Contents

1 Introduction. .................................................. 3

2 The Basics of Commands. .................................. 3
  2.1 Spaces. .................................................. 4
  2.2 Hyphens. .................................................. 5
  2.3 Quotes. .................................................. 5

3 The Structure of a \LaTeX Document. .................. 6
  3.1 Popular Class Options. .................................. 6
  3.2 Other Options. .......................................... 6
  3.3 Page Styles. ............................................. 6
  3.4 Page Numbering. ........................................ 7
  3.5 Page Layout. ............................................ 7
  3.6 Preparing a Title. ...................................... 8

4 The Body. .................................................. 9
  4.1 Fonts — Sizes, Shapes and Series. .................... 9
  4.2 Line and Page Breaking. ................................ 10
  4.3 Introductory Environments. ............................ 10
    4.3.1 Center. ............................................ 10
    4.3.2 Lists. ............................................. 11
    4.3.3 Tables. ........................................... 12
  4.4 Miscellaneous. ......................................... 13
    4.4.1 Labels ............................................. 13
    4.4.2 Footnotes ......................................... 13
    4.4.3 The Package \url. ................................ 14

5 Typesetting Mathematics. ................................. 14
  5.1Fractions/Binomials, Modulo, and Roots. ............ 15
  5.2 The \amsymb Package .................................. 15
  5.3 The \amsmath Package ................................ 15
  5.4 The \amsthm Package .................................. 18
  5.5 The \amsmath Package Continued ...................... 20

6 Bibliography. ............................................. 23
1 Introduction.

A brief history will give the reader some insight into why \LaTeX is designed the way it is. In the late 1970s Donald Knuth, of Stanford University, was motivated to use the rising power of computing to produce professional typesetting including the very technical kind needed for mathematics, engineering and science. The result is called \TeX. The \TeX user creates a source file that contains text and commands using only the symbols that appear on a standard typewriter keyboard. The commands tell the program what symbols to produce. The file is then compiled and a .dvi file or now, more commonly, a .pdf file is produced that can be printed. Using commands rather than keystrokes permit an essentially unlimited number of symbols. And indeed new commands are still being added. After a decade or so of adjusting and expanding the program, Knuth announced that he would not make any more changes.

During that time Leslie Lamport was developing a companion to \TeX that he called \LaTeX which expanded and simplified the use of \TeX as a method to prepare documents. It is the current status of this project that we will study. Using \LaTeX it’s possible to create an entire document that incorporates the rules of standard publishing without knowing any of these rules. The first version was called version 2.09 (for an unknown reason) but a serious problem developed. As changes and additions were made, users were forced to continuously reinstall the program. Failure to do so might mean that they wouldn’t be able to compile a file created using the latest version. To solve this problem the \LaTeX2e project was launched. The idea was to create a version of \LaTeX (called the kernel) that wouldn’t be altered. Any additions were to be done using “packages”. Just how these packages are used will be explained as we work through the structure of a \LaTeX document.

2 The Basics of Commands.

There are three types of commands. The first type consists of the typewriter keys #, $, &, ~, %, _, ^, {}, \, and \%. You may already be familiar with the use of _ to denote subscript and ^ for superscript. The symbol \ indicates the start of a command. The use of each of these key strokes as a command, except for #, will be illustrated in these notes. For example the start of a short mathematical expression begins with a $ sign and ends with a second $. So to produce $f'(x) = x$, type $f'(x)=x$. Note the difference in the font used in mathematical expressions. This change is part of what’s called, “math mode”. Another feature of math mode is that all space bar key strokes are ignored. Consequently typing \$ f' (x) = x \$$ produces the same output. To insert text in the document as a note, such as, “Insert the extra text here”, start a new line with the return key and type \%Insert the extra text here followed by another return key. The line \%Insert the extra text here appears on one line and all text following the % is ignored by \TeX.

The second type of commands are those starting with a \ followed by a
non-letter character. For example the command \$ produces the dollar sign, $.

The other symbols in the list of the preceding paragraph, except for ^, ~ and \ are produced in a similar fashion. To produce the exceptions, see the next two paragraphs. Other examples of commands of this second type are illustrated in the subsection, "Spaces" below. In addition many accent marks are produced using a command of this type. In particular, the accented á is produced by typing \‘a.

The third type of commands are those that begin with a \ followed be a string of letters and letters only. In particular, a space can’t be part of such a command. Any non-letter character indicates the end of the command. These commands are case-sensitive. An example of such a command is \dots, which is used in such expressions as 1, 2, . . . , n. Do not produce these dots with a string of 3 periods. A second example is the command, $\backslash$, which produces \. Note that this command works only in math mode. The same is true of \_ and ^.

Some commands of the second and third types require one or more arguments some of which may be optional. Mandatory arguments are indicated by \{text\} while the optional ones are indicated by [text]. For example the first command that one must put into a \LaTeX{} document is \documentclass{class}. As a second example, typing $\tilde{}$ produces the tilde, ¨. The reason for the pair of braces, {}, is that the tilde usually goes over a symbol. That symbol is typed between the braces. The situation is the same for ^.

Often more than one mandatory argument is needed. For example to create your own command, type \newcommand{\st}{such that}. It will produce the text, “such that”, any time \st is typed. We will encounter many, many more commands as we go along. For the two special commands, _ and ^ arguments are sometimes necessary but not always. For example to produce \(x_1\) it’s suffices to type $x_1$, but to produce \(x_{n+1}\) it’s necessary to type $x_{\{n+1\}}$.

### 2.1 Spaces.

In \LaTeX{} there are two types of horizontal spaces: between words space and between sentences space. Between words space is smaller than between sentences space and they are both “rubber”; that is, they can be changed in order to “justify” the text. Having two different types of horizontal space can produce problems. \LaTeX{} interprets the space following a period as between sentences space if the letter before the period is a lower case letter. If the period is preceded by an upper case letter, then \LaTeX{} thinks it’s just the end of an abbreviation and inserts between words space. So if a sentence ends with an upper case letter, such as, “I love MSU.”, tell \LaTeX{} that the period is the end of a sentence by typing \texttt{MSU@}. If a abbreviation ends in a lower case, type \texttt{\@} to tell \LaTeX{} that it should insert just between words space. Another situation of this type is Dr. Smith. To be sure between words space is inserted, type Dr. “Smith which not only inserts between words space but also tells \LaTeX{} not to break the line after Dr. For that reason the “ is often referred to as a sticky space.

The command \;, produces a very small amount of horizontal space. It’s
used in integration formulas to put a small amount of space between the function and the differential. For example, to produce $\int f(x) \, dx$ type $\int f(x) \, dx$.

(Note the use of the $ symbol to indicate the beginning and end of a mathematical expression and the command \int to produce the integral sign.) Other standard horizontal spaces commands are \: for a full space, \; for a half space, \! for a negative half space, \quad for 10 points if the font size selected is 10pt and \qquad for twice as much.

Two consecutive hits of the space bar don’t insert more space that one hit. For that reason \TeX inserts horizontal space with the command \hspace{length} where the length must be given units of some kind: in, pt, mm, cm for example. For horizontal space at the beginning of a line use \hspace*{length}. It should be noted at one hit of the return key is the same as one hit of the space bar. (The command without the * is ignored at the beginning of a line.) The command \hfill fills in horizontal space. It can be used to evenly space say three entries along the top of a line such as

\begin{verbatim}
MTH 132 Section 23 Second Hour Exam November 26, 2007
\end{verbatim}

which can be produced by typing

\begin{verbatim}
MTH 132 Section 23\hfill Second Hour Exam\hfill November 26, 2007\hfill
\end{verbatim}

(The double \ is the new line command. Another form of it is \newline.)

As has already been observed, one strike of the return key is the same as one strike of the space bar. This feature makes it easy to find specific parts of a document, such as a short mathematical expression. Two strikes of the return key produces a new paragraph. More than two strikes of the return key is the same as two strikes. So vertical space is inserted with the command \vspace{length} and at the top of a page with \vspace*{length}. There are also rubber vertical space commands. They are: \smallskip, \medskip, \bigskip, and \vfill. The latter does for vertical space what \hfill does for horizontal space.

### 2.2 Hyphens.

The hyphen key produces can be used to produce four horizontal lines of different lengths. First is the usual hyphen obtained by striking the hyphen key once. In math mode striking the hyphen key once produces a minus sign which is slightly longer than a regular hyphen. In regular text mode two consecutive strikes of the hyphen are use to indicate a range of values. For example pages 4--9 is produced by typing 4--9. And finally three consecutive strikes of the hyphen key produces a dash. Specifically, “The Hyphen Key—A Versatile Character”.

### 2.3 Quotes.

Quotation marks are often wrongly typed because on a typewriter there is only one key stroke to enter either a left quotation mark or a right quotation mark. But, as you know, in printing there’s a difference between opening quotation marks and closing ones. To get the opening quotation marks, hit the ‘ key
twice, (the lower case of the tilde key) and for right quotation marks, strike the ' key twice (the lower case of the quotation marks key).

