AMTH250 Lecture 1

Introduction to LaTeX

April 7, 2008

Contents

1 Introduction to LaTeX

LaTeX is a software package that is widely used in the scientific community to professionally typeset books and papers and also to create slide presentations.

The main purpose of including LaTeX into a unit of Scientific Computing is to enable you to submit your assignments electronically as LaTeX files.

LaTeX is available as the freeware package Miktex for Windows, Tetex for linux etc.

In order to create, view and print a LaTeX document you need

1. A text editor. Any plain text editor (notepad, vi, emacs but NOT MSWord) would do, but there exist more comfortable editors tailored to LaTeX such as "winedt" or "winshell" for Windows or "Kile", "emacs" for LaTeX.

2. A LaTeX package (such as Miktex). It includes a viewer for the document in .dvi format.

3. For viewing .pdf or .ps files additional software such as Acrobat reader and ghostview, ghostscript might be required.

The reference for these lectures is *The Not So Short Introduction to LaTeX 2ε* by Tobias Oetiker and others (see LaTeX Resources on the web page). In these lectures it will be referred to by the abbreviation NSSI.

1.1 How LaTeX Works

LaTeX is not a WYSIWYG ("What you see is what you get") editor. In fact it is not an editor at all, but a software package that transforms a source file of text and type-setting instructions into a viewable and printable document. LaTeX can choose the correct formatting of the text, the correct font size, margins etc. for you. It can automatically produce tables of contents, indices, numbering of formulae, sections, pages etc. But you have to declare what feature you want, e.g. when you want a new Section, a formula etc.

There are three steps in producing a document using LaTeX:

1. Create a LaTeX input file. This contains the text and formatting commands and must be a plain text (ASCII) file with the extension `.tex`.

2. Run the file through LaTeX or pdflatex. LaTeX produces a `.dvi` (device independent) file which may be viewed, for example, by `xdvi` on Linux systems or can be further transformed into a `.ps` (ps – postscript) or `.pdf` (pdf – portable document format) file. pdflatex produces a `.pdf` file and is the preferred way to do so.

3. If necessary you may convert the `.dvi` or `.pdf` file to another format.

The mechanics of doing all this will be covered in the first practical. The rest of this lecture will cover the structure of LaTeX input files.

1.2 LaTeX Input Files

1.2.1 The preamble

Each LaTeX source file starts with a preamble:

```
\documentclass[12pt]{article}
\usepackage{amsmath}
\usepackage{hyperref}
\usepackage{url}
```

Here we declare what kind of document we want to prepare. In our example we chose "article". Other options would be "book", "letter", "amsart" (for articles in the layout suggested by the American Mathematical Society), "beamer" (for beamer presentations). As you can see LaTeX commands and declarations usually start with a backslash \. Compulsory arguments are included in curly brackets {} and optional arguments are included in square brackets []. The optional argument `[12pt]` tells LaTeX to choose letter size 12 points. (One point equals 0.3514598mm).

The internet has lots of documentations and templates for different classes of documents. Especially for beginners it would be advisable to use a template and to adapt it to your needs. You may take a look at the source files for these lecture notes, which are also produced with LaTeX.

The `\documentclass{}` declaration is usually followed by instructions to load additional packages. Here we load the "amsmath" package that makes additional symbols and fonts for mathematical texts available. Other packages provide features related to different languages, to graphics, to chemistry and much more. The package "hyperref" enables LaTeX to produce clickable cross references, e.g. from a table of contents entry to the actual section, or from a formula reference to the actual formula. The package "url" makes the command `\url{http://www.xxx}` available which formats the url `http://www.une.edu.au` nicely and, in conjunction with "hyperref", produces clickable web-links in the document.

The preamble is also the place where you declare the title and the author of the document:

```
\title{My Latex document}
\author{My name}
```

The command `\maketitle` tells LaTeX to produce a cover sheet and to put title, name and "today's" date at the "right" place. If you want another date, you need to declare this in the preamble by

```
\date{My birthday}
```

To suppress the date declare

```
\date{}
```

The command `\today` can be used to print today's date (e.g. in a letter).

1.3 The text body

The actual text follows the `\maketitle` command between the commands `\begin{document}` and `\end{document}`:

```
\begin{document}
\maketitle
 Your text
\end{document}
```

Any text in the source file after `\end{document}` will be ignored by LaTeX.

1.3.1 Spaces

1. One or more *whitespace* characters such as spaces, tabs or linebreaks are treated as single space.
2. One or more blank lines start a new paragraph.

