Getting Started with LaTeX

David R. Wilkins

2nd Edition

Contents

1 Introduction to LaTeX **2**
- 1.1 What is LaTeX? 2
- 1.2 A Typical LaTeX Input File 2
- 1.3 Characters and Control Sequences 4

2 Producing Simple Documents using LaTeX **5**
- 2.1 Producing a LaTeX Input File 5
- 2.2 Producing Ordinary Text using LaTeX 6
- 2.3 Blank Spaces and Carriage Returns in the Input File 8
- 2.4 Quotation Marks and Dashes 9
- 2.5 Section Headings in LaTeX 10
- 2.6 Changing Fonts in Text Mode 11
- 2.7 Accents used in Text 13
- 2.8 Active Characters and Special Symbols in Text 14

3 Producing Mathematical Formulae using LaTeX **15**
- 3.1 Mathematics Mode 15
- 3.2 Characters in Mathematics Mode 17
- 3.3 Superscripts and Subscripts 17
- 3.4 Greek Letters 18
- 3.5 Mathematical Symbols 19
- 3.6 Changing Fonts in Mathematics Mode 22
- 3.7 Standard Functions (sin, cos etc.) 23
- 3.8 Text Embedded in Displayed Equations 24
- 3.9 Fractions and Roots 24
- 3.10 Ellipsis (i.e., 'three dots') 25
- 3.11 Accents in Mathematics Mode 26

3.12 Brackets and Norms 26
3.13 Multiline Formulae in LaTeX 28
3.14 Matrices and other arrays in LaTeX 29
3.15 Derivatives, Limits, Sums and Integrals 31

4 Further Features of LaTeX **35**
4.1 Producing White Space in LaTeX 35
4.2 Lists 37
4.3 Displayed Quotations 40
4.4 Tables 41
4.5 The Preamble of the LaTeX Input file 44
4.6 Defining your own Control Sequences in LaTeX 46

1 Introduction to LaTeX

1.1 What is LaTeX?

LaTeX is a computer program for typesetting documents. It takes a computer file, prepared according to the rules of LaTeX and converts it to a form that may be printed on a high-quality printer, such as a laser writer, to produce a printed document of a quality comparable with good quality books and journals. Simple documents, which do not contain mathematical formulae or tables may be produced very easily: effectively all one has to do is to type the text straight in (though observing certain rules relating to quotation marks and punctuation dashes). Typesetting mathematics is somewhat more complicated, but even here LaTeX is comparatively straightforward to use when one considers the complexity of some of the formulae that it has to produce and the large number of mathematical symbols which it has to produce.

LaTeX is one of a number of 'dialects' of TeX, all based on the version of TeX created by D. E. Knuth which is known as Plain TeX. LaTeX (created by L. B. Lamport) is one of these 'dialects'. It is particularly suited to the production of long articles and books, since it has facilities for the automatic numbering of chapters, sections, theorems, equations etc., and also has facilities for cross-referencing. It is probably one of the most suitable version of LaTeX for beginners to use.

1.2 A Typical LaTeX Input File

In order to produce a document using LaTeX, we must first create a suitable *input file* on the computer. We apply the LaTeX program to the input file and then use the printer to print out the so-called 'DVI' file produced by the LaTeX program (after first using another program to translate the 'DVI' file into a form that the printer can understand). Here is an example of a typical LaTeX input file:

```
\documentclass[a4paper,12pt]{article}
\begin{document}

The foundations of the rigorous study of \textit{analysis}
were laid in the nineteenth century, notably by the
mathematicians Cauchy and Weierstrass. Central to the
study of this subject are the formal definitions of
\textit{limits} and \textit{continuity}.

Let $D$ be a subset of $\bf R$ and let
$f \colon D \to \textbf{R}$ be a real-valued function on
$D$. The function $f$ is said to be \textit{continuous} on
$D$ if, for all $\epsilon > 0$ and for all $x \in D$,
there exists some $\delta > 0$ (which may depend on $x$)
such that if $y \in D$ satisfies
\[ |y - x| < \delta \]
then
\[ |f(y) - f(x)| < \epsilon. \]

One may readily verify that if $f$ and $g$ are continuous
functions on $D$ then the functions $f+g$, $f-g$ and
$f.g$ are continuous. If in addition $g$ is everywhere
non-zero then $f/g$ is continuous.

\end{document}
```

When we apply LaTeX to these paragraphs we produce the text

> The foundations of the rigorous study of *analysis* were laid in the nineteenth century, notably by the mathematicians Cauchy and Weierstrass. Central to the study of this subject are the formal definitions of *limits* and *continuity*.
>
> Let D be a subset of $\mathbf{R}$ and let $f \colon D \to \mathbf{R}$ be a real-valued function on D. The function f is said to be *continuous* on D if, for all $\epsilon > 0$ and for all $x \in D$, there exists some $\delta > 0$ (which may depend on x) such that if $y \in D$ satisfies
>
> $$|y - x| < \delta$$
>
> then
>
> $$|f(y) - f(x)| < \epsilon.$$

One may readily verify that if f and g are continuous functions on D then the functions $f + g$, $f - g$ and $f.g$ are continuous. If in addition g is everywhere non-zero then f/g is continuous.

This example illustrates various features of LaTeX. Note that the lines

```
\documentclass[a4paper,12pt]{article}
\begin{document}
```

are placed at the beginning of the input file. These are followed by the main body of the text, followed by the concluding line

```
\end{document}
```

Note also that, although most characters occurring in this file have their usual meaning, yet there are special characters such as `\`, `$`, `{` and `}` which have special meanings within LaTeX. Note in particular that there are sequences of characters which begin with a 'backslash' `\` which are used to produce mathematical symbols and Greek letters and to accomplish tasks such as changing fonts. These sequences of characters are known as *control sequences*.

1.3 Characters and Control Sequences

We now describe in more detail some of the features of LaTeX illustrated in the above example.

Most characters on the keyboard, such as letters and numbers, have their usual meaning. However the characters

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

are used for special purposes within LaTeX. Thus typing one of these characters will not produce the corresponding character in the final document. Of course these characters are very rarely used in ordinary text, and there are methods of producing them when they are required in the final document.

In order to typeset a mathematical document it is necessary to produce a considerable number of special mathematical symbols. One also needs to be able to change fonts. Also mathematical documents often contain arrays of numbers or symbols (matrices) and other complicated expressions. These are produced in LaTeX using *control sequences*. Most control sequences consist of a backslash `\` followed by a string of (upper or lower case) letters. For example, `\alpha`, `\textit` and `\sum` are control sequences.

In the example above we used the control sequences `\textit` and `\textbf` to change the font to *italic* and **boldface** respectively. Also we used the control sequences `\to`,

`\in`, `\delta` and `\epsilon` to produce the mathematical symbols $\rightarrow$ and $\in$ and the Greek letters δ and ϵ

There is another variety of control sequence which consists of a backslash followed by a *single* character that is not a letter. Examples of control sequences of this sort are `\{`, `\"` and `\$`.

The special characters `{` and `}` are used for *grouping* purposes. Everything enclosed within matching pair of such brackets is treated as a single unit. We have applied these brackets in the example above whenever we changed fonts. We shall see other instances where one needs to use `{` and `}` in LaTeX to group words and symbols together (e.g., when we need to produce superscripts and subscripts which contain more than one symbol).

The special character `$` is used when one is changing from ordinary text to a mathematical expression and when one is changing back to ordinary text. Thus we used

```
for all $\epsilon > 0$ and for all $x \in D$,
```

to produce the phrase

for all $\epsilon > 0$ and for all $x \in D$,

in the example given above. Note also that we used `\[` and `\]` in the example above to mark the beginning and end respectively of a mathematical formula that is displayed on a separate line.

The remaining special characters

```
^ _ % ~ # &
```

have special purposes within LaTeX that we shall discuss later.

2 Producing Simple Documents using LaTeX

2.1 Producing a LaTeX Input File

We describe the structure of a typical LaTeX input file.

The first line of the input file should consist of a `\documentclass` command. The recommended such `\documentclass` command for mathematical articles and similar documents has the form

```
\documentclass[a4paper,12pt]{article}
```

(You do not have to worry about what this command means when first learning to use LaTeX: its effect is to ensure that the final document is correctly positioned on A4 size paper and that the text is of a size that is easy to read.) There are variants of this `\documentclass` command which are appropriate for letters or for books.

The `documentstyle` command may be followed by certain other optional commands, such as the `\pagestyle` command. It is not necessary to find out about these commands when first learning to use LaTeX.

After the `\documentclass` command and these other optional commands, we place the command

```
\begin{document}
```

This command is then followed by the main body of the text, in the format prescribed by the rules of LaTeX.

Finally, we end the input file with a line containing the command

```
\end{document}
```

2.2 Producing Ordinary Text using LaTeX

To produce a simple document using LaTeX one should create a LaTeX input file, beginning with a `\documentclass` command and the `\begin{document}` command, as described above. The input file should end with the `\end{document}` command, and the text of the document should be sandwiched between the `\begin{document}` and `\end{document}` commands in the manner described below.