3 The Structure of a $\LaTeX$ Document.

A $\LaTeX$ document consists of a preamble and the body of the document. Usually, but not always, the document ends with a bibliography. We begin with the preamble. Every $\LaTeX$ document begins with the command \documentclass{class name}. The most common choices for class are: article, slides, book, report, and letter. They are listed in the order in which you are likely to use them. Most of the documents you will produce will be done in article class. The class slides should be considered if you are going to present material in a room having more than 5 rows. It automatically uses large letters that are easy to read from a distance. The \documentclass command has an optional argument placed immediately after \documentclass, specifically \documentclass[option1, option2, ...]{class}.

3.1 Popular Class Options.

Except for slides, the default font size (size of the letters) is 10pt. There are two other options; namely, 11pt and 12pt. The paper size is also changed in the optional argument. The default setting is 8.5in $\times$ 11in, called letterpaper. Some other choices are legalpaper and the most common European one, a4paper. It's possible to type a document in two column form using the option, twocolumn.

3.2 Other Options.

Odd and even pages differ slightly in the book class to facilitate printing documents with an odd page and the following even page printed on the same sheet of paper. The same option is available in the article and report classes using the option, twoside. (The default is oneside.) See the discussion of page styles below.

When displaying a numbered mathematical formula, the equation number is placed to the right of the equation. To change it to the left side use the option leqno.

3.3 Page Styles.

The standard $\LaTeX$ page has, among other things, a header and a footer. The command \pagestyle{style} can be used to change the look of pages. The default page style for the article class is plain. It is described together with others in the following list.

plain The header is empty and the page number is in the center of the footer at the bottom of the page.

empty The header and footer are both empty. No page numbers are printed anywhere on the page.
The `headings` page style is used to display a heading at the top of each page. The header contains both the page number and a page title. If the `twoside` option is selected, the page number appears in the right side of the header on odd pages and in the left side of the header on even pages. In both cases, the heading is the current chapter title or the current section title.

The `myheading` page style is identical to the `headings` page style except the user selects the headings with the command `\markright{heading}` for the same heading on both odd and even pages or for different headings, `\markboth{odd\_page\_heading}{even\_page\_heading}`.

### 3.4 Page Numbering.

Page numbers may be displayed in any of three styles: arabic, roman, or alpha, with arabic being the default. The style is changed using the command `\pagenumbering{style}`. For example, `\pagenumbering{Roman}` numbers the pages using upper case Roman numerals. The actual page number (no matter in what style it is displayed) is set using the `counter` called page. (LATEX has a large number of counters. We will meet more of them later.) The value of that counter can be changed using the command, `\setcounter{page}{number}` or `\addtocounter{adjustment}{}`. One example where the ability to use different number styles and to adjust the corresponding counter is an article that begins with a preface that should be numbered in lowercase roman numerals. Immediately after the `\begin{document}` command, type `\pagenumbering{roman}` and then type `\section{Preface}` followed by the text of the preface. Once the preface is finished, type `\newpage` to begin a new page and then type the command `\pagenumbering{arabic}` followed by `\addtocounter{page}{-n}`, where `n` is the number of pages in the preface. The result will be that the next page to be printed will be numbered 1.

### 3.5 Page Layout.

Here is what a page in LATEX looks like.
For a one page document, such as an exam, it may be desirable to expand the Body of the document to include more of the margins. This is accomplished in much the same fashion that page numbering was changed. To increase the text height, first reduce the topmargin by typing \addtolength{\topmargin}{-2in}. Then type \addtolength{\textheight}{2in}. By subtracting the same amount that is added, the overall vertical space remains unchanged. Provided you haven’t selected the option twoside, to change the width first type the command \addtolength{\oddsidemargin}{-2in} followed by \addtolength{\textwidth}{2in}. (With twoside you must also subtract 2 inches from \evensidemargin.) With these two changes, the text will be 2 inches wider and 2 inches longer, but will be centered on the paper.

3.6 Preparing a Title.

You needn’t design your own title; \LaTeX{} has done that for you. You need only give \LaTeX{} what it needs to know using the following commands all of which go in the preamble.

1. \texttt{\textbackslash title\{The Title of Your Article\}}. Keep the title as brief and to the point as possible. Remember it may need to be printed in page headers along with the page number. If it must be longer, you will need to devise a shorter version for the headers and use the command \texttt{\textbackslash markboth} or \texttt{\textbackslash markright} to put the shortened version into the headers. For a long title, \LaTeX{} will automatically insert page breaks where needed, but you may select where line breaks occur using the command\textbackslash. Each line of
the title is centered on the page.

2. \author{author1,address1 \ and \ author2,address2}. The two author listings will be placed side-by-side underneath the title. If you prefer, you may have them appear one on top of the other by typing instead \author{author1,address1 \[2mm\]author2,address2}. This form is almost essential if there are three or more authors.

3. \date{date}. If this command is omitted, \TeX\ will enter the date you compiled the file using the command \today, which enters the date as it is found on your computer. To skip entering a date completely, type \date{}.

Once the preamble is finished it’s time to begin the document. Of course you may always return to the preamble to add more as we will discuss soon. The body of the document is begun by typing \begin{document}. To cause the title to be created, type \maketitle.

4 The Body.

4.1 Fonts — Sizes, Shapes and Series.

The standard font family used by \TeX\ for text in the body of a document is Roman. Also available without specifically loading them are the typewriter and the sans serif font families. (Sans is the French word meaning without and serifs are the short lines that appear in such letters at T which is sans serif is T.) The latter is the default in the slides documentclass and the typewriter font is used extensively in this document. All three have different sizes, shapes and series as discussed below. It is possible to use additional font families, but they must be in your computer’s collection of font families and need to be loaded with the \newfont command. There are two ways to change to a different font. For just a few words in a different font, use the declaration method whose syntax is \{\tt short text\} or for longer stretches of text use the command \textsf{the longer stretch of text}.

All three of these fonts are available in 10 different sizes. The default size is \normalsize.

\begin{itemize}
\item \tiny \hspace{2cm} smallest
\item \scriptsize \hspace{2cm} very small
\item \footnotesize \hspace{2cm} smaller
\item \small \hspace{2cm} small
\item \normalsize \hspace{2cm} normal
\item \large \hspace{2cm} large
\item \Large \hspace{2cm} larger
\item \LARGE \hspace{2cm} larger yet
\item \huge \hspace{2cm} still larger
\item \Huge \hspace{2cm} largest
\end{itemize}

For example to change a few words to a larger size for emphasis, type \{\LARGE Pay very close attention to this\} which will come out as Pay very close attention to this. These sizes are computed in proportion to the font size selected in \documentclass. Note that here the command \LARGE appears inside of the curly brackets. Such a form is called a declaration.
All three of the font families come in four shapes, the default shape being “upright”. The others are *italics*, slanted and SMALL CAPS. There are two ways to change the shape of a font; either with a declaration {\sc Short Text} to get SHORT TEXT or with a command \textit{a much longer stretch of text} to get *a much longer stretch of text*. Finally to produce text in boldface use either the declaration form \textbf{text} to produce SHORT TEXT or for longer text \textbf{much longer text} to produce **much longer text**. Size and shape attributes may be used together as can size and boldface, but shape and boldface can’t be combined.

### 4.2 Line and Page Breaking.

The command \newline will fill the remainder of the current line with space and start a new line. But a better way is with \ with an optional argument. Alone \ is identical to \newline but \[2mm] will fill the remainder of the current line with space, and then skip down 2mm before starting the next line. This feature is useful and can be employed anywhere the \ is used. It also has a cousin, \* which prevents a page break from occurring before starting the new line. The same optional argument is available for this form of the newline command.

A related command is \linebreak which has one optional argument. The major difference is that the remaining space in the current line is distributed between the words already in the line resulting in a line that ends at the right margin. If the remaining space is substantial, the result can be ugly. The optional argument is [imp-digit], where imp-digit is an integer between 0 and 4 establishing the importance of breaking the line at that point. A value of 0 means it’s not too important to break there, whereas \linebreak[4] means the break is mandatory; that is, it’s equivalent to \linebreak. The command \nolinebreak[imp-digit] works in exactly the opposite way to prevent a line break. There are other ways to prevent a line break. For example, using a “sticky space”, ~ as in Dr.~Smith. Sometimes it’s desirable to keep an entire string of words on one line. The easiest way to accomplish this is as follows. \mbox{the string of words to be kept on one line}. \LaTeX treats what’s in the brackets as one object rather than several objects.

The commands related to page breaks are \newpage, \pagebreak[imp-digit], and \nopagebreak[imp-digit]. These commands are analogous to the corresponding ones for line breaking.

### 4.3 Introductory Environments.

#### 4.3.1 Center.

Several constructions are accomplished with *environments*. Here we discuss only a few. The first, and perhaps the easiest, example is the *center environment*. To center text type \begin{center}
The text to be centered
\end{center}.
The amount of text can be longer than one line as in the following example.

This text is being centered on the page using the \textit{center environment} which
puts line breaks in automatically. But you may put them in where you wish
with the newline command \\.

