1 LaTeX Concepts

1.1 Environments and commands

LaTeX is a macro-package for TeX which has many preset *environments* where much of the setting up that TeX users have to do explicitly is done for you. An environment has the form

`\begin{`*environment name*`}`

.

`\end{`*environment name*`}`

LaTeX also has commands which affect the formatting of the document. Their arguments are given in braces. For example,

```
\textit{This is much more important} than this.
```

produces as output *This is much more important* than this.

ports and OHP slides are useful.

- See the comp.text.tex newsgroup.
- Use the CTAN Archive ftp.tex.ac.uk
- Engineering Department users can see the examples in `/export/Examples/LaTeX`. The `Tutorial` subdirectory contains exercises.
- Engineering Department users can borrow a LaTeX manual from the machine room.
- Look at the LaTeX files in your system. Ours are in `/opt/latex/inputs`.

- improper use of special characters
- forgetting to have the appropriate `\usepackage{...}` lines.
- a error like this

```
Overfull \hbox (15.42563pt too wide) in paragraph at lines
285--288 \OT1/ppl/m/n/10 You can cre-ate ver-tical space
between lines or ho-ri-zontal space   between
```

  means that lines 285 to 288 are producing a text line about 15 pts (about 5mm) too wide. `\OT1/ppl/m/n/10` is the specification of the font used. When TeX does right and left alignment, it works out how much space it needs to leave between words and where to hypenate words if necessary. But the amount of space it's prepared to leave has to fall within a certain range and it will only split words in certain places (shown in the error message by a hyphen). If these restrictions mean that LaTeX can't produce a satisfactory line, it will produce as much of the line as it can. A re-phrasing of the offending sentence will usually solve the problem. Another thing you might try is to control the way a troublesome word is hyphenated using something like

```
\hypenation{furthermore fur-ther-more}
```

  at the top of your file. If all else fails, use `{\sloppy...}` to enclose the offending text.

- LaTeX issues a `*`.

  This means LaTeX needs more input. It probably means you've missed out an `\end{document}`, but if not you may be able to get LaTeX to continue processing as best it can by typing `<Return>`.

- `! Argument of \label  has an extra }`

  If you're sure you haven't left out a left brace, then maybe you need to *protect* the inmost (fragile) command. For example, `\label` is fragile which causes a problem in `\caption{Picture\label{margin}}`, so the safer construction `\caption{Picture\protect\label{margin}}` has to be used.

1.4 Files created

More than just the `.dvi` file may be produced. Don't worry about them – except perhaps for the log file you won't need to look at them.

- `.aux` cross references, etc
- `.toc` created by `\tableofcontents`
- `.lof` created by `\listoffigures`
- `.lot` created by `\listoftables`
- `.log` a copy of the diagnostic output that usually comes out on the screen.

Some common errors can be found using the `lacheck` program on the LaTeX file.


```
{\baselineskip}%
{\baselineskip}{\normalfont\normalsize\scshape\centering}}%
\makeatother
\begin{document}
```

5.3 New Commands

A completely new command can be created using

`\newcommand{\`*commandname*`[`*number of arguments*`]{` *command text, using* `#1`, `#2` *etc to denote arguments*`}`

For example,

```
\newcommand{\ve}[1]{\(#1_1 ... #1_n\)}
\ve{x}
```

produces as output $x_1 ... x_n$

A problem with this example is that it should change to math mode if LaTeX is already in that mode. A better try would be

```
\newcommand{\ve}[1]{\ensuremath{#1_1 ... #1_n}}
```

which will only change to math mode if it's necessary.

A new environment is just as easily created - give the name of the environment, and what you want to happen on entering and leaving the environment. The following provides a variant of the itemize command.

```
\newenvironment{emlist}{\begin{itemize} \em}{\end{itemize}}
\begin{emlist}
\item first comment
\item second comment
\end{emlist}
The end of the environment ends the scope of the emphasis.
```

- *first comment*
- *second comment*

The end of the environment ends the scope of the emphasis.

5.4 Packages

There are many features and options not mentioned in this handout. See the packages[12] section of the online LaTeX page for more details. You can often find out what you want by looking at the files in `/opt/latex/inputs`. If your file begins

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

then various macro files are read when you run `latex`, namely (`article.cls` and `size12.sty`). Some of these files are well-enough commented to be useful documentation.