If one merely wishes to type in ordinary text, without complicated mathematical formulae or special effects such as font changes, then one merely has to type it in as it is, leaving a completely blank line between successive paragraphs. You do not have to worry about paragraph indentation: LaTeX will automatically indent all paragraphs with the exception of the first paragraph of a new section (unless you take special action to override the conventions adopted by LaTeX)

For example, suppose that we wish to create a document containing the following paragraphs:

> If one merely wishes to type in ordinary text, without complicated mathematical formulae or special effects such as font changes, then one merely has to type it in as it is, leaving a completely blank line between successive paragraphs.
>
> You do not have to worry about paragraph indentation: all paragraphs will be indented with the exception of the first paragraph of a new section. One must take care to distinguish between the 'left quote' and the 'right quote' on the computer terminal. Also, one should use two 'single quote'

characters in succession if one requires “double quotes”. One should never use the (undirected) ‘double quote’ character on the computer terminal, since the computer is unable to tell whether it is a ‘left quote’ or a ‘right quote’. One also has to take care with dashes: a single dash is used for hyphenation, whereas three dashes in succession are required to produce a dash of the sort used for punctuation—such as the one used in this sentence.

To create this document using LaTeX we use the following input file:

```
\documentclass[a4paper,12pt]{article}
\begin{document}

If one merely wishes to type in ordinary text, without
complicated mathematical formulae or special effects such
as font changes, then one merely has to type it in as it
is, leaving a completely blank line between successive
paragraphs.

You do not have to worry about paragraph indentation:
all paragraphs will be indented with the exception of
the first paragraph of a new section.

One must take care to distinguish between the `left quote'
and the `right quote' on the computer terminal.  Also, one
should use two `single quote' characters in succession if
one requires ``double quotes''.  One should never use the
(undirected) `double quote' character on the computer
terminal, since the computer is unable to tell whether it
is a `left quote' or a `right quote'.  One also has to
take care with dashes: a single dash is used for
hyphenation, whereas three dashes in succession are required
to produce a dash of the sort used for punctuation---such as
the one used in this sentence.

\end{document}
```

Having created the input file, one then has to run it through the LaTeX program and then print it out the resulting output file (known as a ‘DVI’ file).

2.3 Blank Spaces and Carriage Returns in the Input File

LaTeX treats the carriage return at the end of a line as though it were a blank space. Similarly LaTeX treats tab characters as blank spaces. Moreover, LaTeX regards a sequence of blank spaces as though it were a single space, and similarly it will ignore blank spaces at the beginning or end of a line in the input file. Thus, for example, if we type

```
This is
    a
        silly
  example   of   a
file with many spaces.



                       This is the beginning
of a new paragraph.
```

then we obtain

> This is a silly example of a file with many spaces.
> This is the beginning of a new paragraph.

It follows immediately from this that one will obtain the same results whether one types one space or two spaces after a full stop: LaTeX does not distinguish between the two cases.

Any spaces which follow a control sequence will be ignored by LaTeX.

If you really need a blank space in the final document following whatever is produced by the control sequence, then you must precede this blank by a *backslash* `\`. Thus in order to obtain the sentence

> LaTeX is a very powerful computer typesetting program.

we must type

```
\LaTeX\ is a very powerful computer typesetting program.
```

(Here the control sequence `TeX` is used to produce the LaTeX logo.)

In general, preceding a blank space by a backslash forces LaTeX to include the blank space in the final document.

As a general rule, you should never put a blank space after a left parenthesis or before a right parenthesis. If you were to put a blank space in these places, then you run the risk that LaTeX might start a new line immediately after the left parenthesis or before the right parenthesis, leaving the parenthesis marooned at the beginning or end of a line.

2.4 Quotation Marks and Dashes

Single quotation marks are produced in LaTeX using ` ` ` and `'`. Double quotation marks are produced by typing ` `` ` and `''`. (The 'undirected double quote character `"` produces double right quotation marks: it should *never* be used where left quotation marks are required.)

LaTeX allows you to produce dashes of various length, known as 'hyphens', 'en-dashes' and 'em-dashes'. Hyphens are obtained in LaTeX by typing `-`, en-dashes by typing `--` and em-dashes by typing `---`.

One normally uses en-dashes when specifying a range of numbers. Thus for example, to specify a range of page numbers, one would type

```
on pages 155--219.
```

Dashes used for punctuating are often typeset as em-dashes, especially in older books. These are obtained by typing `---`.

The dialogue

> "You *were* a little grave," said Alice.
>
> "Well just then I was inventing a new way of getting over a gate—would you like to hear it?"
>
> "Very much indeed," Alice said politely.
>
> "I'll tell you how I came to think of it," said the Knight. "You see, I said to myself 'The only difficulty is with the feet: the *head* is high enough already.' Now, first I put my head on the top of the gate—then the head's high enough—then I stand on my head—then the feet are high enough, you see—then I'm over, you see."

(taken from *Alice through the Looking Glass*, by Lewis Carroll) illustrates the use of quotation marks and dashes. It is obtained in LaTeX from the following input:

```
``You \emph{were} a little grave,'' said Alice.

``Well just then I was inventing a new way of
getting over a gate---would you like to hear it?''

``Very much indeed,'' Alice said politely.

``I'll tell you how I came to think of it,'' said
the Knight.  ``You see, I said to myself `The only
difficulty is with the feet: the \emph{head} is
high enough already.'  Now, first I put my head on
the top of the gate---then the head's high
```

```
enough---then I stand on my head---then the feet
are high enough, you see---then I'm over, you see.''
```

Sometimes you need single quotes immediately following double quotes, or vica versa, as in

> "I regard computer typesetting as being reasonably 'straightforward' "
> he said.

The way to typeset this correctly in LaTeX is to use the control sequence `\,` between the quotation marks, so as to obtain the necessary amount of separation. The above example is thus produced with the input

```
``I regard computer typesetting as being reasonably
`straightforward'\,'' he said.
```

2.5 Section Headings in LaTeX

Section headings of various sizes are produced (in the **article** document style) using the commands `\section`,`\subsection` and `\subsubsection` commands. LaTeX will number the sections and subsections automatically. The title of the section should be surrounded by curly brackets and placed immediately after the relevant command. Thus if we type

```
\section{Section Headings}

We explain in this section how to obtain headings
for the various sections and subsections of our
document.

\subsection{Headings in the `article' Document Style}

In the `article' style, the document may be divided up
into sections, subsections and subsubsections, and each
can be given a title, printed in a boldface font,
simply by issuing the appropriate command.
```

then the title of the section and that of the subsection will be printed in a large boldface font, and will be numbered accordingly.

Other document styles (such as the **book** and **letter** styles) have other 'sectioning' commands available (for example, the **book** style has a `\chapter` command for beginning a new chapter).

Sometimes one wishes to suppress the automatic numbering provided by LaTeX. This can be done by placing an asterisk before the title of the section or subsection. Thus, for example, the section numbers in the above example could be suppressed by typing

```
\section*{Section Headings}

We explain in this section how to obtain headings
for the various sections and subsections of our
document.

\subsection*{Headings in the ‘article’ Document Style}

In the ‘article’ style, the document may be divided up
into sections, subsections and subsubsections, and each
can be given a title, printed in a boldface font,
simply by issuing the appropriate command.
```

2.6 Changing Fonts in Text Mode

LaTeX has numerous commands for changing the typestyle. The most useful of these is `\emph{`*text*`}` which *emphasizes* some piece of text, setting it usually in an *italic font* (unless the surrounding text is already italicized). Thus for example, the text

> The basic results and techniques of *Calculus* were discovered and developed by *Newton* and *Leibniz*, though many of the basic ideas can be traced to earlier work of *Cavalieri*, *Fermat*, *Barrow* and others.

is obtained by typing

```
The basic results and techniques of \emph{Calculus}
were discovered and developed by \emph{Newton}
and \emph{Leibniz}, though many of the basic ideas
can be traced to earlier work of \emph{Cavalieri},
\emph{Fermat}, \emph{Barrow} and others.
```

Another useful font-changing command is `\textbf{`*text*`}`, which typesets the specified portion of text in **boldface**.

A *font family* or *typeface* in LaTeX consists of a collection of related fonts characterized by *size*, *shape* and *series*. The font families available in LaTeX include roman, sans serif and `typewriter`:

- Roman is normally the default family and includes upright, *italic*, *slanted*, SMALL CAPS and **boldface** fonts of various sizes.

- There is a sans serif family with upright, *slanted* and **boldface** fonts of various sizes.

- There is a typewriter family with upright, *italic*, *slanted* and SMALL CAPS fonts of various sizes.

The sizes of fonts used in LaTeX are can be determined and changed by means of the control sequences `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge` and `\HUGE`:

This text is `tiny`.

This text is `scriptsize`.

This text is `footnotesize`.

This text is `small`.

This text is `normalsize`.

This text is `large`.

This text is `Large`.

This text is `LARGE`.

This text is `huge`.

This text is `Huge`.

The *shape* of a font can be upright, *italic*, *slanted* or SMALL CAPS:

- The LaTeX command `\textup{`*text*`}` typesets the specified text with an upright shape: this is normally the default shape.