4.3.2 Lists.

Next we introduce the list environments: itemize, enumerate and description.
All three have the same basic format but differ in what each produces. Itemize
and enumerate indent each item and labels each, while description does
not indent nor label. Itemize labels each item with a \textbullet while enumerate labels
items with numbers. Each is produced using essentially the same format.
\begin{list}{\textbullet}
\item The first item in the list.
\item The second item in the list.
\item The third item in the list.
\item The forth item in the list.
\item The fifth item in the list.
\end{list}

To demonstrate the difference the same list will be set in all three list envi-
ronments.

\begin{itemize}
\item The first item in the list.
\item The second item in the list.
\item The third item in the list.
\item The forth item in the list.
\item The fifth item in the list.
\end{itemize}

1. The first item in the list.
2. The second item in the list.
3. The third item in the list.
4. The forth item in the list.
5. The fifth item in the list.

The first item in the list.
The second item in the list.
The third item in the list.

The forth item in the list.

The fifth item in the list.

The command `\item` has an optional argument that is used to label items differently from the default manner. For example typing `\item[(a)] This item` produces, “(a) This item.”

A list may be included as an item in a list. The two lists may have the same or different list names. In the case of the same list, the second level labels are different from the first level. For example, in itemize the second level items are labeled —, the third level, * and the fourth level, ·.

### 4.3.3 Tables.

There are two methods for producing tables. The first is almost identical with the process of setting tab stops on a typewriter and is called the *tabbing environment*. Once the `\begin{tabbing}` command has been issued, the tabs are set with the command `\=` command and the end is designated by the usual new line command, `\`. To place the tabs, enter text as long or longer than the longest text to appear before the first tab and continue in the same fashion to set the remaining tabs. For example

```latex
\begin{tabbing}
text to the first tab set= shorter distance= now a longer distance to tab \kill
first item first line=>second item=>now the long item on the first line\\
first item second line=>second one=>here's the longest item allowed
\end{tabbing}
```

If you don’t want that text to actually appear in the document, replace the `\` with the command `\kill`. Then in the next line begin typing text. To go to the first tab, type `\>`. The same command moves to the next tab, but you must remember to insert `\` before starting the next line. For example here’s what to type to produce a table using the tabbing environment.

```latex
\begin{tabbing}
text to the first tab stop= shorter distance= now a longer distance to tab \kill
first item first line=>second item=>now the long item on the first line\\
first item second line=>second one=>here’s the longest item allowed
\end{tabbing}
```

First item first line second item now the long item on the first line
first item second line second one here’s the longest item allowed

Note that `\end{tabbing}` ends the last line. No `\` should appear.

The second method for producing tables is the *tabular environment*, which has two arguments; one optional and one mandatory. The syntax is `\begin{tabular}{[pos]}{cols}`. The mandatory argument sets the number of
columns that the table is to have and how the text is to be set in each. For example \{lcr\} indicates that there is to be three columns; the text in the first column is to be left aligned, in the second the text is to be centered and in the third the text is to be right aligned. To separate the columns by two vertical lines, enter \{l|c|r\}. The optional argument \[cols\] is used only in the case where the table is narrow and is to be contained within surrounding text. The default placement is centered on the baseline of the text where the table is inserted. Use \t to put the top of the table at the baseline of the text where the table is inserted and \b at the bottom of the baseline of that text. If the table is wide enough to stand alone without surrounding text, the optional argument has no purpose.

Once the number of columns has been determined and any line separation designated, the first row of the table is typed. The & character signifies the end of the first column of text. The next text goes in the second column. This process continues until the last column has been filled. The type \\ to indicate the start of the next row of the table. Rows of a table may be separated by a line using the command \hline after the \\ command. To produce the table below type

\begin{tabular}{|l||c|c|c|c|c|}
\hline
Name & Test 1 & Test 2 & Test 3 & Final & Grade \\
\hline
Alan Adams & 78 & 100 & 72 & 156 & 3.5 \\
Beverly Brown & 84 & 92 & 100 & 182 & 4.0 \\
Carl Clod & 62 & 51 & 39 & 82 & 0.0 \\
\hline
\end{tabular}

4.4 Miscellaneous.

4.4.1 Labels

Structures to which \LaTeX assigns a number can be labeled for reference later in the document. For example to refer to a specific section, type \section{Main Lemma}\label{ML} and then later when you wish to refer to the section, Main Lemma, type \texttt{in Section \ref{ML}} and \LaTeX will produce, “in Section n” where Main Lemma is the n\textsuperscript{th} section. Labeling will be very useful once numbered equations are introduced.

4.4.2 Footnotes

To enter a footnote about some word or phrase, immediately after the text to be footnoted, type \idfootnote \footnote{the text for the footnote}. \LaTeX will
draw a short line at the bottom of the page followed by the footnote number and footnote text. There is a variant of footnoting specifically designed for authors wishing to thank a funding agency or a person for support. Immediately after the author name type `\thanks{the gushing note of thanks}`. The thanks will appear as a footnote, but instead of a number, with a symbol, such as an *.

4.4.3 The Package url.

Entering internet and email addresses can be tricky because a hyphen is often part of such an address, so L\TeX must be prevented from breaking a line with a hyphen in an address. The package `url` is designed to do just that. In the preamble type \usepackage{url} and then when you wish to enter, say an internet address, type \url{http://www.ams.org/tex/amslatex.html} and the result will be the internet address http://www.ams.org/tex/amslatex.html. The spaces in the argument tell L\TeX where it is allowed to break the line.

5 Typesetting Mathematics.

\LaTeX employs a different font for characters used as mathematical symbols. So be certain that every symbol you enter that is to be mathematics is set in “math mode”. If a mathematical expression is to be included as part of the text in a line, it is enclosed between a pair of $ signs. For example to typeset the sentence, “The factoring of squares formula is \((a^2 - b^2) = (a - b)(a + b)\)” you type The factoring of squares formula is $(a^2 - b^2) = (a - b)(a + b)$.

(Spaces are ignored in math mode, so type as many as you need to make reading the source document as easy as possible.) Longer and more complicated expressions are displayed. Displayed expressions are enclosed between a pair of $$ signs, or inside of an environment. Use the first technique when the expression to be displayed is less than a line in length and you don’t wish to number it. For example type

The formula for the factoring of the difference of \(n\)^th powers is \[
\sum a^n - b^n = (a - b)(a^{n-1} + a^{n-2}b + a^{n-3}b^2 + \cdots + a^2b^{n-3} + ab^{n-2} + b^{n-1})
\]

to get,

The formula for the factoring of the difference of \(n\)^th powers is

\[
a^n - b^n = (a - b)(a^{n-1} + a^{n-2}b + a^{n-3}b^2 + \cdots + a^2b^{n-3} + ab^{n-2} + b^{n-1})
\]

(Note that exponents longer than one character must be contained in a pair of curly brackets, \{ \}). Some authors will type \cdots to get the dots in the center of the line, but L\TeX (with the amsmath package) knows from contents where to put them according to the convention adopted by the AMS.
5.1 Fractions/Binomials, Modulo, and Roots.

The commands for producing fractions and binomial expressions are similar. For fractions use the command \( \frac{\text{numerator}}{\text{denominator}} \). A larger font is used for displayed fractions than for fractions within text. If a fraction within text has expressions with exponents, the smaller font may be hard to read. In that case replace \( \frac{\text{numerator}}{\text{denominator}} \) with \( \dfrac{\text{numerator}}{\text{denominator}} \). Sometimes in displayed formulas a smaller fraction is desired. In that case type \( \tfrac{\text{numerator}}{\text{denominator}} \). For binomial expressions, the identical command structure is available beginning with \( \binom{\text{}}{\text{}} \).

The basic command for producing modulo expressions such as \( x \equiv y \mod{n} \) is \( x \equiv y \mod{n} \). The variation \( \bmod{n} \) diminishes the space before mod, \( x \equiv y \mod{n} \). Parentheses are placed around mod \( n \) with \( \pmod{\text{}} \) and \( \pod \) eliminates the word mod, \( x \equiv y \mod{n} \) and \( x \equiv y \mod{n} \).

Roots are created with the command \( \sqrt[n]{\text{expression}} \). For example \( \sqrt[3]{x^2+1} \) produces \( 3 \sqrt[3]{x^2+1} \) in text and \( 3 \sqrt[3]{x^2+1} \) in displayed formulas. There is no \( \dsqrt \) form as might be expected, but any mathematical expression in text can be given its displayed version with the command \( \text{displaystyle} \) which is a declaration type command. As would be expected there’s a corresponding declaration \( \text{textstyle} \) to produce text style in display mode.

5.2 The amssymb Package

Loading the amssymb package makes the Blackboard Bold and Euler Fraktur alphabets available in math mode. For example, typing \( \text{let } \mathfrak{c} \text{ denote the cardinality of } \mathbb{R} \) yields, “let \( c \) denote the cardinality of \( \mathbb{R} \).”

