrow`
$\mapsto$	`\mapsto`	$\longmapsto$	`\longmapsto`	$\nearrow$	`\nearrow`
$\hookleftarrow$	`\hookleftarrow`	$\hookrightarrow$	`\hookrightarrow`	$\searrow$	`\searrow`
$\leftharpoonup$	`\leftharpoonup`	$\rightharpoonup$	`\rightharpoonup`	$\swarrow$	`\swarrow`
$\leftharpoondown$	`\leftharpoondown`	$\rightharpoondown$	`\rightharpoondown`	$\nwarrow$	`\nwarrow`

Table 6: Arrow symbols

$\ldots$	`\ldots`	$\cdots$	`\cdots`	$\vdots$	`\vdots`	$\ddots$	`\ddots`	$\aleph$	`\aleph`
$\prime$	`\prime`	$\forall$	`\forall`	∞	`\infty`	$\hbar$	`\hbar`	$\emptyset$	`\emptyset`
$\exists$	`\exists`	∇	`\nabla`	$\surd$	`\surd`	$\Box$	`\Box`[a]	$\triangle$	`\triangle`
$\Diamond$	`\Diamond`[a]	$\imath$	`\imath`	$\jmath$	`\jmath`	ℓ	`\ell`	$\neg$	`\neg`
$\top$	`\top`	$\flat$	`\flat`	$\natural$	`\natural`	$\sharp$	`\sharp`	$\wp$	`\wp`
$\bot$	`\bot`	$\clubsuit$	`\clubsuit`	$\diamondsuit$	`\diamondsuit`	$\heartsuit$	`\heartsuit`	$\spadesuit$	`\spadesuit`
$\mho$	`\mho`[a]	$\Re$	`\Re`	$\Im$	`\Im`	$\angle$	`\angle`	∂	`\partial`

[a] Not predefined in LaTeX 2_ε. Use the packages `latexsym` or `amssymb`

Table 7: Miscellaneous symbols

\arccos	\cos	\csc	\exp	\ker	\limsup	\min	\sinh
\arcsin	\cosh	\deg	\gcd	\lg	\ln	\Pr	\sup
\arctan	\cot	\det	\hom	\lim	\log	\sec	\tan
\arg	\coth	\dim	\inf	\liminf	\max	\sin	\tanh

Table 8: Log-like symbols

$\hat{a}$	\hat{a}	$\acute{a}$	\acute{a}	$\bar{a}$	\bar{a}	$\dot{a}$	\dot{a}	$\breve{a}$	\breve{a}
$\check{a}$	\check{a}	$\grave{a}$	\grave{a}	$\vec{a}$	\vec{a}	$\ddot{a}$	\ddot{a}	$\tilde{a}$	\tilde{a}

Table 9: Math mode accents

$\sum$	\sum	$\prod$	\prod	$\coprod$	\coprod	$\int$	\int	$\oint$	\oint
$\bigcap$	\bigcap	$\bigcup$	\bigcup	$\bigsqcup$	\bigsqcup	$\bigvee$	\bigvee	$\bigwedge$	\bigwedge
$\bigodot$	\bigodot	$\bigotimes$	\bigotimes	$\bigoplus$	\bigoplus	$\biguplus$	\biguplus		

Table 10: Variable-sized symbols

$\widetilde{abc}$	\widetilde{abc}	$\widehat{abc}$	\widehat{abc}
$\overleftarrow{abc}$	\overleftarrow{abc}	$\overrightarrow{abc}$	\overrightarrow{abc}
$\overline{abc}$	\overline{abc}	$\underline{abc}$	\underline{abc}
$\overbrace{abc}$	\overbrace{abc}	$\underbrace{abc}$	\underbrace{abc}
$\sqrt{abc}$	\sqrt{abc}	$\sqrt[n]{abc}$	\sqrt[n]{abc}
f'	f'	$\frac{abc}{xyz}$	\frac{abc}{xyz}