- *The LaTeX command* `\textit{`*text*`}` *typesets the specified text with an italic shape.*

- *The LaTeX command* `\textsl{`*text*`}` *typesets the specified text with a slanted shape: slanted text is similar to italic.*

- THE LATEX COMMAND `\textsc{`*text*`}` TYPESETS THE SPECIFIED TEXT WITH A SMALL CAPS SHAPE IN WHICH ALL LETTERS ARE CAPITALS (WITH UPPERCASE LETTERS TALLER THAN LOWERCASE LETTERS).

The *series* of a font can be medium (the default) or **boldface**:

- The LaTeX command `\textmd{`*text*`}` typesets the specified text with a medium series font.

- **The LaTeX command** `\textbf{`*text*`}` **typesets the specified text with a boldface series font.**

If the necessary fonts are available, one can combine changes to the size, shape and series of a font, for example producing ***boldface slanted text*** by typing

```
\textbf{\textsl{boldface slanted text}}.
```

There are in LaTeX font declarations corresponding to the the font-changing commands described above. When included in the LaTeX input such declarations determine the type-style of the subsequent text (till the next font declaration or the end of the current 'group' delimited by curly brackets or by appropriate `\begin` and `\end` commands). Here is a list of font-changing commands and declarations in text mode:

Command	*Declaration*	
`\textrm`	`\rmfamily`	Roman family
`\textsf`	`\sffamily`	Sans serif family
`\texttt`	`\ttfamily`	`Typewriter family`
`\textup`	`\upshape`	Upright shape
`\textit`	`\itshape`	*Italic shape*
`\textsl`	`\slshape`	*Slanted shape*
`\textsc`	`\scshape`	SMALL CAPS SHAPE
`\textmd`	`\mdseries`	Medium series
`\textbf`	`\bfseries`	**Boldface series**

2.7 Accents used in Text

There are a variety of control sequences for producing accents. For example, the control sequence `\'{o}` produces an acute accent on the letter `o`. Thus typing

```
Se\'{a}n \'{O} Cinn\'{e}ide.
```

produces

Seán Ó Cinnéide.

Similarly we use the control sequence `` \` `` to produce the grave accent in 'algèbre' and we use `\"` to produce the umlaut in 'Universität'. The accents provided by LaTeX include the following:

`\'{e}`	é	e.g., `math\'{e}matique` yields 'mathématique'
``\`{e}``	è	e.g., ``alg\`{e}bre`` yields 'algèbre'
`\^{e}`	ê	e.g., `h\^{o}te` yields 'hôte'
`\"{o}`	ö	e.g., `H\"{o}lder` yields 'Hölder'
`\~{n}`	ñ	e.g., `ma\~{n}ana` yields 'mañana'
`\={o}`	ō	
`\.{o}`	ȯ	
`\u{o}`	ŏ	
`\v{c}`	č	e.g., `\v{C}ech` yields 'Čech'
`\H{o}`	ő	
`\t{oo}`	o͡o	
`\c{c}`	ç	e.g., `gar\c{c}on` yields 'garçon'
`\d{o}`	ọ	
`\b{o}`	o̱	

These accents are for use in ordinary text. They cannot be used within mathematical formulae, since different control sequences are used to produce accents within mathematics.

The control sequences `\i` and `\j` produce dotless 'ı' and 'ȷ'. These are required when placing an accent on the letter. Thus ı́ is produced by typing `\'{\i}`.

2.8 Active Characters and Special Symbols in Text

The 'active characters'

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

have special purposes within LaTeX. Thus they cannot be produced in the final document simply by typing them directly. On the rare occasions when one needs to use the special characters

$ % & _ { }

in the final document, they can be produced by typing the control sequences

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

respectively. However the characters `\`, `^` and `~` cannot be produced simply by preceding them with a backslash. They can however be produced using `\char92` (in the `\texttt` font only), `\char94` and `\char126` respectively. (The decimal numbers 92, 94 and 126 are the ASCII codes of these characters.)

Other special symbols can be introduced into text using the appropriate control sequences:

Symbol	*Control Sequence*
œ, Œ	`\oe, \OE`
æ, Æ	`\ae, \AE`
å, Å	`\aa, \AA`
ø, Ø	`\o, \O`
ł, Ł	`\l, \L`
ß	`\ss`
¿	`?‘`
¡	`!‘`
†	`\dag`
‡	`\ddag`
§	`\S`
¶	`\P`
©	`\copyright`
£	`\pounds`
ı	`\i`
ȷ	`\j`

3 Producing Mathematical Formulae using LaTeX

3.1 Mathematics Mode

In order to obtain a mathematical formula using LaTeX, one must enter *mathematics mode* before the formula and leave it afterwards. Mathematical formulae can occur either embedded in text or else displayed between lines of text. When a formula occurs within the text of a paragraph one should place a `$` sign before and after the formula, in order to enter and leave mathematics mode. Thus to obtain a sentence like

> Let f be the function defined by $f(x) = 3x + 7$, and let a be a positive real number.

one should type

```
Let $f$ be the function defined by $f(x) = 3x + 7$, and
let $a$ be a positive real number.
```

In particular, note that even mathematical expressions consisting of a single character, like f and a in the example above, are placed within `$` signs. This is to ensure that they are set in italic type, as is customary in mathematical typesetting.

LaTeX also allows you to use `\(` and `\)` to mark the beginning and the end respectively of a mathematical formula embedded in text. Thus

> Let f be the function defined by $f(x) = 3x + 7$.

may be produced by typing

```
Let \( f \) be the function defined by \( f(x) = 3x + 7 \).
```

However this use of `\(` ... `\)` is only permitted in LaTeX: other dialects of TeX such as Plain TeX and AmSTeX use `$` ... `$`.

In order to obtain an mathematical formula or equation which is displayed on a line by itself, one places `\[` before and `\]` after the formula. Thus to obtain

> If $f(x) = 3x + 7$ and $g(x) = x + 4$ then
> $$f(x) + g(x) = 4x + 11$$
> and
> $$f(x)g(x) = 3x^2 + 19x + 28.$$

one would type

```
If $f(x) = 3x + 7$ and $g(x) = x + 4$ then
\[ f(x) + g(x) = 4x + 11 \]
and
\[ f(x)g(x) = 3x^2 + 19x +28. \]
```

(Here the character `^` is used to obtain a superscript.)

LaTeX provides facilities for the automatic numbering of displayed equations. If you want an numbered equation then you use `\begin{equation}` and `\end{equation}` instead of using `\[` and `\]` . Thus

```
If $f(x) = 3x + 7$ and $g(x) = x + 4$ then
\begin{equation}
f(x) + g(x) = 4x + 11
\end{equation}
and
\begin{equation}
f(x)g(x) = 3x^2 + 19x +28.
\end{equation}
```

produces

> If $f(x) = 3x + 7$ and $g(x) = x + 4$ then
> $$f(x) + g(x) = 4x + 11 \tag{1}$$
> and
> $$f(x)g(x) = 3x^2 + 19x + 28. \tag{2}$$

3.2 Characters in Mathematics Mode

All the characters on the keyboard have their standard meaning in mathematics mode, with the exception of the characters

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

Letters are set in italic type. In mathematics mode the character `'` has a special meaning: typing `$u' + v''$` produces $u' + v''$ When in mathematics mode the spaces you type between letters and other symbols do not affect the spacing of the final result, since LaTeX determines the spacing of characters in formulae by its own internal rules. Thus `$u v + w = x$` and `$uv+w=x$` both produce $uv + w = x$ You can also type carriage returns where necessary in your input file (e.g., if you are typing in a complicated formula with many Greek characters and funny symbols) and this will have no effect on the final result if you are in mathematics mode.

To obtain the characters

```
#   $   %   &   _   {   }
```

in mathematics mode, one should type

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

To obtain in mathematics mode, one may type `\backslash`.

3.3 Superscripts and Subscripts

Subscripts and superscripts are obtained using the special characters `_` and `^` respectively. Thus the identity

$$ds^2 = dx_1^2 + dx_2^2 + dx_3^2 - c^2 dt^2$$

is obtained by typing

```
\[ ds^2 = dx_1^2 + dx_2^2 + dx_3^2 - c^2 dt^2 \]
```

It can also be obtained by typing

```
\[ ds^2 = dx^2_1 + dx^2_2 + dx^2_3 - c^2 dt^2 \]
```

since, when a superscript is to appear above a subscript, it is immaterial whether the superscript or subscript is the first to be specified.

Where more than one character occurs in a superscript or subscript, the characters involved should be enclosed in curly brackets. For example, the polynomial $x^{17} - 1$ is obtained by typing `$x^{17} - 1$`.

One may not type expressions such as `$s^n^j$` since this is ambiguous and could be interpreted either as s^{nj} or as s^{n^j} The first of these alternatives is obtained by typing `$s^{n j}$`, the second by typing `$s^{n^j}$`. A similar remark applies to subscripts. Note that one can obtain in this way double superscripts (where a superscript is placed on a superscript) and double subscripts.

It is sometimes necessary to obtain expressions in which the horizontal ordering of the subscripts is significant. One can use an 'empty group' `{}` to separate superscripts and subscripts that must follow one another. For example, the identity

$$R_i{}^j{}_{kl} = g^{jm} R_{imkl} = - g^{jm} R_{mikl} = - R^j{}_{ikl}$$

can be obtained by typing

```
\[ R_i{}^j{}_{kl} = g^{jm} R_{imkl}
   = - g^{jm} R_{mikl} = - R^j{}_{ikl} \]
```

3.4 Greek Letters

Greek letters are produced in mathematics mode by preceding the name of the letter by a backslash `\`. Thus to obtain the formula $A = \pi r^2$ one types `A = \pi r^2`.