5.3 The amsmath Package.

Use this package when typing any mathematics that will require multi-line formulas or computations. It provides you with several environments to produce formulas: equation, align, gather, and multiline. Each of these numbers each line in the environment. To suppress the numbers, follow the name with a *; for example equation*. In addition the environment split is provided to be used inside of a number-producing environment. As with all environments, \LaTeX{} produces a small amount of (rubber) vertical space before beginning the environment and more after the environment ends. So adding vertical space should be avoided. For ease of editing the source document, always put the \begin{... command on a new line and start the environment text on the next line.

The equation environment produces a single line of mathematics which is numbered. For example

\[ \sum_{i=1}^{\infty} p(f(t_i)) \mu(\sigma_i) \leq \sum_{i=1}^{\infty} g(t_i) \mu(\sigma_i) \leq \int_{\Omega} p(f(t)) dt + \epsilon \]  \tag{1} 

is created by typing
\[ \sum_{i=1}^{\infty} p(f(t_i)) \mu(\sigma_i) \leq \sum_{i=1}^{\infty} g(t_i) \mu(\sigma_i) \leq \overline{\int_{\Omega}} p(f(t)) \, dt + \epsilon \]

To produce the same expression without the number, replace \texttt{equation} with \texttt{equation*}. You might question why use such a form when the double dollar signs would do the same thing. There is a situation when the environment must be used. To refer to a numbered equation later in the document, type \texttt{\begin{equation} \label{eq1} \end{equation}} and then typing \texttt{\eqref{eq1}} will produce (1). The \texttt{\eqref} puts in the parentheses.

For a calculation that requires several lines, use the \texttt{align} environment. (It replaces the \texttt{eqnarray} environment.) A simple example of its use is

\[
|f(x) - f(y)| = |h(x) - h(p^i_k)| \leq L|x - p^i_k| \leq L(|x - v_i| + |v_i - p^i_k|)
\leq L(|x - v_i| + |v_i - x^i_{k+1}|) \leq L(|x - y| + |x - y|) = 2 \cdot L|x - y|
\]

which is produce by typing

```latex
\begin{align}
|f(x) - f(y)| &= |h(x) - h(p^i_k)| \\
&\leq L|x - p^i_k| \\
&\leq L(|x - v_i| + |v_i - p^i_k|)\\
&\leq L(|x - v_i| + |v_i - x^i_{k+1}|)\\
&\leq L(|x - y| + |x - y|) = 2 \cdot L|x - y| \\
\end{align}
```

New lines are created with the \texttt{\\} command and the point of alignment in each line is at the ampersand \&. Note that there isn't a new line command at the end of the last line nor should there be. If one is included, too much vertical space results. To avoid the lower placement of the number (3), a third line should be created by breaking the second line before the second \texttt{\leq}.

```latex
\begin{align}
|f(x) - f(y)| &= |h(x) - h(p^i_k)| \\
&\leq L|x - p^i_k| \\
&\leq L(|x - v_i| + |v_i - p^i_k|)\\
&\leq L(|x - v_i| + |v_i - x^i_{k+1}|)\\
&\leq L(|x - y| + |x - y|) = 2 \cdot L|x - y| \\
\end{align}
```

To associate just one number with a multi-line calculation, type

```latex
\begin{equation} \label{eq2a} \begin{split}
|f(x) - f(y)| &= |h(x) - h(p^i_k)| \\
&\leq L|x - p^i_k| \\
&\leq L(|x - v_i| + |v_i - p^i_k|)\\
&\leq L(|x - v_i| + |v_i - x^i_{k+1}|)\\
&\leq L(|x - y| + |x - y|) = 2 \cdot L|x - y| \\
\end{split} \end{equation}
```
\begin{align}
  \leq & L |x - p^i_k| \\
  \leq & L (|x - v_i| + |v_i - p^i_k|) \\
  \leq & L (|x - v_i| + |v_i - x_{i, (k+1)}|) \\
  \leq & L (|x - y| + |x - y|) = 2 \cdot L |x - y|
\end{align}

which produces

$$|f(x) - f(y)| = |h(x) - h(p^i_k)| \leq L |x - p^i_k| \leq L (|x - v_i| + |v_i - p^i_k|) \leq L (|x - y| + |x - y|) = 2 \cdot L |x - y|$$

The \texttt{align} environment can be used to produce two (or more) columns of aligned equations. The points of alignment and the points where the columns end must be designated with ampersands. For example

\begin{align}
  8 &\equiv 0 \pmod{8} & 9 &\equiv 1 \pmod{8} & 10 &\equiv 2 \pmod{8} \\
  11 &\equiv 3 \pmod{8} & 12 &\equiv 4 \pmod{8} & 13 &\equiv 5 \pmod{8} \\
  14 &\equiv 6 \pmod{8} & 15 &\equiv 7 \pmod{8} & 16 &\equiv 0 \pmod{8}
\end{align}

is the result of typing

\begin{align}
\begin{align*}
  8 \equiv 0 \pmod{8} & 9 \equiv 1 \pmod{8} & 10 \equiv 2 \pmod{8} \\
  11 \equiv 3 \pmod{8} & 12 \equiv 4 \pmod{8} & 13 \equiv 5 \pmod{8} \\
  14 \equiv 6 \pmod{8} & 15 \equiv 7 \pmod{8} & 16 \equiv 0 \pmod{8}
\end{align*}
\end{align}

You can designate a displayed expression by whatever symbol you wish, say (*), by typing \texttt{\tag{*}} after the expression but before the line is ended by a \texttt{\textbackslash} or the end of the environment. Using \texttt{\textbackslash tag{*}} omits the parentheses.

To omit the equation numbers, replace \texttt{align} by \texttt{align*}. Some displayed mathematics contains too many lines to be completed before the end of the page on which it starts. In this case \LaTeX will move the entire calculation to the next page and spread out the text on the previous page to fill the page by using large sections of white space. The result can be ugly. To correct the situation, you can tell \LaTeX that the calculation can be broken part way through by typing the declaration, \texttt{\allowdisplaybreaks} inside the environment producing the calculation. Or, if you prefer, you can put the declaration in the preamble and it will apply to all calculation environments.

Neither \texttt{gather} nor \texttt{multiline} employs ampersands because no alignment is involved. The \texttt{gather} environment simply centers each of its lines giving a number to each except if the \texttt{*} version is selected. To have one number associated to the entire collections of lines, put \texttt{gather*} inside of an \texttt{equation} environment as was done with \texttt{split} above. The \texttt{multiline} environment moves the first line to the left hand margin (unless the option \texttt{leqno} has been selected), centers the
5.4 The \texttt{amsthm} Package.

This package allows you to create environments for stating assertions, such as definitions, lemmas, theorems, propositions, corollaries, remarks, etc., in a variety of styles and numbering techniques. The environments are created using the commands \texttt{newtheorem} for numbering assertions and \texttt{newtheorem*} to skip numbering. The commands have two mandatory arguments and one optional one, but where the optional one is placed makes a difference. We begin with just the two mandatory arguments. The first argument establishes the name of the environment and the second, its title. For example, \texttt{newtheorem\{lemma\}\{Lemma\}} in the preamble creates the “lemma” environment which is used to enunciate Lemmas. The statement of the lemma is then stated followed by \texttt{end\{lemma\}}. The first time $\LaTeX$ encounters this environment, it numbers it as 1. For example,

\begin{lemma}
If $f(x)=c$ for all $x$, then $f'(x)=0$ for all $x$.
\end{lemma}

Because of the label, later in the document, typing by \texttt{Lemma \ref\{lem1\}} produces, “by Lemma 1”. The next occurrence is numbered \texttt{Lemma 2}.

To suppress the number, replace \texttt{newtheorem} by \texttt{newtheorem*}. For example typing, \texttt{newtheorem*\{WO\}\{Well Ordering\}} in the preamble and in the text typing,

\begin{WO}
Every set can be well-ordered.
\end{WO}

yields

\textbf{Well Ordering.} Every set can be well-ordered.
Some authors prefer to number all assertions, lemmas, theorem, propositions etc. consecutively; for example, Lemma 1, Theorem 2, Corollary 3, etc. To accomplish this goal, in the preamble, type,

\newtheorem{lemma}{Lemma}
\newtheorem{theorem}{lemma}{Theorem}
\newtheorem{corollary}{lemma}{Corollary}

The optional argument, \[lemma\] tells \LaTeX{} not to define a counter, “theorem”, but rather to use the counter, “lemma” that it defined in the previous line. Note that the same optional argument, \[lemma\] occurs in the creation of the “corollary” environment as in the “theorem” environment. If \[theorem\] had been placed there instead, an error message would have resulted stating that there is no counter, “theorem”. Environments for additional assertion to be numbered consecutively can be defined in a similar fashion. But also it’s possible to type, for example, \newtheorem{example}{Example}, which, when evoked, will produce Example 1. no matter how many assertions have been stated before it.