[12] http://www-h.eng.cam.ac.uk/help/tpl/textprocessing/LaTeX_intro.html#Packages

parskip:- determines the gap between paragraphs.

baselineskip:- determines the usual distance from the bottom of one line to the bottom of the next. You can adjust this to produce double spacing, but a better way, which takes a stretch *factor* as argument is to use `\linespread`. For instance *before* `\begin{document}` you can do

```
\linespread{1.6}
```

to get doublespacing through your document. The `setspace` package offers more options.

2.2 Document and page organisation

- **Document classes**: The standard classes are `article`, `book`, `letter`, `report`, `slides`. The differences between these are often minor. A `book` can have chapters. In a `report` sections begin at `0` whereas in an `article` they begin at `1`. Just about all the handouts are `article`s.
- **Big Documents** :- It's best to split your document into smaller files and have a master file looking like this

```
\documentclass[dvips,12pt]{book}
\usepackage{a4,color,graphics,palatino,fancyhdr}
\begin{document}
\pagestyle{empty}
\tableofcontents
\listoffigures
\pagestyle{fancy}
\input{chapter1}
\input{chapter2}
\input{fig2}
\input{chapter3}
\input{chapter4}
\appendix
\input{appendices}
\end{document}
```

on which you can run `latex` just as if the master file contained all the text of `chapter1.tex` etc. The advantage of this is that once you have a chapter correct, you can comment out the corresponding 'input' line and avoid unnecessary processing. Remember to take out the `\begin{document}` and `\end{document}` lines from the component files.

- **Page Size** :- You can choose the margin sizes yourself by changing the following dimensions *before* the `\begin{document}` line.

```
\setlength{\topmargin}{-0.4in}
\setlength{\topskip}{0.3in}    % between header and text
\setlength{\textheight}{9.5in} % height of main text
\setlength{\textwidth}{6in}    % width of text
\setlength{\oddsidemargin}{0.75in} % odd page left margin
\setlength{\evensidemargin}{0.75in} % even page left margin
```

Note that a margin width of 0cm gives you a margin 4cm wide. Rather than set absolute sizes you can modify the default sizes using commands like the following –

Figure 1: 1 cm of space

Putting `!` as the first argument in the square brackets will encourage LaTeX to do what you say, even if the result's sub-optimal.

If you have a label defined in the caption, LaTeX is likely to give an error message. `\label` is a *fragile* command (see the LaTeX book for details) so you'll need to do something like

```
...
\caption{1 cm of space\protect\label{EMPTY}}
```

or simply put the `\label` command after the caption. Note that if you put the `\label` *before* the caption, the resulting reference will be the section number and not the figure number.

4.8 Tabbing

Within this environment tabs can be set by `\=` and the next tab moved to by using `\>`.

```
\begin{tabbing}
if \= it's raining\\      % set tab here, after the 'if'
   \> get an umbrella \=\\ % go to the defined tab and set a new one
else\\
   \> get wet \> * next tab is here\\
endif
\end{tabbing}
```

if it's raining
  get an umbrella
else
  get wet   * next tab is here
endif

4.9 Verbatim

Within this environment things come out unformatted. It's useful for showing examples of typed input and provides a way of printing characters that have a special meaning for LaTeX.

`\begin{verbatim} caret is ^, tilde is ~ and backslash is \ \end{verbatim}`

produces

```
caret is ^, tilde is ~ and backslash is \
```

If you just want to quote a few characters, use `\verb|`*quoted text*`|`. The characters delimiting the quote can be anything as long as they are the same.

A title will only be numbered if its 'depth' isn't more than `secnumdepth` and will only appear in the contents page if the 'depth' isn't more than the value of `tocdepth`. So, for example, doing

```
\setcounter{tocdepth}{2}
\setcounter{secnumdepth}{3}
```

will cause section `1.4.3` to be numbered, but it won't appear in the contents.

2.3 Pagebreaks, space, footnotes, references, boxes, etc

- **Page Breaks** :- you can force a page break using `\newpage`.
- **Preventing line breaks** :- If there's a word that you don't want broken, put it in an `mbox`. *E.g.*,

```
One shouldn't try to break up \mbox\emph{relationships}
```

- **Space** :-

```
You can create vertical space \vspace{.5cm} between lines
or horizontal space \hspace{1.5cm} between words.
The arguments to these commands can be negative.
```

You can create vertical space between lines or horizontal space between words. The arguments to these commands can be negative. `\vspace*` will create space even at the top of a page. It's sometimes useful to create stretchable space. The following creates space that pushes the letters to the edge of the page

```
\noindent A\hspace{\stretch{1}} B\\
C\hspace{\stretch{1}} D
```

A B
C D

- **Footnotes** :-

```
Do them like this\footnote{I told you so.}
```

Do them like this[7] The footnotes are numbered by default. If you want to use symbols (stars, daggers etc) then you need to redefine how the `footnote` counter is displayed.

```
\def\thefootnote{\fnsymbol{footnote}}
```

- **Margin notes** :-

```
Do them like this\marginpar{margin note}
```

Do them like this margin note

[7] I told you so.