Here are the control sequences for the standard forms of the lowercase Greek letters:-

α	`\alpha`	ι	`\iota`	ρ	`\rho`
β	`\beta`	κ	`\kappa`	σ	`\sigma`
γ	`\gamma`	λ	`\lambda`	τ	`\tau`
δ	`\delta`	μ	`\mu`	υ	`\upsilon`
ϵ	`\epsilon`	ν	`\nu`	ϕ	`\phi`
ζ	`\zeta`	ξ	`\xi`	χ	`\chi`
η	`\eta`	o	`o`	ψ	`\psi`
θ	`\theta`	π	`\pi`	ω	`\omega`

There is no special command for omicron: just use `o`.

Some Greek letters occur in variant forms. The variant forms are obtained by preceding the name of the Greek letter by 'var'. The following table lists the usual form of these letters and the variant forms:-

ϵ	`\epsilon`	ε	`\varepsilon`
θ	`\theta`	ϑ	`\vartheta`
π	`\pi`	ϖ	`\varpi`
ρ	`\rho`	ϱ	`\varrho`
σ	`\sigma`	ς	`\varsigma`
ϕ	`\phi`	φ	`\varphi`

Upper case Greek letters are obtained by making the first character of the name upper case. Here are the control sequence for the uppercase letters:—

Γ	`\Gamma`	Ξ	`\Xi`	Φ	`\Phi`
Δ	`\Delta`	Π	`\Pi`	Ψ	`\Psi`
Θ	`\Theta`	Σ	`\Sigma`	Ω	`\Omega`
Λ	`\Lambda`	Υ	`\Upsilon`		

3.5 Mathematical Symbols

There are numerous mathematical symbols that can be used in mathematics mode. These are obtained by typing an appropriate control sequence.

Miscellaneous Symbols:

$\aleph$	`\aleph`	$\prime$	`\prime`	$\forall$	`\forall`
$\hbar$	`\hbar`	$\emptyset$	`\emptyset`	$\exists$	`\exists`
$\imath$	`\imath`	∇	`\nabla`	$\neg$	`\neg`
$\jmath$	`\jmath`	$\surd$	`\surd`	$\flat$	`\flat`
ℓ	`\ell`	$\top$	`\top`	$\natural$	`\natural`
$\wp$	`\wp`	$\bot$	`\bot`	$\sharp$	`\sharp`
$\Re$	`\Re`	$\|$	`|`	$\clubsuit$	`\clubsuit`
$\Im$	`\Im`	$\angle$	`\angle`	$\diamondsuit$	`\diamondsuit`
∂	`\partial`	$\triangle$	`\triangle`	$\heartsuit$	`\heartsuit`
∞	`\infty`	$\backslash$	`\backslash`	$\spadesuit$	`\spadesuit`

"Large" Operators:

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

Binary Operations:

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

Relations:

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

Negated Relations:

$\not<$	`\not<`	$\not>$	`\not>`	$\not=$	`\not=`
$\not\leq$	`\not\leq`	$\not\geq$	`\not\geq`	$\not\equiv$	`\not\equiv`
$\not\prec$	`\not\prec`	$\not\succ$	`\not\succ`	$\not\sim$	`\not\sim`
$\not\preceq$	`\not\preceq`	$\not\succeq$	`\not\succeq`	$\not\simeq$	`\not\simeq`
$\not\subset$	`\not\subset`	$\not\supset$	`\not\supset`	$\not\approx$	`\not\approx`
$\not\subseteq$	`\not\subseteq`	$\not\supseteq$	`\not\supseteq`	$\not\cong$	`\not\cong`
$\not\sqsubseteq$	`\not\sqsubseteq`	$\not\sqsupseteq$	`\not\sqsupseteq`	$\not\asymp$	`\not\asymp`

Arrows:

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

Openings:

$[$	`\lbrack`	$\lfloor$	`\lfloor`	$\lceil$	`\lceil`
$\{$	`\lbrace`	$\langle$	`\langle`		

Closings:

$]$	`\rbrack`	$\rfloor$	`\rfloor`	$\rceil$	`\rceil`
$\}$	`\rbrace`	$\rangle$	`\rangle`		

Alternative Names:

$\neq$	`\ne` or `\neq`	(same as `\not=`)
$\leq$	`\le`	(same as `\leq`)
$\geq$	`\ge`	(same as `\geq`)
$\{$	`\{`	(same as `\lbrace`)
$\}$	`\}`	(same as `\lbrace`)
$\to$	`\to`	(same as `\rightarrow`)
$\gets$	`\gets`	(same as `\leftarrow`)
$\owns$	`\owns`	(same as `\ni`)
$\land$	`\land`	(same as `\wedge`)
$\lor$	`\lor`	(same as `\vee`)
$\lnot$	`\lnot`	(same as `\neg`)
$\vert$	`\vert`	(same as `|`)
$\Vert$	`\Vert`	(same as `\\|`)
$\iff$	`\iff`	(same as `\Longleftrightarrow`, but with extra space at each end)
$\colon$	`\colon`	(same as `:`, but with less space around it and less likelihood of a line break after it)

3.6 Changing Fonts in Mathematics Mode

(The following applies to LaTeX2ε, a recent version of LaTeX. It does not apply to older versions of LaTeX.)

The ‘math italic’ font is automatically used in mathematics mode unless you explicitly change the font. The rules for changing the font in mathematics mode are rather different to those applying when typesetting ordinary text. In mathematics mode any change only applies to the single character or symbol that follows (or to any text enclosed within curly brackets immediately following the control sequence). Also, to change a character to the roman or boldface font, the control sequences `\mathrm` and `\mathbf` must be used (rather than `\textrm` and `\textbf`).

The following example illustrates the use of boldface in mathematical formulae. To obtain

> Let $\mathbf{u}$,$\mathbf{v}$ and $\mathbf{w}$ be three vectors in $\mathbf{R}^3$. The volume V of the parallelepiped with corners at the points $\mathbf{0}$, $\mathbf{u}$, $\mathbf{v}$, $\mathbf{w}$, $\mathbf{u}+\mathbf{v}$, $\mathbf{u}+\mathbf{w}$, $\mathbf{v}+\mathbf{w}$ and $\mathbf{u}+\mathbf{v}+\mathbf{w}$ is given by the formula
>
> $$V = (\mathbf{u} \times \mathbf{v}) \cdot \mathbf{w}.$$

one could type

```
Let $\mathbf{u}$,$\mathbf{v}$ and $\mathbf{w}$ be three
```

```
vectors in ${\mathbf R}^3$. The volume~$V$ of the
parallelepiped with corners at the points
$\mathbf{0}$, $\mathbf{u}$, $\mathbf{v}$,
$\mathbf{w}$, $\mathbf{u}+\mathbf{v}$,
$\mathbf{u}+\mathbf{w}$, $\mathbf{v}+\mathbf{w}$
and $\mathbf{u}+\mathbf{v}+\mathbf{w}$
is given by the formula
\[ V = (\mathbf{u} \times \mathbf{v}) \cdot \mathbf{w}.\]
```

There is also a 'calligraphic' font available in mathematics mode. This is obtained using the control sequence `\cal`. *This font can only be used for uppercase letters.* These calligraphic letters have the form

$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}.$$

3.7 Standard Functions (sin, cos etc.)

The names of certain standard functions and abbreviations are obtained by typing a backlash `\` before the name. For example, one obtains

$$\cos(\theta + \phi) = \cos\theta \cos\phi - \sin\theta \sin\phi$$

by typing

```
\[ \cos(\theta + \phi) = \cos \theta \cos \phi
      - \sin \theta \sin \phi \]
```

The following standard functions are represented by control sequences defined in LaTeX:

`\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`

Names of functions and other abbreviations not in this list can be obtained by converting to the roman font. Thus one obtains $\mathrm{cosec} A$ by typing `$\mathrm{cosec} A$`. Note that if one were to type simply `$cosec A$` one would obtain $cosec A$, because LaTeX has treated `cosec A` as the product of six quantities c, o, s, e, c and A and typeset the formula accordingly.

3.8 Text Embedded in Displayed Equations

Text can be embedded in displayed equations (in LaTeX) by using `\mbox{`*embedded text*`}`. For example, one obtains

$$M^\perp = \{ f \in V' : f(m) = 0 \mbox{ for all } m \in M \}.$$

by typing

```
\[ M^\bot = \{ f \in V' : f(m) = 0 \mbox{ for all } m \in M \}.\]
```

Note the blank spaces before and after the words 'for all' in the above example. Had we typed

```
\[ M^\bot = \{ f \in V' : f(m) = 0 \mbox{for all} m \in M \}.\]
```

we would have obtained

$$M^\perp = \{ f \in V' : f(m) = 0 \mbox{for all} m \in M \}.$$

(In Plain TeX one should use `\hbox` in place of `\mbox`.)

3.9 Fractions and Roots

Fractions of the form

$$\frac{numerator}{denominator}$$

are obtained in LaTeX using the construction

`\frac{`*numerator*`}{`*denominator*`}`.