Table 11: LATEX math constructs

$\hbar$	\hbar	$\hslash$	\hslash	$\vartriangle$	\vartriangle
$\triangledown$	\triangledown	$\square$	\square	$\lozenge$	\lozenge
$\circledS$	\circledS	$\angle$	\angle	$\measuredangle$	\measuredangle
$\nexists$	\nexists	$\mho$	\mho	$\Finv$	\Finv[a]
$\Game$	\Game[a]	$\Bbbk$	\Bbbk[a]	$\backprime$	\backprime
$\varnothing$	\varnothing	$\blacktriangle$	\blacktriangle	$\blacktriangledown$	\blacktriangledown
$\blacksquare$	\blacksquare	$\blacklozenge$	\blacklozenge	$\bigstar$	\bigstar
$\sphericalangle$	\sphericalangle	$\complement$	\complement	$\eth$	\eth
$\diagup$	\diagup[a]	$\diagdown$	\diagdown[a]		

[a] Not defined in style amssymb, define using the LATEX 2_ε \DeclareMathSymbol command

Table 12: AMS miscellaneous symbols

$\digamma$	\digamma	$\varkappa$	\varkappa	$\beth$	\beth	$\daleth$	\daleth	$\gimel$	\gimel

Table 13: AMS Greek and Hebrew

$\ulcorner$	\ulcorner	$\urcorner$	\urcorner	$\llcorner$	\llcorner	$\lrcorner$	\lrcorner

Table 14: AMS delimiters

$\dashrightarrow$	\dashrightarrow	$\dashleftarrow$	\dashleftarrow	$\leftleftarrows$	\leftleftarrows
$\leftrightarrows$	\leftrightarrows	$\Lleftarrow$	\Lleftarrow	$\twoheadleftarrow$	\twoheadleftarrow
$\leftarrowtail$	\leftarrowtail	$\looparrowleft$	\looparrowleft	$\leftrightharpoons$	\leftrightharpoons
$\curvearrowleft$	\curvearrowleft	$\circlearrowleft$	\circlearrowleft	$\Lsh$	\Lsh
$\upuparrows$	\upuparrows	$\upharpoonleft$	\upharpoonleft	$\downharpoonleft$	\downharpoonleft
$\multimap$	\multimap	$\leftrightsquigarrow$	\leftrightsquigarrow	$\rightrightarrows$	\rightrightarrows
$\rightleftarrows$	\rightleftarrows	$\rightrightarrows$	\rightrightarrows	$\rightleftarrows$	\rightleftarrows
$\twoheadrightarrow$	\twoheadrightarrow	$\rightarrowtail$	\rightarrowtail	$\looparrowright$	\looparrowright
$\rightleftharpoons$	\rightleftharpoons	$\curvearrowright$	\curvearrowright	$\circlearrowright$	\circlearrowright
$\Rsh$	\Rsh	$\downdownarrows$	\downdownarrows	$\upharpoonright$	\upharpoonright
$\downharpoonright$	\downharpoonright	$\rightsquigarrow$	\rightsquigarrow		

Table 15: AMS arrows

$\nleftarrow$	\nleftarrow	$\nrightarrow$	\nrightarrow	$\nLeftarrow$	\nLeftarrow
$\nRightarrow$	\nRightarrow	$\nleftrightarrow$	\nleftrightarrow	$\nLeftrightarrow$	\nLeftrightarrow

Table 16: AMS negated arrows

$\dotplus$	\dotplus	$\smallsetminus$	\smallsetminus	$\Cap$	\Cap
$\Cup$	\Cup	$\barwedge$	\barwedge	$\veebar$	\veebar
$\doublebarwedge$	\doublebarwedge	$\boxminus$	\boxminus	$\boxtimes$	\boxtimes
$\boxdot$	\boxdot	$\boxplus$	\boxplus	$\divideontimes$	\divideontimes
$\ltimes$	\ltimes	$\rtimes$	\rtimes	$\leftthreetimes$	\leftthreetimes
$\rightthreetimes$	\rightthreetimes	$\curlywedge$	\curlywedge	$\curlyvee$	\curlyvee
$\circleddash$	\circleddash	$\circledast$	\circledast	$\circledcirc$	\circledcirc
$\centerdot$	\centerdot	$\intercal$	\intercal		