The `rotating`[9] package is useful if you have a wide table that you want to display in landscape mode. You need to put your table inside `\begin{sidewaystable}` and `\end{sidewaystable}`.

If you want the table to have a caption and *float* (float up the page if it's started right near the foot of a page, for example), use

```
\begin{table}[htbp]
\begin{tabular}...
...
\end{tabular}
\caption{...}
\end{table}
```

See section 4.7 for details.

4.4 Array

The `array` environment is similar to the `tabular` but must be within a *math* environment. This

```
\begin{math}
\left(
\begin{array}{clrr}
      a+b+c & uv & x-y & 27 \\
       x+y  & w  & +z  & 363
\end{array}
\right)
\end{math}
```

produces $\left(\begin{array}{clrr} a+b+c & uv & x-y & 27 \\ x+y & w & +z & 363 \end{array}\right)$

4.5 Pictures

LaTeX has some graphics capabilities. It's much better to import an encapsulated postscript file. See the *LaTeX Maths and Graphics*[10] document for more details.

```
\newcounter{cms}
\setlength{\unitlength}{1mm}
\begin{picture}(50,39)
\put(0,7){\makebox(0,0)[bl]{cm}}
\multiput(10,7)(10,0){5}{\addtocounter
   {cms}{1}\makebox(0,0)[b]{\arabic{cms}}}
\put(15,20){\circle{6}}
\put(30,20){\circle{6}}
\put(15,20){\circle*{2}}
\put(30,20){\circle*{2}}
\put(10,24){\framebox(25,8){a box}}
\put(10,32){\vector(-2,1){10}}
\multiput(1,0)(1,0){49}{\line(0,1){2,5}}
```

[9] http://www-h.eng.cam.ac.uk/help/tpl/textprocessing/rotating.ps
[10] http://www-h.eng.cam.ac.uk/help/tpl/textprocessing/latex_maths+pix/latex_maths+pix.html

3.2 Special characters

† is created by `\dag`, ‡ by `\ddag`, § by `\S`, ¶ by `\P`, £ by `\pounds`, ö by `\"{o}`, ©by `\copyright`. Many others are available in the `math` environment, including all the lower case greek letters and most of the upper case ones. If you only want to use a few characters you can bracket the symbols using `$` and `$` rather than `\begin{math}` and `\end{math}`. You can put a slash through any of these characters by prefacing them with `\not`