For example, to obtain

> The function f is given by
>
> $$f(x) = 2x + \frac{x - 7}{x^2 + 4}$$
>
> for all real numbers x.

one would type

```
The function $f$ is given by
\[ f(x) = 2x + \frac{x - 7}{x^2 + 4}\]
for all real numbers $x$.
```

To obtain square roots one uses the control sequence

`\sqrt{`*expression*`}`.

For example, to obtain

> The roots of a quadratic polynomial $ax^2 + bx + c$ with $a \neq 0$ are given by the formula
> $$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

one would type

```
The roots of a quadratic polynomial $a x^2 + bx + c$ with
$a \neq 0$ are given by the formula
\[ \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \]
```

In LaTeX, an nth root is produced using

`\sqrt[n]{`*expression*`}`.

For example, to obtain

> The roots of a cubic polynomial of the form $x^3 - 3px - 2q$ are given by the formula
> $$\sqrt[3]{q + \sqrt{q^2 - p^3}} + \sqrt[3]{q - \sqrt{q^2 - p^3}}$$
> where the values of the two cube roots must are chosen so as to ensure that their product is equal to p.

in LaTeX, one would type

```
The roots of a cubic polynomial of the form $x^3 - 3px - 2q$
are given by the formula
\[ \sqrt[3]{q + \sqrt{ q^2 - p^3 }}
  + \sqrt[3]{q - \sqrt{ q^2 - p^3 }} \]
where the values of the two cube roots must are chosen
so as to ensure that their product is equal to $p$.
```

3.10 Ellipsis (i.e., 'three dots')

Ellipsis (three dots) is produced in mathematics mode using the control sequences `\ldots` (for dots aligned with tbe baseline of text), and `\cdots` (for dots aligned with the centreline of mathematical formulae). Thus the formula

$$f(x_1, x_2, \ldots, x_n) = x_1^2 + x_2^2 + \cdots + x_n^2$$

is obtained by typing

```
\[ f(x_1, x_2,\ldots, x_n) = x_1^2 + x_2^2 + \cdots + x_n^2 \]
```

Similarly the formula

$$\frac{1 - x^{n+1}}{1 - x} = 1 + x + x^2 + \cdots + x^n$$

is produced using `\cdots`, by typing

```
\[ \frac{1 - x^{n+1}}{1 - x} = 1 + x + x^2 + \cdots + x^n \]
```

3.11 Accents in Mathematics Mode

There are various control sequences for producing underlining, overlining and various accents in mathematics mode. The following table lists these control sequences, applying them to the letter a:

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

It should be borne in mind that when a character is underlined in a mathematical manuscript then it is normally typeset in bold face without any underlining. Underlining is used very rarely in print.

The control sequences such as `\'` and `\"`, used to produce accents in ordinary text, may not be used in mathematics mode.

3.12 Brackets and Norms

The frequently used left delimiters include (, [and {, which are obtained by typing `(`, `[` and `\{` respectively. The corresponding right delimiters are of course obtained by typing `)`, `]` and `\}`. In addition $|$ and $\|$ are used as both left and right delimiters, and are obtained by typing `|` and `\|` respectively. For example, we obtain

Let X be a Banach space and let $f\colon B \to \textbf{R}$ be a bounded linear functional on X. The *norm* of f, denoted by $\|f\|$, is defined by

$$\|f\| = \inf \{ K \in [0,+\infty) : |f(x)| \leq K \|x\| \mbox{ for all } x \in X \}.$$

by typing

```
Let $X$ be a Banach space and let $f \colon B \to \textbf{R}$
be a bounded linear functional on $X$. The \textit{norm} of
$f$, denoted by $\|f\|$, is defined by
\[ \|f\| = \inf \{ K \in [0,+\infty) :
          |f(x)| \leq K \|x\| \mbox{ for all } x \in X \}.\]
```

Larger delimiters are sometimes required which have the appropriate height to match the size of the subformula which they enclose. Consider, for instance, the problem of typesetting the following formula:

$$f(x,y,z) = 3y^2 z \left(3 + \frac{7x+5}{1 + y^2} \right).$$

The way to type the large parentheses is to type `\left(` for the left parenthesis and `\right)` for the right parenthesis, and let LaTeX do the rest of the work for you. Thus the above formula was obtained by typing

```
\[ f(x,y,z) = 3y^2 z \left( 3 + \frac{7x+5}{1 + y^2} \right).\]
```

If you type a delimiter which is preceded by `\left` then LaTeX will search for a corresponding delimiter preceded by `\right` and calculate the size of the delimiters required to enclose the intervening subformula. One is allowed to balance a `\left(` with a `\right]` (say) if one desires: there is no reason why the enclosing delimiters have to have the same shape. One may also nest pairs of delimiters within one another: by typing

```
\[ \left| 4 x^3 + \left( x + \frac{42}{1+x^4} \right) \right|.\]
```

we obtain

$$\left| 4 x^3 + \left(x + \frac{42}{1+x^4} \right) \right|.$$

By typing `\left.` and `\right.` one obtains *null delimiters* which are completely invisible. Consider, for example, the problem of typesetting

$$\left. \frac{du}{dx} \right|_{x=0}.$$

We wish to make the vertical bar big enough to match the derivative preceding it. To do this, we suppose that the derivative is enclosed by delimiters, where the left delimiter is invisible and the right delimiter is the vertical line. The invisible delimiter is produced using `\left.` and thus the whole formula is produced by typing

$$\left.\frac{du}{dx}\right|_{x=0}.$$

3.13 Multiline Formulae in LaTeX

Consider the problem of typesetting the formula

$$\begin{eqnarray*}
\cos 2\theta & = & \cos^2 \theta - \sin^2 \theta \\
& = & 2 \cos^2 \theta - 1.
\end{eqnarray*}$$

It is necessary to ensure that the = signs are aligned with one another. In LaTeX, such a formula is typeset using the `eqnarray*` environment. The above example was obtained by typing the lines

```
\begin{eqnarray*}
\cos 2\theta & = & \cos^2 \theta - \sin^2 \theta \\
             & = & 2 \cos^2 \theta - 1.
\end{eqnarray*}
```

Note the use of the special character `&` as an *alignment tab*. When the formula is typeset, the part of the second line of the formula beginning with an occurrence of `&` will be placed immediately beneath that part of the first line of the formula which begins with the corresponding occurrence of `&`. Also `\\` is used to separate the lines of the formula.

Although we have placed corresponding occurrences of `&` beneath one another in the above example, it is not necessary to do this in the input file. It was done in the above example merely to improve the appearance (and readability) of the input file.

The more complicated example

If $h \leq \frac{1}{2}|\zeta - z|$ then

$$|\zeta - z - h| \geq \frac{1}{2}|\zeta - z|$$

and hence

$$\left|\frac{1}{\zeta - z - h} - \frac{1}{\zeta - z}\right| = \left|\frac{(\zeta - z) - (\zeta - z - h)}{(\zeta - z - h)(\zeta - z)}\right|$$

$$
\begin{array}{rcl}
& = & \left| \frac{h}{(\zeta - z - h)(\zeta - z)} \right| \\
& \leq & \frac{2|h|}{|\zeta - z|^2}.
\end{array}
$$

was obtained by typing

```
If $h \leq \frac{1}{2} |\zeta - z|$ then
\[ |\zeta - z - h| \geq \frac{1}{2} |\zeta - z|\]
and hence
\begin{eqnarray*}
\left| \frac{1}{\zeta - z - h} - \frac{1}{\zeta - z} \right|
& = & \left|
\frac{(\zeta - z) - (\zeta - z - h)}{(\zeta - z - h)(\zeta - z)}
\right| \\  & = &
\left| \frac{h}{(\zeta - z - h)(\zeta - z)} \right| \\
  & \leq & \frac{2 |h|}{|\zeta - z|^2}.
\end{eqnarray*}
```

The asterisk in `eqnarray*` is put there to suppress the automatic equation numbering produced by LaTeX. If you wish for an automatically numbered multiline formula, you should use `\begin{eqnarray}` and `\end{eqnarray}`.