Table 17: AMS binary operators

$\leqq$	\leqq	$\leqslant$	\leqslant	$\eqslantless$	\eqslantless
$\lesssim$	\lesssim	$\lessapprox$	\lessapprox	$\approxeq$	\approxeq
$\lessdot$	\lessdot	$\lll$	\lll	$\lessgtr$	\lessgtr
$\lesseqgtr$	\lesseqgtr	$\lesseqqgtr$	\lesseqqgtr	$\doteqdot$	\doteqdot
$\risingdotseq$	\risingdotseq	$\fallingdotseq$	\fallingdotseq	$\backsim$	\backsim
$\backsimeq$	\backsimeq	$\subseteqq$	\subseteqq	$\Subset$	\Subset
$\sqsubset$	\sqsubset	$\preccurlyeq$	\preccurlyeq	$\curlyeqprec$	\curlyeqprec
$\precsim$	\precsim	$\precapprox$	\precapprox	$\vartriangleleft$	\vartriangleleft
$\trianglelefteq$	\trianglelefteq	$\vDash$	\vDash	$\Vvdash$	\Vvdash
$\smallsmile$	\smallsmile	$\smallfrown$	\smallfrown	$\bumpeq$	\bumpeq
$\Bumpeq$	\Bumpeq	$\geqq$	\geqq	$\geqslant$	\geqslant
$\eqslantgtr$	\eqslantgtr	$\gtrsim$	\gtrsim	$\gtrapprox$	\gtrapprox
$\gtrdot$	\gtrdot	$\ggg$	\ggg	$\gtrless$	\gtrless
$\gtreqless$	\gtreqless	$\gtreqqless$	\gtreqqless	$\eqcirc$	\eqcirc
$\circeq$	\circeq	$\triangleq$	\triangleq	$\thicksim$	\thicksim
$\thickapprox$	\thickapprox	$\supseteqq$	\supseteqq	$\Supset$	\Supset
$\sqsupset$	\sqsupset	$\succcurlyeq$	\succcurlyeq	$\curlyeqsucc$	\curlyeqsucc
$\succsim$	\succsim	$\succapprox$	\succapprox	$\vartriangleright$	\vartriangleright
$\trianglerighteq$	\trianglerighteq	$\Vdash$	\Vdash	$\shortmid$	\shortmid
$\shortparallel$	\shortparallel	$\between$	\between	$\pitchfork$	\pitchfork
$\varpropto$	\varpropto	$\blacktriangleleft$	\blacktriangleleft	$\therefore$	\therefore
$\backepsilon$	\backepsilon	$\blacktriangleright$	\blacktriangleright	$\because$	\because

Table 18: AMS binary relations

$\nless$	\nless	$\nleq$	\nleq	$\nleqslant$	\nleqslant
$\nleqq$	\nleqq	$\lneq$	\lneq	$\lneqq$	\lneqq
$\lvertneqq$	\lvertneqq	$\lnsim$	\lnsim	$\lnapprox$	\lnapprox
$\nprec$	\nprec	$\npreceq$	\npreceq	$\precnsim$	\precnsim
$\precnapprox$	\precnapprox	$\nsim$	\nsim	$\nshortmid$	\nshortmid
$\nmid$	\nmid	$\nvdash$	\nvdash	$\nvDash$	\nvDash
$\ntriangleleft$	\ntriangleleft	$\ntrianglelefteq$	\ntrianglelefteq	$\nsubseteq$	\nsubseteq
$\subsetneq$	\subsetneq	$\varsubsetneq$	\varsubsetneq	$\subsetneqq$	\subsetneqq
$\varsubsetneqq$	\varsubsetneqq	$\ngtr$	\ngtr	$\ngeq$	\ngeq
$\ngeqslant$	\ngeqslant	$\ngeqq$	\ngeqq	$\gneq$	\gneq
$\gneqq$	\gneqq	$\gvertneqq$	\gvertneqq	$\gnsim$	\gnsim
$\gnapprox$	\gnapprox	$\nsucc$	\nsucc	$\nsucceq$	\nsucceq
$\succnsim$	\succnsim	$\succnapprox$	\succnapprox	$\ncong$	\ncong
$\nshortparallel$	\nshortparallel	$\nparallel$	\nparallel	$\nvDash$	\nvDash
$\nVDash$	\nVDash	$\ntriangleright$	\ntriangleright	$\ntrianglerighteq$	\ntrianglerighteq
$\nsupseteq$	\nsupseteq	$\nsupseteqq$	\nsupseteqq	$\supsetneq$	\supsetneq
$\varsupsetneq$	\varsupsetneq	$\supsetneqq$	\supsetneqq	$\varsupsetneqq$	\varsupsetneqq

