Module-1

Beginning typesetting Using Latex

1 Introduction

LaTeX is the de-facto standard for typesetting scientific documents. It is an accepted format for submission to a number of scientific journals, as well as many professional publishers. Its popularity stems from a history of flexibility, power and just 'doing the right thing'. LaTeX automatically numbers your equations and figures, lays content out according to strict style rules so as to obtain a readable, polished finish, and rarely gets in the way of the task of writing!

In this document, I introduce many of the LaTeX commands and concepts required to produce a reasonable scientific document of any length, emphasising some of the more complicated aspects while skipping over the simpler things. In some cases, these simple things are silently included as part of a more involved example, so look closely at the example code provided!

2 Starting a LaTeX Document

A LaTeX document must have a preamble (the stuff that goes before \begin{document}). If your LaTeX is split across multiple files, only the top-level file (the one from which you include the others, and run the latex program against) needs the preamble. The preamble is where you specify the document class, include packages to provide useful functionality, and define any custom commands you need. At the very least, you must specify a document class.

```
\documentclass[10pt,a4paper]{article}
\usepackage{amsmath,amsthm,amssymb}
```

The LaTeX above starts a document with the *article* document class. Several document classes exist, but *article* serves for most purposes. Two others see some use; these are *book* and *report*. I generally prefer to modify the *article* class to suit my purposes, since it is usually the most sensible. The main argument to the command is enclosed in { and }, and arguments which are passed on to the *article* class are in [and]. In this case, they specify to use the A4 paper size and a 10 point font. The \usepackage command imports LaTeX style files; in this case the American Mathematical Society styles, which extend and improve the mathematical typesetting provided by LaTeX. Common packages you may want to include are listed below, with short descriptions.

amsmath provides additional mathematical typesetting

amssymb provides mathematical symbols

amsthm provides theorem formatting for AMS publications

fancyhdr provides customisable headers and footers

graphics provides graphics support

graphicx provides extended graphics support (use instead of **graphics**)

hyperref provides navigation support for PDF documents (clickable references, etc.)

url provides clickable URL support

Many of the packages listed require arguments and customisations to function correctly. Each one has online documentation which may be consulted.

The title, authors and date are also usually included in the preamble. This is achieved as follows:

```
\title{My Report}
\author{C. Turk \and J. Dorian}
\date{\today}
```

The contents of those fields can be anything you like — you could use the date field to insert a lecture number as well as a date, for example
`\date{\today -- Lecture 2}`. If the date field is omitted, the current date will be included automatically, but explicitly stating it makes it clearer!

Finally, the document itself is started with `\begin{document}` and ended with `\end{document}`. The title, author and date fields are typeset with the `\maketitle` command, and a table of contents can be included with the cunningly named `\tableofcontents` command. A complete LaTeX document template might look something like this:

```
\documentclass[12pt,a4paper]{article}
\usepackage{amsmath,amssymb,fancyhdr,url}
\usepackage[pdftex]{graphicx}

\title{My Report, Mk. II}
\author{J. Dorian \and B. Kelso}
\date{Before time began\ldots}

\begin{document}
\maketitle
\newpage
\tableofcontents
\newpage

\section{Life at Sacred Heart}
This is the story of \ldots

\end{document}
```

There are a few new things introduced here; sections will be covered later, but let us briefly mention the `\newpage` command, which does exactly what it says on the tin, and the `\ldots` command, which produces: . . .

3 Sections and Subsections

Sections are started with the `\section{section-title}` command. The section title goes between the {}. Similarly, subsections are started with the

`\subsection{sub-title}` command, and sub-subsections with the `\subsubsection` command. By default, all sections and nested subsections are automatically numbered. This behaviour can be turned off on a per-section basis by using the `\section*{section-title}` form, where the `*` tells LaTeX not to number these sections according to the document numbering scheme. The `*` can be used on (almost) anything which would normally be numbered; (sub)sections, equations, etc. Figures and tables use the `*` option in a different way because their naming and numbering is controlled through the `\caption` command.

4 Typesetting Considerations

LaTeX is a little more pedantic than most word-processors, and requires certain punctuation marks to be typeset in slightly different ways. This section discusses the most commonly encountered issues.

4.1 Single Quotes

LaTeX requires left and right quotes, rather than straight quotes. Using the apostrophe, ', always produces a right quote, so you need to use the *backtick*, ', to obtain a left-quote. Quoted text then looks something like *the 'flavour' of a particle* and not like *the 'flavour' of a particle*. The former produces the correct result when typeset, the latter begins and ends with end-quotes!

4.2 Double Quotes

Double left-quotes are input using two single left quotes (the backtick), while double right-quotes are input using two single right quotes (the apostrople). Double quoted text is typeset something like `\ldots said ''the project is on schedule and an investigation has been launched''`, which looks something like this when typeset: . . . said "the project is on schedule and an investigation has been launched"

4.3 An Aside on Quotation Marks

In British English, the usual practice is to enclose quoted matter between single quotation marks, reserving double quotation marks for a quotation within a quotation. In US English, the reverse is true. For quotations nested within the second quotation, one should revert to the original mark, alternating accordingly for subsequent levels.

4.4 Hyphenation and Dashes

Hyphens and dashes in printed text have three sizes; the soft hyphen, used to split a word across a line, is the shortest. This is automatically inserted by LaTeX when required. The hard hyphen is the same length as its soft sibling. It is used to hyphenate words within a line, and is typeset using a single dash (- ⇒ -). The en-rule (or en-dash) marks a range (often of numbers), such as 3–5. This is slightly longer (one *en* in length), and is typeset using two hyphens (-- ⇒ –). Finally, the em-rule (or em-dash) is used to insert a pause in text —

which should probably be replaced by a comma or semicolon anyway — and is the longest (one *em* in length). It is typeset using three hyphens (--- ⇒ —).

4.5   Special Characters

There are some characters which are used by LaTeX as control sequences. These must be escaped to be used in normal text. The characters most often encoutered are `\ _ & % { } #` (begin command, underscore, table separator, comment, left curly brace, right curly brace, hash or "pound" sign) and they can be produced in the text verbatim by 'escaping' with a leading backslash: `\_ \& \% \{ \} \#`, with the exception of the backslash, which can be inserted with the `\textbackslash` command: \.

4.6   Emphasising Text

To place emphasis on a certain phrase, use the `\emph{something}` command. This usually produces *italic* text, but the effect depends on the document style used. The advantage of using `\emph` is that emphasised text within an already emphasised section is typeset in Roman (non-italic) text. For example, `The \emph{President of  the \emph{United States} of America}` produces: The *President of the* United States *of America*.

In addition, it is possible to directly specify the typeface required. The command `\textit{italic text}` produces *italic text*, while `\textbf{bold text}` produces **bold text** and `\texttt{monospaced text}` produces `monospaced text`.