$\sqrt{i}$	`\sqrt{i}`	$\sqrt[5]{x+iy}$	`\sqrt[5]{x+iy}`	$\ldots$	`\ldots`
$\cdots$	`\cdots`	$\vdots$	`\vdots`	$\ddots$	`\ddots`
α	`\alpha`	β	`\beta`	γ	`\gamma`
δ	`\delta`	ω	`\omega`	Γ	`\Gamma`
Θ	`\Theta`	Ω	`\Omega`	$\pm$	`\pm`
$\mp$	`\mp`	$\times$	`\times`	$\div$	`\div`
$\ast$	`\ast`	$\star$	`\star`	$\circ$	`\circ`
$\bullet$	`\bullet`	$\cdot$	`\cdot`	$\cap$	`\cap`
$\bigcap$	`\bigcap`	$\cup$	`\cup`	$\bigcup$	`\bigcup`
$\uplus$	`\uplus`	$\biguplus$	`\biguplus`	$\sqcap$	`\sqcap`
$\sqcup$	`\sqcup`	$\bigsqcup$	`\bigsqcup`	$\vee$	`\vee`
$\bigvee$	`\bigvee`	$\wedge$	`\wedge`	$\bigwedge$	`\bigwedge`
$\setminus$	`\setminus`	$\wr$	`\wr`	$\diamond$	`\diamond`
$\bigtriangleup$	`\bigtriangleup`	$\bigtriangledown$	`\bigtriangledown`	$\triangleleft$	`\triangleleft`
$\triangleright$	`\triangleright`	$\oplus$	`\oplus`		
$\bigoplus$	`\bigoplus`	$\ominus$	`\ominus`	$\otimes$	`\otimes`
$\bigotimes$	`\bigotimes`	$\oslash$	`\oslash`	$\odot$	`\odot`
$\bigodot$	`\bigodot`	$\bigcirc$	`\bigcirc`	$\amalg$	`\amalg`
$\leq$	`\leq`	$\prec$	`\prec`	$\preceq$	`\preceq`
$\ll$	`\ll`	$\subset$	`\subset`	$\subseteq$	`\subseteq`
$\in$	`\in`	$\vdash$	`\vdash`		
$\geq$	`\geq`	$\succ$	`\succ`	$\succeq$	`\succeq`
$\gg$	`\gg`	$\supset$	`\supset`	$\supseteq$	`\supseteq`
$\sqsupseteq$	`\sqsupseteq`	$\ni$	`\ni`	$\dashv$	`\dashv`
$\equiv$	`\equiv`	$\sim$	`\sim`	$\simeq$	`\simeq`
$\asymp$	`\asymp`	$\approx$	`\approx`	$\cong$	`\cong`
$\neq$	`\neq`	$\doteq$	`\doteq`	$\propto$	`\propto`
$\models$	`\models`	$\perp$	`\perp`	$\mid$	`\mid`
$\parallel$	`\parallel`	$\bowtie$	`\bowtie`	$\smile$	`\smile`
$\frown$	`\frown`	$\leftarrow$	`\leftarrow`	$\Leftarrow$	`\Leftarrow`
$\rightarrow$	`\rightarrow`	$\Rightarrow$	`\Rightarrow`	$\leftrightarrow$	`\leftrightarrow`
$\Leftrightarrow$	`\Leftrightarrow`	$\mapsto$	`\mapsto`	$\hookleftarrow$	`\hookleftarrow`
$\leftharpoonup$	`\leftharpoonup`	$\leftharpoondown$	`\leftharpoondown`	$\rightleftharpoons$	`\rightleftharpoons`
$\longleftarrow$	`\longleftarrow`	$\Longleftarrow$	`\Longleftarrow`	$\longrightarrow$	`\longrightarrow`
$\Longrightarrow$	`\Longrightarrow`	$\longleftrightarrow$	`\longleftrightarrow`	$\Longleftrightarrow$	`\Longleftrightarrow`
$\longmapsto$	`\longmapsto`	$\hookrightarrow$	`\hookrightarrow`	$\rightharpoonup$	`\rightharpoonup`
$\rightharpoondown$	`\rightharpoondown`	$\uparrow$	`\uparrow`	$\Uparrow$	`\Uparrow`
$\downarrow$	`\downarrow`	$\Downarrow$	`\Downarrow`	$\updownarrow$	`\updownarrow`
$\nearrow$	`\nearrow`	$\searrow$	`\searrow`	$\swarrow$	`\swarrow`
$\nwarrow$	`\nwarrow`	$\aleph$	`\aleph`	$\hbar$	`\hbar`
$\imath$	`\imath`	$\jmath$	`\jmath`	ℓ	`\ell`
$\wp$	`\wp`	$\Re$	`\Re`	$\Im$	`\Im`
$\prime$	`\prime`		`\empty`	∇	`\nabla`
$\surd$	`\surd`	$\top$	`\top`	$\bot$	`\bot`
$\Vert$	`|`	$\angle$	`\angle`	$\forall$	`\forall`
$\exists$	`\exists`	$\neg$	`\neg`	$\flat$	`\flat`
$\natural$	`\natural`	$\sharp$	`\sharp`	$\backslash$	`\backslash`
∂	`\partial`	∞	`\infty`	$\triangle$	`\triangle`


```
\item numbered, using \texttt{enumerate}

\begin{enumerate}
\item one
\item two
\item three
\end{enumerate}

\item or with a label, using \texttt{description}

\begin{description}
\item[itemize] bullets
\item[enumerate] automatic numbering
\item[description] labelling
\end{description}

\end{itemize}
```

- just by a bullet, using `itemize`
- numbered, using `enumerate`
  1. one
  2. two
  3. three
- or with a label, using `description`

  itemize bullets
  enumerate automatic numbering
  description labelling

The `pifont` package includes a variant of the `itemize` command that will replace the usual 'bullet' by a Zapf Dingbat symbol of your choice

```
\begin{dinglist}{43}
\item First
\item Second
\end{dinglist}
```

☞ First

☞ Second

and a variant of the `enumerate` command that given an initial Zapf Dingbat symbol will increment the symbol for each item.

```
\begin{dingautolist}{172}
\item First
\item Second
\end{dingautolist}
```

① First

② Second

3.5 Postscript Fonts

It is easy to write a document that has postscript fonts. We have package support for helvetica (`helvetic`), utopia, times, optima, newcentury (`newcentu`), palatino and courier. To use palatino, for instance, all you need to do is add

```
\usepackage{palatino}
```

to your file. The `pifont` package has special commands for using the Zapf Dingbats font. `\dingfill{40}` completes the line with the specified symbol ✍ ✍

and `\dingline{36}` draws a whole line of symbols.

✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂ ✂

It's a good idea to use a font that's installed in the printer you intend to use. See the table on page 12 for information about the teaching system's `ljmr1` printer.

3.6 Font attributes

The commands above should give you sufficient control over fonts. If you don't want to know more at the moment then turn to section 4

Every text font in LaTeX has five *attributes*:

encoding This specifies the order that characters appear in the font. The most common values for the font encoding is `OT1`.

family The name for a collection of fonts, usually grouped under a common name by the font foundry. For example, 'Adobe Times' and Knuth's 'Computer Modern Roman' are font families. There are far too many font families to list them all, but some common ones are:

Internal fontname	*Fontname*	*In ljmr1?*
`cmr`	Computer Modern Roman	No
`cmss`	Computer Modern Sans	No
`cmtt`	Computer Modern Typewriter	No
`cmm`	Computer Modern Math Italic	No
`cmsy`	Computer Modern Math Symbols	No
`cmex`	Computer Modern Math Extensions	No
`ptm`	Adobe Times	Yes
`phv`	Adobe Helvetica	Yes
`pcr`	Adobe Courier	Yes
`pun`	Univers	No
`ppl`	Palatino	Yes
`pagk`	AvantGarde-Book	Yes
`pagd`	AvantGarde-Demi	Yes
`pbk`	Bookman	Yes
`put`	Utopia	No
`pop`	Optima	No
`pnc`	New Century Schoolbook	Yes
`pzd`	ZapfDingbats	Yes
`rpad`	Garamond	No

series How heavy or expanded a font is. For example, 'medium weight', 'narrow' and 'bold extended' are all series. The most common values for the font series are:

`m`	Medium
`b`	Bold
`bx`	Bold extended
`sb`	Semi-bold
`c`	Condensed

shape The form of the letters within a font family. For example, 'italic', 'oblique' and 'upright' are all font shapes. The most common values for the font shape are:

`n`	Normal (that is 'upright' or 'roman')
`it`	Italic
`sl`	Slanted (or 'oblique')
`sc`	Caps and small caps

size The design size of the font, for example '10pt'.

These five parameters specify every LaTeX font, for example:

	LaTeX specification				*Font*
OT1	cmr	m	n	10pt	Computer Modern Roman 10pt
OT1	cmss	m	sl	12pt	Computer Modern Sans Oblique 12pt
OML	cmm	m	it	10pt	Computer Modern Math Italic 10pt
T1	ptm	b	it	18pt	Adobe Times Bold Italic 18pt

3.7 Selection commands

There are commands to set attributes one at a time:

Command	*Attribute*	*Value in* article *class*, 10pt
`\textrm{..}` or `\rmfamily`	family	`cmr`
`\textsf{..}` or `\sffamily`	family	`cmss`
`\texttt{..}` or `\ttfamily`	family	`cmtt`
`\textmd{..}` or `\mdseries`	series	`m`
`\textbf{..}` or `\bfseries`	series	`bx`
`\textup{..}` or `\upshape`	shape	`n`
`\textit{..}` or `\itshape`	shape	`it`
`\textsl{..}` or `\slshape`	shape	`sl`
`\textsc{..}` or `\scshape`	shape	`sc`
`\tiny`	size	`5pt`
`\scriptsize`	size	`7pt`
`\footnotesize`	size	`8pt`
`\small`	size	`9pt`
`\normalsize`	size	`10pt`
`\large`	size	`12pt`
`\Large`	size	`14.4pt`
`\LARGE`	size	`17.28pt`
`\huge`	size	`20.74pt`
`\Huge`	size	`24.88pt`

The low-level commands used to change font attributes are as follows.

`\fontencoding{`*encoding*`}`
`\fontfamily{`*family*`}`
`\fontseries{`*series*`}`
`\fontshape{`*shape*`}`
`\fontsize{`*size*`}{`*baselineskip*`}`