Table 19: AMS negated binary relations

B Horrible Mathematical Examples to Study

$$\phi(t)=\frac{1}{\sqrt{2\pi}}\int_0^t e^{-x^2/2}dx \tag{2}$$

```
\begin{equation}
  \phi(t)=\frac{1}{\sqrt{2\pi}}
  \int^t_0 e^{-x^2/2} dx
\end{equation}
```

$$\prod_{j\geq 0}\left(\sum_{k\geq 0}a_{jk}z^k\right)=\sum_{k\geq 0}z^n\left(\sum_{k_0,k_1,\ldots\geq 0 \atop k_0+k_1+\ldots=n}a_0k_0a_{1k_1}\ldots\right) \tag{3}$$

```
\begin{equation}
  \prod_{j\geq 0}
  \left(\sum_{k\geq 0}a_{jk} z^k\right)
= \sum_{k\geq 0} z^n
  \left( \sum_{{k_0,k_1,\ldots\geq 0}
          \atop{k_0+k_1+\ldots=n}    }
        a{_0k_0}a_{1k_1}\ldots  \right)
\end{equation}
```

$$\pi(n)=\sum_{m=2}^{n}\left\lfloor\left(\sum_{k=1}^{m-1}\lfloor(m/k)/\lceil m/k\rceil\rfloor\right)^{-1}\right\rfloor \tag{4}$$

```
\begin{equation}
\pi(n) = \sum_{m=2}^{n}
  \left\lfloor \left(\sum_{k=1}^{m-1}
       \lfloor(m/k)/\lceil m/k\rceil
       \rfloor \right)^{-1}
  \right\rfloor
\end{equation}
```

$$\{\underbrace{\overbrace{\mathstrut a,\ldots,a}^{k\ a's},\overbrace{\mathstrut b,\ldots,b}^{l\ b's}}_{k+1\ \mathrm{elements}}\} \tag{5}$$

```
\begin{equation}
\{\underbrace{%
    \overbrace{\mathstrut a,\ldots,a}^{k\ a’s},
    \overbrace{\mathstrut b,\ldots,b}^{l\ b’s}}
  _{k+1\ \mathrm{elements}}                      \}
\end{equation}
```

$$\mbox{W}^+\ \begin{array}{l}
\nearrow\raise5pt\hbox{$\mu^+ + \nu_{\mu}$}\\
\rightarrow \pi^+ +\pi^0 \\[5pt]
\rightarrow \kappa^+ +\pi^0 \\
\searrow\lower5pt\hbox{$\mathrm{e}^+ +\nu_{\scriptstyle\mathrm{e}}$}
\end{array}$$

```
\begin{displaymath}
\mbox{W}^+\
\begin{array}{l}
\nearrow\raise5pt\hbox{$\mu^+ + \nu_{\mu}$}\\
\rightarrow          \pi^+ +\pi^0          \\[5pt]
\rightarrow \kappa^+ +\pi^0                \\
\searrow\lower5pt\hbox{$\mathrm{e}^+
            +\nu_{\scriptstyle\mathrm{e}}$}
\end{array}
\end{displaymath}
```

$$F(x,y)=0\quad\mathrm{and}\quad\left|\begin{array}{ccc}
F_{xx}'' & F_{xy}'' & F_{x}' \\
F_{yx}'' & F_{yy}'' & F_{y}' \\
F_{x}' & F_{y}' & 0
\end{array}\right|=0$$

```
\begin{displaymath}
{F}(x,y)=0\quad\mathrm{and}\quad
 \left|\begin{array}{ccc}
  F_{xx}’’ & F_{xy}’’ & F_{x}’ \\
  F_{yx}’’ & F_{yy}’’ & F_{y}’ \\
  F_{x}’   & F_{y}’   & 0
       \end{array}      \right| =0
\end{displaymath}
```