Example:

Note: Examples in these notes will usually take the form of LaTeX input in typewriter text, followed by the result in slightly smaller type.

```
                        This is
            a silly
                      way                    to type
      a                                    sentence.


           Followed     by                        a
new                            paragraph.
```

This is a silly way to type a sentence.
Followed by a new paragraph.

1.3.2 Special Characters

Some characters have a special meaning related to LaTeX commands and declarations and will not print as expected. The most important of these are:

```
#   $   %   &   {   }   \
```

Except for the backslash itself, these can be printed by preceding them with a backslash. The `$\backslash$` command is used to print a backslash.

Example:

```
\#   \$   \%   \&   \{   \}   $\backslash$
```

$ % & { } \

1.3.3 Comments

A `%` is used for comments. When LaTeX encounters a `%` the rest of the line is ignored.

Example:

```
The rest of this line will be ignored, % THIS IS A COMMENT
% and another comment
as will the line above.
```

The rest of this line will be ignored, as will the line above.

1.3.4 LaTeX Commands

LaTeX commands begin with a backslash `\`. A big part of learning LaTeX consists of understanding its commands and their effects.

Example:

```
\LaTeX{} is the topic of this lecture.
```

LaTeX is the topic of this lecture.

Remark. `\LaTeX` is the command that produces the fancy LaTeX logo. Usually curly brackets at the end of a command are used to enter compulsory arguments. Here the function of the brackets is to tell LaTeX where the command ends. The same effect could be achieved by a blank. But since LaTeX does not distinguish between one or more blanks the following blanks would be swallowed. (Try it.)

Example:

The following is a small but complete input file for a LaTeX document:

```
\documentclass{article}
\begin{document}
  This is a very short article.
\end{document}
```

2 Formatting Text

2.1 Special Symbols

2.1.1 Quotation Marks

1. For quotation marks use `‘‘` for opening quotes and `’’` for closing quotes.
2. For single quotes use one of each.

Example:

```
Do you mean ‘‘eye’’ or ‘i’?
```

Do you mean “eye” or ‘i’?

2.1.2 Dashes and Hyphens

There are three types of dashes in LaTeX.

Example:

```
1 - short-dashes and hyphens

2 -- long--dashes

3 --- longer---dashes
```

1 - short-dashes and hyphens
2 – long–dashes
3 — longer—dashes

2.2 Font Selection

2.2.1 Font Types

The font types generally available in LaTeX are:

1. `\textrm{...}` roman
2. `\texttt{...}` `typewriter`
3. `\textsl{...}` *slanted*
4. `\textsf{...}` sans serif
5. `\textbf{...}` **bold face**
6. `\textit{...}` *italic*
7. `\textsc{...}` SMALL CAPITALS
8. `\emph{...}` *emphasized*

Example:

```
This \textit{sentence} \texttt{uses} a \textsl{number} of
\textsf{different} \textbf{fonts} \textit{which}
\textsc{makes} it \textbf{hard} to \textsc{read}. \\
\emph{Emphasized text} differs from \textit{italic text}
in that \textsf{it can be \emph{combined} with other font
changes.}
```

This *sentence* uses a *number* of different **fonts** *which* MAKES it **hard** to READ. *Emphasized text* differs from *italic text* in that it can be *combined* with other font changes.

2.2.2 Font Sizes

The font size, either 10pt (the default), 11pt or 12pt, for the whole document is set within the initial `\documentclass` command, e.g.

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

The font size and type of title and section headings are chosen automatically by LaTeX.

The font sizes generally available in LaTeX are:

1. `{\tiny ...}` tiny
2. `{\scriptsize ...}` very small
3. `{\footnotesize ...}` quite small
4. `{\small ...}` small
5. `{\normalsize ...}` normal

6. `{\large ...}` larger
7. `{\Large ...}` larger still
8. `{\LARGE ...}` quite large
9. `{\huge ...}` very large
10. `{\Huge ...}` huge

Example:

```
When combining changes of font {\Large \textbf{size and
type}}, remember that the \textbf{size}
change comes first.
```

When combining changes of font **size and type**, remember that the **size** change comes first.

2.3 Spacing and Indentation

2.3.1 Paragraphs and Indentation

We have already seen that in LaTeX a blank line starts a new paragraph. By default LaTeX indents each paragraph except the first paragraph of a Chapter, Section etc. This can be controlled using the commands `\indent` and `\noindent`.

Example:

```
Normally paragraphs are indented.

\noindent But this one isn't.
```

Normally paragraphs are indented.
But this one isn't.

2.3.2 Line and Page Breaks

1. The commands `\\` or `\newline` force a new line to be started without starting a new paragraph.
2. The command `\newpage` can be used to force a new page to be started.