For a very long article, numbering all assertions starting with number 1 can result in some rather large numbers. To avoid this, it’s possible to number assertion according to the section in which they occur; for example, Lemma 2.1 would be the first lemma in the second section. Again an optional argument will result in such a numbering scheme. In the preamble, type \newtheorem{lemma}{Lemma}[section]. Of course the two optional commands can be combined to number all assertion in a given section consecutively. For example, put

\newtheorem{lemma}{Lemma}[section]
\newtheorem{theorem}{lemma}{Theorem}
\newtheorem{corollary}{lemma}{Corollary}

in the preamble and the first use of the “lemma” environment in the third section will result in an assertion named Lemma 3.1 and the first occurrence of the theorem environment in that section will produce an assertion named, Theorem 3.2 and then Corollary 3.3. However,

\newtheorem{lemma}{Lemma}[section]
\newtheorem{theorem}{lemma}{Theorem}
\newtheorem{example}{Example}[section]
\newtheorem{remark}{Remark}

in the preamble will produced Example 3.1 when the “example” environment is used for the first time in Section 3 but Remark 1. the first time “remark” is used no matter what the section.

The \texttt{amsthm} package also provides three “styles” in which theorem-like environments can appear: \texttt{plain}, \texttt{definition}, and \texttt{remark}; \texttt{plain} being the default. For that reason all theorem-like environments that are to produce assertion to be proved, such as theorems, should be defined first and then the style can be
changed, which is accomplished by the command \texttt{\theoremstyle{new\_style\_name}}.
For example, in the preamble type
\begin{verbatim}
\newtheorem{lemma}{Lemma}
\newtheorem{theorem}{Theorem}
\theoremstyle{definition} \index{theoremstyle!definition}
\newtheorem{definition}{Definition}
\theoremstyle{remark}
\newtheorem{remark}{Remark}
\end{verbatim}
to produce

\textbf{Definition 1.} The derivative of \( f \) at \( x \) is defined by \( f'(x) = \lim_{h \to 0} \frac{f(x+h)-f(x)}{h} \).

\textbf{Lemma 2.} Let \( f : [a, b] \) be differentiable on \((a, b)\) and continuous on \([a, b]\). Suppose \( f(a) = f(b) = 0 \). Then there is a \( c \in (a, b) \) such that \( f'(c) = 0 \).

\textbf{Theorem 3.} Let \( f : [a, b] \) be differentiable on \((a, b)\) and continuous on \([a, b]\). Then there is a \( c \in (a, b) \) such that \( f'(c) = \frac{f(b)-f(a)}{b-a} \).

\textbf{Remark 1.} The assumption that \( f \) is differentiable at every \( x \) in \((a, b)\) is essential for the validity of the above theorem.

Also provided in the \texttt{amsthm} package is the “proof” environment, which, as its name suggests, is to contain the proof of an assertion. Typing \texttt{\begin{proof}} automatically begins the proof with, \texttt{Proof.} unless you use of the optional argument to start with something like, \texttt{Proof of Lemma 3.1.} by typing \texttt{\begin{proof}[{\sc Proof of Lemma 3.1}]}\texttt{]. Ending the environment with \texttt{\end{proof} produces the default end-of-proof symbol \( \square \) at the end of the line. A problem arises if the proof ends with a displayed expression because the \( \square \) comes then at the end of the line after the end of the displayed expression; for example,

\[(a^n - b^n) = (a-b)(a^{n-1} + a^{n-2}b + a^{n-3}b^2 + \cdots + a^2b^{n-3} + ab^{n-2} + b^{n-1}).\]

\( \square \)

The command \texttt{\qedhere} is provided to put the \( \square \) at the end of the last line of a displayed expression, but it won’t work with if the display is created with a pair of dollar signs. In that case replace the pair of dollar signs with \texttt{\begin{equation*}} and \texttt{\qedhere\end{equation*}}. This is the use of the environment \texttt{equation*} referred to on page 16. (The page number 16 is produced by typing \texttt{\label{eq*}} at the point where the reference is made and then here typing \texttt{\pageref{eq*}} illustrating another use of the \texttt{\label} command.)

\textbf{5.5 The amsmath Package Continued.}
Before the \texttt{amsmath} package was available, to produce a structure like

\[ f(x) = \begin{cases} 0 & \text{if } x \text{ is a rational number} \\ 1 & \text{if } x \text{ is an irrational number} \end{cases} \]
required using the \textit{array} environment in a somewhat clumsy fashion. But with \texttt{amsmath} it is done by typing

\begin{equation}
\begin{cases}
0 & \text{if } x \text{ is a rational number} \\
1 & \text{if } x \text{ is an irrational number}
\end{cases}
\end{equation}

To number the expression, use \begin{equation} \label{} \end{equation} in place of the pair of double dollar signs. The environment may be used in line as well as in display mode.

The \texttt{amsmath} package provides 6 environment for producing matrices. They are: \texttt{matrix}, \texttt{pmatrix}, \texttt{bmatrix}, \texttt{Bmatrix}, \texttt{vmatrix}, \texttt{Vmatrix}, and \texttt{smallmatrix}. Each is an offshoot of the \texttt{tabular} environment in that columns are separated by \& signs. To produce the matrix, \begin{tabular}{cccc} a \ b \ c \ d \end{tabular}, type

\begin{equation}
\begin{pmatrix}
1 & 2 & 3 & 4 \\
a & b & c & d & \\
-3 & -2 & -1 & 0
\end{pmatrix}
\end{equation}

The next four matrix environments enclose the matrix in different symbols; \texttt{pmatrix} encloses the matrix in parentheses, \texttt{bmatrix}, in brackets \[ \], \texttt{Bmatrix}, in braces \{ \}, \texttt{vmatrix} in a pair of vertical lines \| \|, and \texttt{Vmatrix} in a pair of double vertical lines \| | |. The height of the symbols involved is expanded to match the height of the matrix. For example,

\begin{equation}
\begin{vmatrix}
1 & 2 & 3 & 4 \\
a & b & c & d & \\
-3 & -2 & -1 & 0
\end{vmatrix}
\end{equation}

will create

\begin{tabular}{cccc}
1 & 2 & 3 & 4 \\
a & b & c & d & \\
-3 & -2 & -1 & 0
\end{tabular}

The above examples are used mostly in display mode. Use \texttt{smallmatrix} for in line math mode. However, you must supply the enclosing symbol and adjust the height of the symbols. See the next paragraph. Often in matrices dots are desired. For example

\begin{verbatim}
\begin{pmatrix}
a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & \ldots & a_{1,n-1} & a_{1,n} \\
a_{2,1} & a_{2,2} & a_{2,3} & \ldots & \ldots & \ldots & a_{2,n}
\end{pmatrix}
\end{verbatim}

The matrix has 7 (actual) columns. The second row is produced by typing \begin{verbatim} a_{2,1} & a_{2,2} & a_{2,3} & \ldots & a_{2,n} \end{verbatim}, the third by \begin{verbatim} a_{3,1} & a_{3,2} & a_{3,3} & \ldots & a_{3,n} \end{verbatim}, and the last row, by \begin{verbatim} a_{m,1} & a_{m,2} & a_{m,3} & \ldots & a_{m,n} \end{verbatim}. The argument of the command, \texttt{\bsmallmatrix} is the number of columns that are to be taken up by dots.

The height of bracketing symbols like those used to enclose matrices can by controlled by typing, for example, \texttt{\left(} however, this technique often produces a symbol that is too high. So the \texttt{amsmath} package provides commands
that produce bracketing symbols of varying heights. They are \big, \id{big}\Big, \id{Big} \bigg, and \Bigg. One must try each one and pick the one that looks best. For example, the using the first with the smallmatrix environment yields \( \begin{smallmatrix} a & b \\ c & d \end{smallmatrix} \) comes from typing $\big( \begin{smallmatrix} a & b \\ c & d \end{smallmatrix} \big)$, but for \( \begin{pmatrix} 1 & 0 & 2 \\ 0 & 3 & 4 \\ 2 & 5 & 0 \end{pmatrix} \) the command \Big works better.

Besides the command \vec{x} to produce the simple expression \( \vec{x} \) commands are available to put arrows pointing in each direction or both directions over or under expressions. They are \overleftarrow, \overrightarrow, \underleftarrow, \underrightarrow, and \underleftrightarrow. They work in the same way as \vec does with one mandatory argument. For left or right arrows between symbols with expressions either over the arrow or under the arrow the commands \xleftarrow and \xrightarrow are available. Each has one optional argument and one mandatory argument. The mandatory argument contains the expression to go over the arrow and the optional one, the expression to go under the arrow. For example, $X\xrightarrow[\text{onto}]{}Y$ produces $X \xrightarrow[\text{onto}]{} Y$.

The mandatory argument may be empty, \{ \} to omit an expression above the arrow.

It’s possible to have two lines in a subscript such as $\lim_{n \to \infty} n_{\text{even}}$. Such a construction is accomplished by typing

$\lim_{\substack{n \to \infty \\ n \text{ even}}} \ a_n$. \substack may also be used in superscripts.

The commands \overset and \underset have two mandatory arguments. The first contains a symbol and the second a character over which (or under which) the symbol is to go. For example, $A_{\bullet}$ produces $A$.

The commands \overset and \underset have two mandatory arguments. The first contains a symbol and the second a character over which (or under which) the symbol is to go. For example, $\sum_{\{1,1\} \sim \{1,u\}} \{\{r,1\} \sim \{r,u\}}$ produces $\sum_{\{1,1\} \sim \{1,u\}} \{\{r,1\} \sim \{r,u\}}$.