$$\frac{\pm\left|\begin{array}{ccc} x_1-x_2 & y_1-y_2 & z_1-z_2\\ l_1 & m_1 & n_1\\ l_2 & m_2 & n_2\end{array}\right|}{\sqrt{\left|\begin{array}{cc}l_1&m_1\\ l_2&m_2\end{array}\right|^2+\left|\begin{array}{cc}m_1&n_1\\ n_1&l_1\end{array}\right|^2+\left|\begin{array}{cc}m_2&n_2\\ n_2&l_2\end{array}\right|^2}}$$

```
\begin{displaymath}
\frac{\pm
\left|\begin{array}{ccc}
x_1-x_2  & y_1-y_2 & z_1-z_2 \\
l_1      & m_1     & n_1     \\
l_2      & m_2     & n_2
\end{array}\right|}{
\sqrt{\left|\begin{array}{cc}l_1&m_1\\
l_2&m_2\end{array}\right|^2
+     \left|\begin{array}{cc}m_1&n_1\\
n_1&l_1\end{array}\right|^2
+     \left|\begin{array}{cc}m_2&n_2\\
n_2&l_2\end{array}\right|^2}}
\end{displaymath}
```

$$\begin{aligned}
\sigma^f_0(Q,T_{3R},\beta,s) \quad = \quad & \frac{4\pi\alpha^2}{3s}\beta \times \left[\frac{Q^2\left\{\frac{\displaystyle 3-\beta^2}{\displaystyle 2}\right\} - 2QC_{\rm V}{C'}_{\rm V}s(s-M^2_{\rm Z)}}{{(s-M^2_{\rm Z)}}^2 + M^2_{\rm Z}\Gamma^2_{\rm Z}\left\{\frac{\displaystyle 3-\beta^2}{\displaystyle 2}\right\}} \right. \\
+ \quad & \left. \frac{(C_{\rm V}^2+C_{\rm A}^2)s^2}{{(s-M^2_{\rm Z)}}^2+M^2_{\rm Z}\Gamma^2_{\rm Z}\left\{{C'}^2_{\rm V}\left\{\frac{\displaystyle 3-\beta^2}{\displaystyle 2}\right\}+{C'}^2_{\rm A}\{\beta^2\}\right\}}\right]
\end{aligned} \qquad (6)$$

```
\newcommand{\CA}{C_{\rm A}}      \newcommand{\CV}{C_{\rm V}}
\newcommand{\CPA}{{C'}_{\rm A}} \newcommand{\CPV}{{C'}_{\rm V}}
\newcommand{\GZ}{\Gamma^2_{\rm Z}}
\newcommand{\MZ}{M^2_{\rm Z}}    \newcommand{\MZs}{{(s-M^2_{\rm Z})}}
\newcommand{\BE}{\left\{\frac{\displaystyle 3-\beta^2}{\displaystyle 2}\right\}}
\begin{eqnarray}
\sigma^f_0(Q,T_{3R},\beta,s) & = &
      \frac{4\pi\alpha^2}{3s}\beta \times
        \left[ \frac{Q^2 \BE - 2Q \CV \CPV s \MZs}{\MZs^2 + \MZ \GZ \BE}
          \right.                                 \nonumber \\[-3mm]
                             &   &                          \\[-3mm]
                             & + &
      \left.\frac{(\CV^2 + \CA^2) s^2}%
                 {\MZs^2+\MZ\GZ\left\{\CPV^2 \BE+\CPA^2 \{\beta^2\}\right\}}
        \right]                                   \nonumber
\end{eqnarray}
```

Bibliography

1 Leslie Lamport. *LATEX—A Document Preparation System—User's Guide and Reference Manual*. Addison-Wesley, Reading, MA, USA, 1985.

2 Donald E. Knuth. *The TEXbook*, volume A of *Computers and Typesetting*. Addison-Wesley, Reading, MA, USA, 1986.

3 Michel Goossens, Frank Mittelbach and Alexander Samarin. *The LATEX Companion* Addison-Wesley, Reading, MA, USA, 1993.