3.14 Matrices and other arrays in LaTeX

Matrices and other arrays are produced in LaTeX using the **array** environment. For example, suppose that we wish to typeset the following passage:

> The *characteristic polynomial* $\chi(\lambda)$ of the 3×3 matrix
>
> $$\left(\begin{array}{ccc} a & b & c \\ d & e & f \\ g & h & i \end{array} \right)$$
>
> is given by the formula
>
> $$\chi(\lambda) = \left| \begin{array}{ccc} \lambda - a & -b & -c \\ -d & \lambda - e & -f \\ -g & -h & \lambda - i \end{array} \right|.$$

This passage is produced by the following input:

```
The \emph{characteristic polynomial} $\chi(\lambda)$ of the
$3 \times 3$~matrix
\[ \left( \begin{array}{ccc}
a & b & c \\
d & e & f \\
g & h & i \end{array} \right)\]
is given by the formula
\[ \chi(\lambda) = \left| \begin{array}{ccc}
\lambda - a & -b & -c \\
-d & \lambda - e & -f \\
-g & -h & \lambda - i \end{array} \right|.\]
```

First of all, note the use of `\left` and `\right` to produce the large delimiters around the arrays. As we have already seen, if we use

```
\left) ... \right)
```

then the size of the parentheses is chosen to match the subformula that they enclose. Next note the use of the alignment tab character `&` to separate the entries of the matrix and the use of `\\` to separate the rows of the matrix, exactly as in the construction of multiline formulae described above. We begin the array with `\begin{array}` and end it with `\end{array}`. The only thing left to explain, therefore, is the mysterious `{ccc}` which occurs immediately after `\begin{array}`. Now each of the `c`'s in `{ccc}` represents a column of the matrix and indicates that the entries of the column should be *centred*. If the `c` were replaced by `l` then the corresponding column would be typeset with all the entries flush *left*, and `r` would produce a column with all entries flush *right*. Thus

```
\[ \begin{array}{lcr}
\mbox{First number} & x & 8 \\
\mbox{Second number} & y & 15 \\
\mbox{Sum} & x + y & 23 \\
\mbox{Difference} & x - y & -7 \\
\mbox{Product} & xy & 120 \end{array}\]
```

produces

$$\begin{array}{lcr}
\mbox{First number} & x & 8 \\
\mbox{Second number} & y & 15 \\
\mbox{Sum} & x + y & 23 \\
\mbox{Difference} & x - y & -7 \\
\mbox{Product} & xy & 120 \end{array}$$

We can use the array environment to produce formulae such as

$$|x| = \left\{ \begin{array}{ll} x & \mbox{if $x \geq 0$;}\\ -x & \mbox{if $x < 0$.}\end{array} \right.$$

Note that both columns of this array are set flush left. Thus we use {ll} immediately after \begin{array}. The large curly bracket is produced using \left\{. However this requires a corresponding \right delimiter to match it. We therefore use the *null delimiter* \right. discussed earlier. This delimiter is invisible. We can therefore obtain the above formula by typing

```
\[ |x| = \left\{ \begin{array}{ll}
         x & \mbox{if $x \geq 0$};\\
        -x & \mbox{if $x < 0$}.\end{array} \right. \]
```

3.15 Derivatives, Limits, Sums and Integrals

The expressions

$$\frac{du}{dt} \text{ and } \frac{d^2 u}{dx^2}$$

are obtained in LaTeX by typing \frac{du}{dt} and \frac{d^2 u}{dx^2} respectively. The mathematical symbol ∂ is produced using \partial. Thus the Heat Equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}$$

is obtained in LaTeX by typing

```
\[\frac{\partial u}{\partial t}
   = \frac{\partial^2 u}{\partial x^2}
      + \frac{\partial^2 u}{\partial y^2}
      + \frac{\partial^2 u}{\partial z^2} \]
```

To obtain mathematical expressions such as

$$\lim_{x \to +\infty}, \inf_{x > s} \text{ and } \sup_K$$

in displayed equations we type \lim_{x \to +\infty}, \inf_{x > s} and \sup_K respectively. Thus to obtain

$$\lim_{x \to 0} \frac{3x^2 + 7}{x^2 + 1} = 3.$$

(in LaTeX) we type

```
\[ \lim_{x \to 0} \frac{3x^2 +7x^3}{x^2 +5x^4} = 3.\]
```

To obtain a summation sign such as

$$\sum_{i=1}^{2n}$$

we type `sum_{i=1}^{2n}`. Thus

$$\sum_{k=1}^n k^2 = \frac{1}{2} n (n+1).$$

is obtained by typing

```
\[ \sum_{k=1}^n k^2 = \frac{1}{2} n (n+1).\]
```

We now discuss how to obtain *integrals* in mathematical documents. A typical integral is the following:

$$\int_a^b f(x)\,dx.$$

This is typeset using

```
\[ \int_a^b f(x)\,dx.\]
```

The integral sign $\int$ is typeset using the control sequence `\int`, and the *limits of integration* (in this case a and b are treated as a subscript and a superscript on the integral sign.

Most integrals occurring in mathematical documents begin with an integral sign and contain one or more instances of d followed by another (Latin or Greek) letter, as in dx, dy and dt. To obtain the correct appearance one should put extra space before the d, using `\,`. Thus

$$\int_0^{+\infty} x^n e^{-x} \,dx = n!.$$

$$\int \cos \theta \,d\theta = \sin \theta.$$

$$\int_{x^2 + y^2 \leq R^2} f(x,y)\,dx\,dy = \int_{\theta=0}^{2\pi} \int_{r=0}^R f(r\cos\theta, r\sin\theta) r\,dr\,d\theta.$$

and

$$\int_0^R \frac{2x\,dx}{1+x^2} = \log(1+R^2).$$

are obtained by typing

```
\[ \int_0^{+\infty} x^n e^{-x} \,dx = n!.\]

\[ \int \cos \theta \,d\theta = \sin \theta.\]
```

```
\[ \int_{x^2 + y^2 \leq R^2} f(x,y)\,dx\,dy
   = \int_{\theta=0}^{2\pi} \int_{r=0}^R
      f(r\cos\theta,r\sin\theta) r\,dr\,d\theta.\]
```

and

```
\[ \int_0^R \frac{2x\,dx}{1+x^2} = \log(1+R^2).\]
```

respectively.

In some multiple integrals (i.e., integrals containing more than one integral sign) one finds that LaTeX puts too much space between the integral signs. The way to improve the appearance of of the integral is to use the control sequence `\!` to remove a thin strip of unwanted space. Thus, for example, the multiple integral

$$\int_0^1 \! \int_0^1 x^2 y^2\,dx\,dy.$$

is obtained by typing

```
\[ \int_0^1 \! \int_0^1 x^2 y^2\,dx\,dy.\]
```

Had we typed

```
\[ \int_0^1 \int_0^1 x^2 y^2\,dx\,dy.\]
```

we would have obtained

$$\int_0^1 \int_0^1 x^2 y^2\,dx\,dy.$$

A particularly noteworthy example comes when we are typesetting a multiple integral such as

$$\int \!\!\! \int_D f(x,y)\,dx\,dy.$$

Here we use `\!` three times to obtain suitable spacing between the integral signs. We typeset this integral using

```
\[ \int \!\!\! \int_D f(x,y)\,dx\,dy.\]
```

Had we typed

```
\[ \int \int_D f(x,y)\,dx\,dy.\]
```

we would have obtained

$$\int \int_D f(x,y)\,dx\,dy.$$

The following (reasonably complicated) passage exhibits a number of the features which we have been discussing:

In non-relativistic wave mechanics, the wave function $\psi(\mathbf{r}, t)$ of a particle satisfies the *Schrödinger Wave Equation*

$$i\hbar\frac{\partial \psi}{\partial t} = \frac{-\hbar^2}{2m} \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) \psi + V \psi.$$

It is customary to normalize the wave equation by demanding that

$$\int \!\!\! \int \!\!\! \int_{\textbf{R}^3} \left| \psi(\mathbf{r},0) \right|^2\,dx\,dy\,dz = 1.$$

A simple calculation using the Schrödinger wave equation shows that

$$\frac{d}{dt} \int \!\!\! \int \!\!\! \int_{\textbf{R}^3} \left| \psi(\mathbf{r},t) \right|^2\,dx\,dy\,dz = 0,$$

and hence

$$\int \!\!\! \int \!\!\! \int_{\textbf{R}^3} \left| \psi(\mathbf{r},t) \right|^2\,dx\,dy\,dz = 1$$

for all times t. If we normalize the wave function in this way then, for any (measurable) subset V of $\mathbf{R}^3$ and time t,

$$\int \!\!\! \int \!\!\! \int_{V} \left| \psi(\mathbf{r},t) \right|^2\,dx\,dy\,dz$$

represents the probability that the particle is to be found within the region V at time t.

One would typeset this in LaTeX by typing

```
In non-relativistic wave mechanics, the wave function
$\psi(\mathbf{r},t)$ of a particle satisfies the
\textit{Schr\"{o}dinger Wave Equation}
\[ i\hbar\frac{\partial \psi}{\partial t}
  = \frac{-\hbar^2}{2m} \left(
    \frac{\partial^2}{\partial x^2}
    + \frac{\partial^2}{\partial y^2}
    + \frac{\partial^2}{\partial z^2}
  \right) \psi + V \psi.\]
It is customary to normalize the wave equation by
demanding that
\[ \int \!\!\! \int \!\!\! \int_{\textbf{R}^3}
      \left| \psi(\mathbf{r},0) \right|^2\,dx\,dy\,dz = 1.\]
A simple calculation using the Schr\"{o}dinger wave
equation shows that
```

```
\[ \frac{d}{dt} \int \!\!\! \int \!\!\! \int_{\textbf{R}^3}
      \left| \psi(\mathbf{r},t) \right|^2\,dx\,dy\,dz = 0,\]
and hence
\[ \int \!\!\! \int \!\!\! \int_{\textbf{R}^3}
      \left| \psi(\mathbf{r},t) \right|^2\,dx\,dy\,dz = 1\]
for all times~$t$. If we normalize the wave function in this
way then, for any (measurable) subset~$V$ of $\textbf{R}^3$
and time~$t$,
\[ \int \!\!\! \int \!\!\! \int_V
      \left| \psi(\mathbf{r},t) \right|^2\,dx\,dy\,dz\]
represents the probability that the particle is to be found
within the region~$V$ at time~$t$.
```

4 Further Features of LaTeX

4.1 Producing White Space in LaTeX

To produce (horizontal) blank space within a paragraph, use `\hspace`, followed by the length of the blank space enclosed within curly brackets. The length of the skip should be expressed in a unit recognized by LaTeX. These recognized units are given in the following table:

`pt`	point	(1 in = 72.27 pt)
`pc`	pica	(1 pc = 12 pt)
`in`	inch	(1 in = 25.4 mm)
`bp`	big point	(1 in = 72 bp)
`cm`	centimetre	(1 cm = 10 mm)
`mm`	millimetre	
`dd`	didot point	(1157 dd = 1238 pt)
`cc`	cicero	(1 cc = 12 dd)
`sp`	scaled point	(65536 sp = 1 pt)

Thus to produce a horizontal blank space of 20 mm in the middle of a paragraph one would type `\hspace{20 mm}`.