The commands such as \sin, \tan \log are fine as far as they go, but there are others that you might need. These commands are used in math mode to produce the name of the function in text font and insert a small space after the function name before the next mathematical symbol. For example $\sin x$ produces $\sin x$. The command \DeclareMathOperator allows you to produce function names that aren’t provided. For example typing \DeclareMathOperator{\dist}{dist} in the preamble produces $\dist(A,B)$ when $\dist(A,B)$ is typed.
6 Bibliography.

A bibliography is produced with the environment \texttt{thebibliography} which has one mandatory argument, \texttt{\begin{thebibliography}{longest\_label}}. Each bibliographic entry has a label; usually a number, but not always. If numbers are used and there are fewer than 9 bibliographic entries, the mandatory argument should be \{9\}. With 10 or more but fewer than 100, use \{99\}. On the other hand if bibliographic entries are to be labeled with, say, a sequence of letters, the longest of which is 4 letters, then use \{XXXX\}. The bibliographic entries begin with the command \texttt{\bibitem} which has two arguments; one mandatory and the second optional. The mandatory one is like the mandatory one in the \texttt{\label} command; it assigns a label to the particular bibliographic reference, making it possible to cite the reference in the body of the text by typing \texttt{\cite{label}}. If you wish to list the reference by something other than a number, that label goes in the optional argument. For example the bibliography at the end of these notes is produced by typing

\texttt{\begin{thebibliography}{9}}
\end{thebibliography}

Another option is

\texttt{\begin{thebibliography}{XXX}}
\end{thebibliography}
which would produce

References


7 Importing Graphics.

The latest package used for importing graphics into a \LaTeX document is the package, *graphicx*, which replaced the package, *graphics*. The package *graphicx* recognizes that pdf is quickly replacing post script as the format of choice. It can import graphics in several different form, pdf, jpeg, gif, tiff, but it can’t deal with eps files. (eps stands for encapsulated postscript.) If you use Mathematica, or Maple to create your graphics file, you will need to export it as a pdf file rather than an eps one. (Double-clicking an eps file on a Mac will automatically convert the file into a pdf file.)

The command used to import the graphics file is \verbatim{\includegraphics} which has an optional argument and a mandatory one. The optional one comes first and the mandatory one, which comes second, contains the name of the graphics file; for example, \verbatim{graphics_file.pdf} although the suffix may be omitted. In particular, the command \verbatim{\includegraphics[graphics_file]} will put the graphics immediately after the word preceding the command, even if the command is on the next line. To put it below the current line of text, put two carriage returns after the line. To line the graphics up with the left margin, precede the command with \verbatim{\noindent}. For example typing

Here’s a graphics file sent to me by one of the members of this class.

\verbatim{\noindent\includegraphics[fullwidthantenna.pdf]}
will produce

Here’s a graphics file sent to me by one of the members of this class.

But as you can see, the file is too big for the page. That’s where the optional argument come in. Its formate is \[key=value\]. There are two types of keys; ones that take numbers for values and those that take words for values. The two keys that are used most often are \texttt{width} and \texttt{height}. For example the optional argument \texttt{[width=5in]} will expand or shrink the graphics to a width of 5 inches. The height is also adjusted so that the ratio of width/height remains unchanged. For example typing

Here’s a graphics file sent to me by one of the members of this class.

\includegraphics[width=\textwidth]{fullwidthantenna.pdf}

will produce

Here’s a graphics file sent to me by one of the members of this class.

It’s also possible to change both the width and length with \texttt{[width=5in, height=3in]}. For example typing

Here’s a graphics file sent to me by one of the members of this class.
It’s also possible to use the optional argument to rotate the graphics using the key \texttt{angle} measured in degrees. For example \texttt{[angle=90]} will rotate the graphic around the bottom, left corner by 90 degrees. To change the point of rotation use the key \texttt{loc}. The choices are \texttt{c} for center, \texttt{t} for top, \texttt{r} for right and \texttt{B} for baseline. For example \texttt{[angle=90,loc=c]} will rotate the graphics 90 degrees around its center.

But wait! There’s more. Suppose the graphics file you wish to display is so large that there isn’t room for it on the current page. In that case \LaTeX{} inserts a \texttt{\newpage} command and puts the graphics at the top of the next page leaving the current page with a large blank space at the bottom. The solution is the environment \texttt{figure}, which has one optional command to control the placement of the graphics file. The optional argument lists the possible location in order of preference. The choices are

- \texttt{h} \textit{here} If possible the graphics should appear at the point of the text where it is entered.
- \texttt{t} \textit{top} If there is room for the text already on the current page to appear under the graphics, it should be placed at the top of the current page. If not, then it goes at the top of the next page.
- \texttt{b} \textit{bottom} If there is room for the graphics at the bottom of the current page after the preceding text on the current page, it should go there and any
room under the current text should be filled by text following the graphics. If not, then the graphics is placed at the bottom of the next page.

\(p\) \textit{page} This option is used to put this graphics on a special page reserved for it (and perhaps others).

The default is \textit{tbp}. The command \texttt{\caption[short\_title]{caption\_text}}. The \textit{short\_title} is used in the list of figures if you create one. The \textit{caption\_text} is the text that will be inserted after the automatic numbering of the figure. For example typing

\begin{verbatim}
\begin{figure}[htb]\label{antenna}
\noindent\includegraphics[width=\textwidth]{fullwidthantenna.pdf}
\caption{Full Width Antenna}
\end{figure}
\end{verbatim}

produces

\begin{figure}
\centering
\includegraphics[width=\textwidth]{fullwidthantenna.pdf}
\caption{Full Width Antenna}
\end{figure}

Figure 1: Full Width Antenna

If you’re using the two column format, then the environment name you should use is \texttt{figure*}. It behaves like \texttt{figure} but suspends the two column text setting setting the graphics across the entire page.

The environment \texttt{table} (\texttt{table*}) does for tables created with the \texttt{tabular} or the \texttt{tabbing} environment what \texttt{figure} does for graphics.

8 More on New Commands.

As has already been mentioned, new commands can be created with the \texttt{\DeclareMathOperator} command and with the \texttt{\newcommand}. There’s more to the latter than was introduced. Besides the mandatory argument, \texttt{\newcommand} has optional arguments as well, whose purpose is to allow the creation of commands that have more than one argument; mandatory and optional. The syntax is \texttt{\newcommand[total \# of arguments]} The default values of the optional commands are listed next, each in a separate optional command. Then in the mandatory argument the
actual command is defined using \#n to denote the position of the value of the
n\textsuperscript{th} argument, optional or mandatory. Here is a simple but very useful example.
The Greek letter chi is often used to denote the characteristic function of a set.
Unfortunately when \(\chi_{A}(x)\) is typed (in math mode) the result is \(\chi_{A}(x)\).
Clearly the letter, chi, is set to low. To remedy the situation a new command is defined.

\newcommand{\charf}[1]{\mbox{\raise.48ex\hbox{$\chi$}$_{#1}$}}

It defines one new command, \charf having one mandatory argument. We
know the argument is mandatory, for if it were to be optional, then \[1\] would
have been followed by a second optional argument giving the default value of
the argument. So the syntax of the command is \charf{letter} where letter
is the letter denoting the set to be the subscript of chi. The command \mbox
creates a “box” meaning that the entire contents of the box must be on the same
line. The next tow commands raise the letter, chi, by .48ex, which seems to be
the correct amount to put it on the same line with all of the other characters
in the sentence. Then the newly positioned chi is subscripted with the letter
designated in the mandatory argument. That is, \(\chi_{A}\).

A situation where you may wish to use an optional argument is if you must
enter several finite sequences such as \(x_{1}, x_{2}, \ldots x_{n}\). Most of the time, you’ll use
the letter \(x\) for the sequence, but not always. The new command would look like
\newcommand{\fvec}[2][x]{#1_1,#1_2,\ldots,#1_{#2}}
which states that the new command, \fvec has one optional argument and one mandatory one. If
no value for the optional argument is given, the letter \(x\) will be assumed. The
value for the mandatory argument is the last subscript. Consequently, typing
\fvec{5} in math mode produces \(x_{1}, x_{2}, \ldots x_{5}\), where as typing \fvec[y]{n}
in math mode produces \(y_{1}, y_{2}, \ldots y_{n}\).

The new command defined above would almost always be used in math
mode, but it might be useful to be able to type it in text mode automatically.
The command \ensuremath can be used to accomplish this goal. So refine the
new command to be
\newcommand{\fvec}[2][x]{\ensuremath{#1_{1},#1_{2},\ldots,#1_{#2}}}
Then typing \fvec[z]{m} will produce \(z_{1}, z_{2}, \ldots z_{m}\) either in text mode or in
math mode.

9 Long Documents.

When producing a long document several new aspect arise. First, the size of
the source file of a long document can be clumsy. Second in a long document
it’s desirable to have a table of contents and also, perhaps, an index. In this
section we deal with all of these matters.