Example:

```
This is how to start a new line \\
without starting a new paragraph.

Of course, a new paragraph is started by a blank line.
```

This is how to start a new line
without starting a new paragraph.

Of course, a new paragraph is started by a blank line.

2.3.3 Spacing Between Paragraphs

By default LaTeX adds no extra space between paragraphs. Sometimes, to make certain paragraphs stand out, you need to add extra space. This can be done with the `\smallskip`, `\medskip` and `\bigskip` commands.

Example:

```
Here is an example of \ldots

different spacings \ldots

\smallskip

between paragraphs.

\medskip

This is useful in highlighting certain paragraphs.

\bigskip

It is also useful with equations, tables and diagrams.
```

Here is an example of ...
different spacings ...

between paragraphs.

This is useful in highlighting certain paragraphs.

It is also useful with equations, tables and diagrams.

2.4 Sections and Subsections

The sectioning commands

```
\section{...}
\subsection{...}
\subsubsection{...}
```

are available in the `article` document class. The additional command `\chapter` is available in the `report` and `book` document classes.

The numbering of sections is done automatically by LaTeX, as is the font selection for titles and spacing between sections.

The

```
\subsubsection*{...}
```

command does not print the subsubsection number.

Example

```
\subsubsection*{Example}

This is how examples are introduced in these notes.
```

Example

This is how examples are introduced in these notes.

2.5 Titles and Tables of Contents

The following example gives the first few lines of *this* document[1]:

Example:

```
\documentclass[11pt,a4paper]{article}

\title{AMTH142 \hfill Lecture 2\\[6mm]
       \LaTeX{} -- Formatting Text}
\author{}

\begin{document}

\maketitle

\tableofcontents
```

[1] You usually need to run a document through LaTeX twice to get the table of contents correct. The reason for this is that LaTeX collects the structure data during the first run and writes them into an auxiliary file (with extension `.aux`). In the second run the data are used to create the table of contents (writen to file with extension `.toc`).

```
\newpage

\section{Formatting Text}

\subsection{Special Symbols}

\subsubsection{Quotation Marks}
```

2.6 Environments

These are generally associated with a pair of matching commands

```
\begin{XXX}
\end{XXX}
```

(This is similar to the `\begin{document}` ... `\end{document}` commands.)

2.6.1 Lists

LaTeX has three types of list environments:

1. `enumerate`
2. `itemize`
3. `description`

The individual items in the list are introduced by the `\item` command. List can be nested, that is you can have lists within lists.

Example:

```
\begin{enumerate}
  \item The \texttt{enumerate} environment numbers the
    elements in the list.
  \item The \texttt{itemize} environment precedes each
    item by a large dot as follows:
    \begin{itemize}
      \item This is the first item of an \texttt{itemize}
        environment.
      \item And this is the second.
    \end{itemize}
  \item This is an example of the \texttt{description}
    environment.
    \begin{description}
```

```
      \item[First] item in the list.
      \item[Second] item in the list.
    \end{description}
\end{enumerate}
```

1. The `enumerate` environment numbers the elements in the list.
2. The `itemize` environment precedes each item by a large dot as follows:
   - This is the first item of an `itemize` environment.
   - And this is the second.
3. This is an example of the `description` environment.

   First item in the list.

   Second item in the list.

2.6.2 Centering Text

Example:

```
\begin{center}
   This is an example \\ of centered \\ text. \\
   Centering is useful when including tables and diagrams.
\end{center}
```

This is an example
of centered
text.
Centering is useful when including tables and diagrams.

2.6.3 Verbatim

Text enclosed between a `\begin{verbatim}` and `\end{verbatim}` pair is printed exactly as is in typewriter font, including spaces and linebreaks, and with LaTeX commands ignored.

Example:

```
\begin{verbatim}
  \LaTeX{} commands are ignored in verbatim environments,
  but spaces   and
  linebreaks are faithfully followed.
\end{verbatim}
```

```
  \LaTeX{} commands are ignored in verbatim environments,
  but spaces   and
  linebreaks are faithfully followed.
```

The `verbatim` environment is used for the examples in these notes. The same effect within paragraphs can be obtained with the `\verb` command. The character immediately following the `\verb` is the delimiting character; the following text will be printed verbatim until this delimiting character is reached again.

Example:

```
An important difference between \verb+\verb+ and
\verb+\texttt+ is that \LaTeX{} commands have their intended
effect inside \verb+\texttt+, while inside \verb+\verb+ they
are printed verbatim.
In this example I have used \texttt{+} as the delimiter.
```

An important difference between `\verb` and `\texttt` is that LaTeX commands have their intended effect inside `\texttt`, while inside `\verb` they are printed verbatim. In this example I have used `+` as the delimiter.