If LaTeX decides to break between lines at a point in the document where an `\hspace` is specified, then no white space is produced. To ensure that white space is produced even at points in the document where line breaking takes place, one should replace `\hspace` by `\hspace*`

To produce (vertical) blank space between paragraphs, use `\vspace`, followed by the length of the blank space enclosed within curly brackets. Thus to obtain

This is the first paragraph of some text. It is separated from the second paragraph by a vertical skip of 10 millimetres.

This is the second paragraph.

one should type

```
This is the first paragraph of some text.  It is
separated from the second paragraph by a vertical skip of
10 millimetres.

\vspace{10 mm}
This is the second paragraph.
```

If LaTeX decides to introduce at a point in the document where a `\vspace` is specified, then no white space is produced. To ensure that white space is produced even at points in the document where page breaking takes place, one should replace `\vspace` by `\vspace*`

We now describe certain features of LaTeX relating to blank spaces and paragraph indentation which will improve the appearance of the final document. Experienced users of LaTeX will improve the appearance of their documents if they bear these remarks in mind.

First note that, as a general rule, you should never put a blank space after a left parenthesis or before a right parenthesis. If you were to put a blank space in these places, then you run the risk that LaTeX might start a new line immediately after the left parenthesis or before the right parenthesis, leaving the parenthesis marooned at the beginning or end of a line.

LaTeX has its own rules for deciding the lengths of blank spaces. For instance, LaTeX will put an extra amount of space after a full stop if it considers that the full stop marks the end of a sentence.

The rule adopted by LaTeX is to regard a period (full stop) as the end of a sentence if it is preceded by a lowercase letter. If the period is preceded by an uppercase letter then LaTeX assumes that it is not a full stop but follows the initials of somebody's name.

This works very well in most cases. However LaTeX occasionally gets things wrong. This happens with a number of common abbreviations (as in 'Mr. Smith' or in 'etc.'), and, in particular, in the names of journals given in abbreviated form (e.g., 'Proc. Amer. Math. Soc.'). The way to overcome this problem is to put a backslash before the blank space in question. Thus we should type

```
Mr.\ Smith
etc.\ and
Proc.\ Amer.\ Math.\ Soc.
```

LaTeX determines itself how to break up a paragraph into lines, and will occasionally hyphenate long words where this is desirable. However it is sometimes necessary to tell LaTeX not to break at a particular blank space. The special character used for this purpose is `~`. It represents a blank space at which LaTeX is not allowed to break between lines. It is often desirable to use `~` in names where the forenames are represented by initials. Thus to obtain 'W. R. Hamilton' it is best to type `W.~R.~Hamilton`. It is also desirable in phrases like 'Example 7' and 'the length l of the rod', obtained by typing `Example~7` and `the length~$l$ of the rod`.

LaTeX will automatically indent paragraphs (with the exception of the first paragraph of a new section). One can prevent LaTeX from indenting a paragraph though by beginning the paragraph with the control sequence `\noindent`. Thus one obtains

> This is the beginning of a paragraph which is not indented in the usual way. This has been achieved by placing an appropriate control sequence at the beginning of the paragraph.

by typing

```
\noindent
This is the beginning of a paragraph which is not
indented in the usual way.  This has been achieved
by placing an appropriate control sequence at the
beginning of the paragraph.
```

Conversely, the control sequence `\indent` forces LaTeX to indent the paragraph.

4.2 Lists

LaTeX provides the following list environments:

- `enumerate` for numbered lists,
- `itemize` for un-numbered lists,
- `description` for description lists

Numbered lists are produced using

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

The items in the list should be enclosed between

```
\begin{enumerate} and \end{enumerate}
```

and should each be preceded by the control sequence `\item` (which will automatically generate the number labelling the item). For example, the text

> A *metric space* (X, d) consists of a set X on which is defined a *distance function* which assigns to each pair of points of X a distance between them, and which satisfies the following four axioms:
>
> 1. $d(x, y) \geq 0$ for all points x and y of X;
> 2. $d(x, y) = d(y, x)$ for all points x and y of X;
> 3. $d(x, z) \leq d(x, y) + d(y, z)$ for all points x, y and z of X;
> 4. $d(x, y) = 0$ if and only if the points x and y coincide.

is generated by LaTeX from the following input:

```
A \emph{metric space} $(X,d)$ consists of a set~$X$ on
which is defined a \emph{distance function} which assigns
to each pair of points of $X$ a distance between them,
and which satisfies the following four axioms:
\begin{enumerate}
\item
$d(x,y) \geq 0$ for all points $x$ and $y$ of $X$;
\item
$d(x,y) = d(y,x)$ for all points $x$ and $y$ of $X$;
\item
$d(x,z) \leq d(x,y) + d(y,z)$ for all points $x$, $y$
and $z$ of $X$;
\item
$d(x,y) = 0$ if and only if the points $x$ and $y$
coincide.
\end{enumerate}
```

Un-numbered lists are produced using

```
\begin{itemize} ... \end{itemize}
```

If we replace

```
\begin{enumerate} and \end{enumerate}
```

in the above input by

```
\begin{itemize} and \end{itemize}
```

respectively, LaTeX generates an itemized list in which each item is preceeded by a 'bullet':

> A *metric space* (X, d) consists of a set X on which is defined a *distance function* which assigns to each pair of points of X a distance between them, and which satisfies the following four axioms:
>
> - $d(x, y) \geq 0$ for all points x and y of X;
> - $d(x, y) = d(y, x)$ for all points x and y of X;
> - $d(x, z) \leq d(x, y) + d(y, z)$ for all points x, y and z of X;
> - $d(x, y) = 0$ if and only if the points x and y coincide.

Description lists (for glossaries etc.) are produced using

```
\begin{description} ... \end{description}
```

The items in the list should be enclosed between

`\begin{description}` and `\end{description}`

and should each be preceded by `\item[`*label*`]`, where *label* is the label to be assigned to each item. For example, the text

> We now list the definitions of *open ball*, *open set* and *closed set* in a metric space.
>
> **open ball** The *open ball* of radius r about any point x is the set of all points of the metric space whose distance from x is strictly less than r;
>
> **open set** A subset of a metric space is an *open set* if, given any point of the set, some open ball of sufficiently small radius about that point is contained wholly within the set;
>
> **closed set** A subset of a metric space is a *closed set* if its complement is an open set.

is generated by LaTeX from the following input:

```
We now list the definitions of \emph{open ball},
\emph{open set} and \emph{closed set} in a metric space.
\begin{description}
\item[open ball]
The \emph{open ball} of radius~$r$ about any point~$x$
is the set of all points of the metric space whose
distance from $x$ is strictly less than $r$;
```

```
\item[open set]
A subset of a metric space is an \emph{open set} if,
given any point of the set, some open ball of
sufficiently small radius about that point is contained
wholly within the set;
\item[closed set]
A subset of a metric space is a \emph{closed set} if its
complement is an open set.
\end{description}
```

4.3 Displayed Quotations

Displayed quotations can be embedded in text using the **quote** and **quotation** environments

```
\begin{quote} ... \end{quote}
```

and

```
\begin{quotation} ... \end{quotation}.
```

The **quote** environment is recommended for short quotations: the whole quotation is indended in the **quote** environment, but the first lines of individual paragraphs are not further indented. The input file

```
Isaac Newton discovered the basic techiques of
the differential and integral calculus, and
applied them in the study of many problems
in mathematical physics.  His main mathematical
works are the \emph{Principia} and the \emph{Optics}.
He summed up his own estimate of his work as follows:
\begin{quote}
I do not know what I may appear to the world; but to
myself I seem to have been only like a boy, playing
on the sea-shore, and diverting myself, in now and
then finding a smoother pebble, or a prettier shell
than ordinary, whilst the great ocean of truth lay
all undiscovered before me.
\end{quote}
In later years Newton became embroiled in a bitter
priority dispute with Leibniz over the discovery
of the basic techniques of calculus.
```

is typeset by LaTeX as follows:

> Isaac Newton discovered the basic techiques of the differential and integral calculus, and applied them in the study of many problems in mathematical physics. His main mathematical works are the *Principia* and the *Optics*. He summed up his own estimate of his work as follows:
>
>> I do not know what I may appear to the world; but to myself I seem to have been only like a boy, playing on the sea-shore, and diverting myself, in now and then finding a smoother pebble, or a prettier shell than ordinary, whilst the great ocean of truth lay all undiscovered before me.
>
> In later years Newton became embroiled in a bitter priority dispute with Leibniz over the discovery of the basic techniques of calculus.