To more efficiently handle a long document, it’s convenient to create the
source file in segments. For this purpose \LaTeX provides the notion of a root
file with sub files that are imported into the document using the \include
command. For example suppose a thesis is being written that has an abstract,
an introduction, three chapters and a bibliography. At the outset the root file

28
would look like this.

\documentclass{book}
\usepackage{packages}
\begin{document}
\include{intro}
\include{chapt1}
\include{chapt2}
\include{chapt3}
\include{biblio}
\end{document}

Then five separate files are created for each of the five parts. Because it’s never certain at the beginning what will go into the bibliography, the \texttt{biblio} file should be updated as the others are in production. These file contain only text; no \texttt{\begin{document}} etc. commands. All of these files must be in the same directory. While working on the \texttt{intro} file, and building the \texttt{biblio} file, the root file is slightly altered as follows.

\documentclass{book}
\usepackage{packages}
\includeonly{intro,biblio}
\begin{document}
\include{intro}
\include{chapt1}
\include{chapt2}
\include{chapt3}
\include{biblio}
\end{document}

Once the introduction is finish (at least in its first form) and chapter 1 is started and the command \texttt{\includeonly{intro,biblio}} is changed to \texttt{\includeonly{chapt1,biblio}}. (Some prefer to comment out the first \texttt{\includeonly} command and add the second.) Each time \LaTeX is run, the auxiliary file is updated; not replaces. Thus all labels and citations created during the construction of the Introduction are preserved and consequently, if any are referenced in the new file, they will be recognized when the root file is compiled. Continue in this fashion to construct all of the files. Then remove all of the \texttt{\includeonly} commands and compile the file. Each time \LaTeX sees an \texttt{\include} command, it enters and \texttt{\clearpage} command thereby starting the next file on a new page. For that reason the files must represent separate parts of the document.

9.1 Table of Contents.

Creating a table of contents for any \LaTeX document is quite easy. Simply type \texttt{\tableofcontents} at the place in the document where you wish the table of
contents to appear, usually immediately after the abstract, or the title page if there is one. Adding \newpage will start the rest of the document on a new page. It takes two runs to create the table of contents.

9.1.1 Depth.

By default sections, subsection and subsubsections are listed in the table of contents, that is, to a depth of three. The user can change the depth with the command \setcounter{tocdepth}{desired_depth} or \addtocounter{tocdepth}{desired_change}. For example, if the depth is set to 4, then paragraph names will be included.

9.1.2 Adding Items.

The command \tableofcontents produces a file names main_file_name.toc to which can entries can be added. For example sections entered with the command \section* are not automatically put into the table of contents. To add the section given by \section*{Extra Section}, type \addtocontentsline{toc}{section}{Extra Section}.

9.2 Index.

Creating an index is far more complex. Unlike the table of contents, the author must specify each item to be included in the index. This is accomplished with a command beginning with \index, which has a very complicated structure due to the fact that index items are nested. To create the index the file must be compiled and then a program must be run on a file created by \LaTeX. Finally the location of the index must be selected by typing the command \printindex at the point in the source file where the index is to appear, usually just before the \end{document} command.

9.2.1 The Command \index.

The process of producing the index is begun by discussing how entries in the index are selected. Those index items that are to be at the first level (i.e., at the left margin of index) are specified by simply typing \index{index_entry} at the point where the index_entry exists in the text to which the author wants the reader to be directed. If the entry is to be in, say, small caps, then the appropriate form for the command is \index{index_entry@\textsc{index_entry}}. If the page number is to be in, say, boldface, then the entry should be \index{index_entry@\textbf{index_entry}}. If the index_item appears on several consecutive pages, type \index{index_entry| on the first of the consecutive pages and \index{index_entry|)} on the last. This item will be given the appropriate run of pages numbers in the index.

For those items in the index that are to be listed under a primary index entry, appropriate syntax is \index{primary_entry! secondary entry}. All of the above formatting commands
discussed for the primary entry apply equally well to the secondary entry. There are several additional formatting procedures, but one worth mentioning concerns Greek letters. If you wish to have a \( \delta \) appear as an index item, type \( \texttt{\index{delta@$\delta$}} \).

9.2.2 Creating the Index File.

To actually create the index, first include the package \texttt{makeidx} with a \texttt{\usepackage}, put the command \texttt{\makeindex} in the preamble, and compile the file twice. In that process, \LaTeX{} will create files named \texttt{file\_name.idx} and \texttt{file\_name.ist}. Then run the program \texttt{MakeIndex}. (How that is done depends on the editor you’re using and the operating system of your computer.) That program creates a file (which is actually an environment) that \LaTeX{} inputs when it runs next. That is when the index appears at the point where the command \texttt{\printindex} appears.

10 Boxes.

There are three types of boxes: \texttt{LR}, \texttt{Rule} and \texttt{Par}. We’ll begin with the first.

10.1 LR Boxes.

These boxes come in two verities: plain and framed. The difference is that the framed verity has a visible frame around it. The contents of a \texttt{LR} box are typeset from left to right on a single line. Consequently, line breaks aren’t allowed in such a box. Such a box can be constructed easily with the command \texttt{\mbox}, which has one mandatory argument or with the more versatile version, \texttt{\makebox}, which has two optional arguments and one mandatory argument. Typing \texttt{\mbox{a line of text}} ensures that a line of text all appears on the same line. The second version, \texttt{\fbox{a line of text}} puts a frame around the text, \texttt{[a line of text]}. The more versatile version is of the form, \texttt{\makebox[width]{pos}{text}}. The optional command \texttt{[width]} allows the width of the box to be determined independent of the text it contains. The default is \texttt{\width}, the width of the text enclosed. So one possible option for the width is \texttt{1.5\width}. Another very useful choice is \texttt{0cm}. The optional command, \texttt{[pos]} determines the position of the text in the box. The default is \texttt{c} for centered. The other two choices are \texttt{l} for flush left and \texttt{r} for flush right. With a box of \texttt{0 width} the first choice puts the text to the left of the box and to the right for the second. The same two optional arguments are available for the command \texttt{\framebox}. With any frame box it’s possible to increase the width of the frame with the command \texttt{\fboxrule} and the space at either end with the command \texttt{\fboxsep}. For example typing

\[\texttt{\setlength{\fboxrule}{1mm}\setlength{\fboxsep}{2mm}\fbox{Some text in a box}}\]

yields \textbf{Some text in a box}
In conjunction with these boxes come the very useful command \texttt{\textbackslash raisebox} which is used to create and raise or lower a LR box. The syntax for the command is \texttt{\textbackslash raisebox}\{lift\} [\texttt{height}][\texttt{depth}] \{contents\}.

### 10.2 Rule Boxes

The command \texttt{\textbackslash rule} has one optional arguments, \texttt{lift}, and two mandatory arguments, \texttt{width} and \texttt{height}. It produces a black line whose height is determined by \texttt{height} and whose width is \texttt{width}. For example \texttt{\textbackslash rule\{3in\}\{2mm\}} produces \textbf{ }. The optional argument, \texttt{lift} may be either positive or negative and will move the rule vertically the designated amount.

### 10.3 Paragraph Boxes.

Paragraph boxes are to pages what LR boxes are to lines. They are created with the \texttt{\textbackslash parbox} command, which has first three optional arguments followed by two mandatory arguments. The first optional argument, \texttt{pos}, has three choices: \texttt{t}, \texttt{c}, \texttt{b} with \texttt{c} being the default. The baseline of the current line is aligned with the base line of the top, center, or bottom line of the text in the box respectively. The default is \texttt{c}. The second optional argument, \texttt{height}, can be a specific length, such as 2in or a multiple of a fixed length, such as \texttt{\textbackslash height}, the height of the text in the box, which is the default setting. Only if the height is other that its default does the third optional argument, \texttt{inner-pos}, get selected. It has three choices, \texttt{t}, \texttt{c}, \texttt{b} with \texttt{c} being the default. The first mandatory argument is \texttt{width}, which, like \texttt{height}, can be assigned a specific number or a multiple of a standard length such as \texttt{\textbackslash textwidth} or \texttt{\textbackslash linewidth}. By the way, it’s possible to put a frame around a paragraph box by putting the \texttt{\parbox} command inside of the argument of a \texttt{\fbox} command. To put, say, footnotes, inside the paragraph box, use the environment, \texttt{minipage}, instead. It has the same arguments as does the \texttt{\parbox} command.

### 11 Commutative Diagrams.

Simple commutative diagrams can be constructed using the package, \texttt{amscd}, which must be entered after the package, \texttt{amsmath}. Only diagrams with vertical and horizontal arrows can be made. These arrow may point in either direction (left or right for horizontal arrows and up or down for vertical ones). In addition symbols may be entered or either or both sides of these arrows. The commands that produce these four types of arrows are: \texttt{\textbackslash arrow} for arrows pointing to the right, \texttt{\textbackslash backarrow} for arrows pointing left, \texttt{\textbackslash downarrow} for arrows pointing down and \texttt{\textbackslash uparrow} for arrows pointing up. The symbols to accompany the arrows are inserted between the characters. For example \texttt{\textbackslash arrow\{f\}1-1\textbackslash arrow} produces a right arrow with an \texttt{f} over it and 1-1 beneath it. All of these commands must be given inside of the environment, \texttt{CD}. For example
\begin{CD}
X @>f>> Y \\
@VhV\text{onto}V @AgA\text{iso}A \\
X/G = Y/H
\end{CD}

produces

\[
\begin{array}{c}
X \xrightarrow{f} Y \\
\downarrow h \downarrow \text{onto} \quad \uparrow \downarrow g \text{iso} \\
X/G \quad Y/H
\end{array}
\]
Appendix.