For longer quotations one may use the **quotation** environment: the whole quotation is indented, and the openings of paragraphs are then further indented in the normal fashion.

4.4 Tables

Tables can be produced in LaTeX using the **tabular** environment. For example, the text

> The first five International Congresses of Mathematicians were held in the following cities:
>
> | | | |
> |---|---|---|
> | Chicago | U.S.A. | 1893 |
> | Zürich | Switzerland | 1897 |
> | Paris | France | 1900 |
> | Heidelberg | Germany | 1904 |
> | Rome | Italy | 1908 |

is produced in LaTeXusing the following input file:

```
The first five International Congresses of Mathematicians
were held in the following cities:
\begin{quote}
\begin{tabular}{lll}
Chicago&U.S.A.&1893\\
Z\"{u}rich&Switzerland&1897\\
Paris&France&1900\\
Heidelberg&Germany&1904\\
Rome&Italy&1908
```

```
\end{tabular}
\end{quote}
```

The `\begin{tabular}` command must be followed by a string of characters enclosed within braces which specifies the format of the table. In the above example, the string `{lll}` is a format specification for a table with three columns of left-justified text. Within the body of the table the ampersand character `&` is used to separate columns of text within each row, and the double backslash `\\` is used to separate the rows of the table.

The next example shows how to obtain a table with vertical and horizontal lines. The text

> The group of permutations of a set of n elements has order $n!$, where $n!$, the factorial of n, is the product of all integers between 1 and n. The following table lists the values of the factorial of each integer n between 1 and 10:
>
> | n | $n!$ |
> |---:|---:|
> | 1 | 1 |
> | 2 | 2 |
> | 3 | 6 |
> | 4 | 24 |
> | 5 | 120 |
> | 6 | 720 |
> | 7 | 5040 |
> | 8 | 40320 |
> | 9 | 362880 |
> | 10 | 3628800 |
>
> Note how rapidly the value of $n!$ increases with n.

is produced in LaTeXusing the following input file:

```
The group of permutations of a set of $n$~elements has
order $n!$, where $n!$, the factorial of $n$, is the
product of all integers between $1$ and $n$.  The
following table lists the values of the factorial of each
integer~$n$ between 1 and 10:
\begin{quote}
\begin{tabular}{|r|r|}
\hline
$n$&$n!$\\
\hline
1&1\\
```

```
2&2\\
3&6\\
4&24\\
5&120\\
6&720\\
7&5040\\
8&40320\\
9&362880\\
10&3628800\\
\hline
\end{tabular}
\end{quote}
Note how rapidly the value of $n!$ increases with $n$.
```

In this example the format specification `{|r|r|}` after `\begin{tabular}` specifies that the table should consist of two columns of right-justified text, with vertical lines to the left and to the right of the table, and between columns.

Within the body of the table, the command `\hline` produces a horizontal line; this command can only be placed between the format specification and the body of the table (to produce a line along the top of the table) or immediately after a row separator (to produce a horizontal line between rows or at the bottom of the table).

In a **tabular** environment, the format specification after `\begin{tabular}` should consist of one or more of the following, enclosed within braces `{` and `}`:

`l`	specifies a column of left-justified text
`c`	specifies a column of centred text
`r`	specifies a column of right-justified text
`p{`*width*`}`	specifies a left-justified column of the given width
`|`	inserts a vertical line between columns
`@{`*text*`}`	inserts the given *text* between columns

A string *str* of characters in the format specification can be repeated *num* times using the construction `*{`*num*`}{`*str*`}`. For example, a table with 15 columns of right-justified text enclosed within vertical lines can be produced using the format specification `{|*{15}{r|}}`.

If additional vertical space is required between rows of the table, then this can be produced by specifying the amount of space within square brackets after `\\`. For example, on would use `\\[6pt]` to separate two rows of the table by 6 points of blank space.

A horizontal line in a table from column i to column j inclusive can be produced using `\cline{`i`-`j`}`. For example `\cline{3-5}` produces a horizontal line spanning columns 3, 4 and 5 of some table.

A command of the form `\multicolumn{`*num*`}{`*fmt*`}{`*text*`}` can be used within the body of a table to produce an entry spanning several columns. Here *num* specifies the number of columns to be spanned, *fmt* specifies the format for the entry (e.g., `l` if the entry is to be left-justified entry, or `c` if the entry is to be centred), and *text* is the text of the entry. For example, to span three columns of a table with the words 'Year of Entry' (centred with respect to the three columns), one would use

```
\multicolumn{3}{c}{Year of entry}
```

4.5 The Preamble of the LaTeX Input file

We describe the options available in LaTeX for specifying the overall style of a document.

A LaTeX document should begin with a `\documentclass` command and any text to be printed must be included between

`\begin{document}` and `\end{document}`

commands. The `\begin{document}` command is sometimes preceded by commands that set the page-style and set up user-defined control sequences.

Here is a typical LaTeX input file:

```
\documentclass[a4paper,12pt]{article}
\begin{document}

This is the first paragraph of a typical document. It is
produced in a '12~point' size. A \textit{point} is a unit
of length used by printers. One point is approximately
$1/72$~inch. In a '12~point' font the height of the
parentheses is 12~points (i.e. about $1/6$~inch) and the
letter~'m' is about 12 points long.

This is the second paragraph of the document. There are
also '10 point' and '11 point' styles available in \LaTeX.
The required size is specified in the 'documentstyle'
command. If no such size is specified then the 10~point
size is assumed.

\end{document}
```

The syntax of the `\documentclass` command is as follows. The command begins with `\documentclass` and ends with the names of one of the available styles, enclosed in curly brackets. The available styles are `article`, `report`, `book` and `letter`. Between the "`\documentclass`" and the name of the document style, one may place a list of *options*. These options are separated by commas and the list of options is enclosed in square brackets (as in the above example). The options available (which are usually the names of certain 'style files') include the following:

11pt Specifies a size of type known as *eleven-point*, which is ten percent larger than the ten-point type normally used.

12pt Specifies a twelve-point type size, which is twenty percent larger than ten-point.

twocolumn Produces two-column output.

a4paper This ensures that the page is appropriately positioned on A4 size paper.

Typing simply `\documentclass{article}` will produce a document in ten-point type size. However the printed output will not be nicely positioned on A4 paper, since the default size is intended for a different (American) paper size.

Pages will be automatically numbered at the bottom of the page, unless you specify otherwise. This can be done using the `\pagestyle` command. This command should come after the `\documentclass` command and before the `\begin{document}` command. This command has the syntax `\pagestyle{`*option*`}`, where the *option* is one of the following:

plain The page number is at the foot of the page. This is the default page style for the `article` and `report` document styles.

empty No page number is printed.

headings The page number (and any other information determined by the document style) is put at the top of the page.

myheadings Similar to the **headings** pagestyle, except that the material to go at the top of the page is determined by `\markboth` and `\markright` commands (see the LaTeX manual).

For example, the input file

```
\documentclass[a4paper]{article}
\pagestyle{empty}
\begin{document}
The main body of the document is placed here.
\end{document}
```

produces a document without page numbers, using the standard ten-point type size.

4.6 Defining your own Control Sequences in LaTeX

Suppose that we are producing a paper that makes frequent use of some mathematical expression. For example, suppose that integrals like

$$\int_{-\infty}^{+\infty} f(x)\,dx.$$

occur frequently throughout the text. This formula is obtained by typing

```
\[ \int_{-\infty}^{+\infty} f(x)\,dx.\]
```

It would be nice if we could type `\inftyint` (say) to obtain the integral sign at the beginning. This can be done using `\newcommand`. What we do is to place a line with the command

```
\newcommand{\inftyint}{\int_{-\infty}^{+\infty}}
```

near the beginning of the input file (e.g., after the `\documentclass` command but before the `\begin{document}` command). Then we only have to type

```
\[ \inftyint f(x)\,dx.\]
```

to obtain the above formula.

We can modify this procedure slightly. Suppose that we we defined a new control sequence `\intwrtx` by putting the line

```
\newcommand{\intwrtx}[1]{\int_{-\infty}^{+\infty} #1 \,dx}
```

at the beginning of the input file. If we then type the line

```
\[ \intwrtx{f(x)}.\]
```

then we obtain

$$\int_{-\infty}^{+\infty} f(x)\,dx.$$

What has happened is that the expression in curly brackets after `\intwrtx` has been substituted in the expression defining `\intwrtx`, replacing the `#1` in that expression.

The number 1 inside square brackets in the `\newcommand` line defining `\intwrtx` indicates to LaTeX that it is to expect one expression (in curly brackets) after `\intwrtx` to substitute for `#1` in the definition of `\intwrtx`. If we defined a control sequence `\intwrt` by

```
\newcommand{\intwrt}[2]{\int_{-\infty}^{+\infty} #2 \,d #1}
```

then it would expect two expressions to substitute in for `#1` and `#2` in the definition of `\intwrt`. Thus if we then type

```
\[ \intwrt{y}{f(y)}.\]
```

we obtain

$$\int_{-\infty}^{+\infty} f(y)\,dy.$$