Dimensions

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Unit</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>mm</td>
<td>millimeter</td>
<td>bp = big point (1 in = 72 bp)</td>
</tr>
<tr>
<td>cm</td>
<td>centimeter</td>
<td>dd = (1157 dd = 1238 pt)</td>
</tr>
<tr>
<td>in</td>
<td>inch (1 in = 2.54 cm)</td>
<td>cc = cicero (1 cc = 12 dd)</td>
</tr>
<tr>
<td>pt</td>
<td>point (1 in = 72.27 pt)</td>
<td>sp = (1 pt = 65536 sp)</td>
</tr>
<tr>
<td>pc</td>
<td>pica (1 pc = 12 pt)</td>
<td></td>
</tr>
</tbody>
</table>

em = The current width of a capital M
ex = The current height of the letter x

Accents

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Accent</th>
</tr>
</thead>
<tbody>
<tr>
<td>ò</td>
<td>\o</td>
</tr>
<tr>
<td>ò</td>
<td>\o</td>
</tr>
<tr>
<td>ò</td>
<td>\o</td>
</tr>
<tr>
<td>ò</td>
<td>\o</td>
</tr>
<tr>
<td>ò</td>
<td>\o</td>
</tr>
<tr>
<td>ç</td>
<td>\c</td>
</tr>
<tr>
<td>è</td>
<td>\e</td>
</tr>
<tr>
<td>ì</td>
<td>\i</td>
</tr>
</tbody>
</table>

Special letters from other languages (2.5.6) - p. 27.

æ = \ae
œ = \oe
æ = \ae
Æ = \AE
å = \aa
Å = \AA
' = \acute

Special symbols

† \dag
§ \S
© \copyright
‡ \ddag
¶ \P
£ \pounds

Command symbols

\{ \}
\{ \}
\{ \}
\{ \}
\{ \}

Greek letters

Lowercase letters

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Greek Letter</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>\alpha</td>
</tr>
<tr>
<td>β</td>
<td>\beta</td>
</tr>
<tr>
<td>γ</td>
<td>\gamma</td>
</tr>
<tr>
<td>δ</td>
<td>\delta</td>
</tr>
<tr>
<td>ε</td>
<td>\epsilon</td>
</tr>
<tr>
<td>ζ</td>
<td>\zeta</td>
</tr>
<tr>
<td>η</td>
<td>\eta</td>
</tr>
<tr>
<td>θ</td>
<td>\theta</td>
</tr>
<tr>
<td>ϑ</td>
<td>\vartheta</td>
</tr>
<tr>
<td>ϕ</td>
<td>\phi</td>
</tr>
<tr>
<td>χ</td>
<td>\chi</td>
</tr>
<tr>
<td>ψ</td>
<td>\psi</td>
</tr>
<tr>
<td>ω</td>
<td>\omega</td>
</tr>
<tr>
<td>τ</td>
<td>\tau</td>
</tr>
<tr>
<td>ι</td>
<td>\iota</td>
</tr>
<tr>
<td>ω</td>
<td>\omega</td>
</tr>
<tr>
<td>θ</td>
<td>\theta</td>
</tr>
<tr>
<td>ρ</td>
<td>\rho</td>
</tr>
<tr>
<td>ϑ</td>
<td>\vartheta</td>
</tr>
<tr>
<td>σ</td>
<td>\sigma</td>
</tr>
<tr>
<td>ς</td>
<td>\varsigma</td>
</tr>
<tr>
<td>ν</td>
<td>\nu</td>
</tr>
<tr>
<td>ξ</td>
<td>\xi</td>
</tr>
</tbody>
</table>

Uppercase letters

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Greek Letter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Α</td>
<td>\Lambda</td>
</tr>
<tr>
<td>Β</td>
<td>\Beta</td>
</tr>
<tr>
<td>Γ</td>
<td>\Gamma</td>
</tr>
<tr>
<td>Δ</td>
<td>\Delta</td>
</tr>
<tr>
<td>Ε</td>
<td>\Epsilon</td>
</tr>
<tr>
<td>Ζ</td>
<td>\Zeta</td>
</tr>
<tr>
<td>Η</td>
<td>\Eta</td>
</tr>
<tr>
<td>Θ</td>
<td>\Theta</td>
</tr>
<tr>
<td>Λ</td>
<td>\Lambda</td>
</tr>
<tr>
<td>Μ</td>
<td>\Mu</td>
</tr>
<tr>
<td>Ν</td>
<td>\Nu</td>
</tr>
<tr>
<td>Ξ</td>
<td>\Xi</td>
</tr>
<tr>
<td>Π</td>
<td>\Pi</td>
</tr>
<tr>
<td>Ρ</td>
<td>\Rho</td>
</tr>
<tr>
<td>Σ</td>
<td>\Sigma</td>
</tr>
<tr>
<td>Τ</td>
<td>\Tau</td>
</tr>
<tr>
<td>Υ</td>
<td>\Upsilon</td>
</tr>
<tr>
<td>Φ</td>
<td>\Phi</td>
</tr>
<tr>
<td>Ψ</td>
<td>\Psi</td>
</tr>
<tr>
<td>Ω</td>
<td>\Omega</td>
</tr>
</tbody>
</table>
Summary tables and figures

Binary operation symbols

\pm \cap \circ \bigcirc \mp \cup \bullet \Box \times \uplus \diamond \div \sqcap \lhd \ast \vee \sqcup \rhd \star \wedge \triangledown \odot \otimes

Relational symbols

\le \leq \ge \geq \neq \sim \ll \gg \leq \geq \approx \asymp \subseteq \supseteq \approx \asymp \subseteq \supseteq \leq \geq \approx \asymp \subseteq \supseteq \leq \geq \approx \asymp \subseteq \supseteq \leq \geq \approx \asymp

Negated relational symbols

\not< \not\le \not\ge \not\prec \not\equiv \not\succ \not\preceq \not\succceq \not\subset \not\supset \not\subseteq \not\supseteq \not\subsetneq \not\supsetneq

Brackets

| ( | ) | \lceil | \rceil | \{ | \} | \langle | \rangle | \lfloor | \rfloor | \lceil | \rceil | \langle | \rangle | \downarrow | \uparrow | \downarrow | \uparrow | \downarrow | \uparrow |

*The underlined commands can only be used with packages \texttt{latexsym} or \texttt{amsfonts}.*
<table>
<thead>
<tr>
<th>Binary operation symbols*</th>
<th>\pm</th>
<th>\cap</th>
<th>\circ</th>
<th>\bigcirc</th>
<th>\mp</th>
<th>\cup</th>
<th>\bullet</th>
<th>\Box</th>
</tr>
</thead>
<tbody>
<tr>
<td>\times</td>
<td>\uplus</td>
<td>\diamond</td>
<td>\Diamond</td>
<td>\div</td>
<td>\sqcap</td>
<td>\triangledown</td>
<td>\ast</td>
<td>\vee</td>
</tr>
<tr>
<td>\cdots</td>
<td>\sqcup</td>
<td>\triangleright</td>
<td>\star</td>
<td>\wedge</td>
<td>\cupdot</td>
<td>\ominus</td>
<td>\odot</td>
<td>\wr</td>
</tr>
</tbody>
</table>

*The underlined commands can only be used with packages latexsym or amsfonts.

<table>
<thead>
<tr>
<th>Relational symbols*</th>
<th>\leq</th>
<th>\geq</th>
<th>\neq</th>
<th>\sim</th>
<th>\ll</th>
</tr>
</thead>
<tbody>
<tr>
<td>\subset</td>
<td>\supset</td>
<td>\approx</td>
<td>\cong</td>
<td>\equiv</td>
<td>\subset</td>
</tr>
<tr>
<td>\subseteq</td>
<td>\supseteq</td>
<td>\models</td>
<td>\models</td>
<td>\not&lt;</td>
<td></td>
</tr>
<tr>
<td>\nsubseteq</td>
<td>\nsubseteq</td>
<td>\models</td>
<td>\models</td>
<td>\not\leq</td>
<td></td>
</tr>
<tr>
<td>\not\prec</td>
<td>\not\preceq</td>
<td>\models</td>
<td>\models</td>
<td>\not\succ</td>
<td></td>
</tr>
<tr>
<td>\not\subseteq</td>
<td>\not\supseteq</td>
<td>\models</td>
<td>\models</td>
<td>\not\succ</td>
<td></td>
</tr>
</tbody>
</table>

The underlined commands can only be used with packages latexsym or amsfonts.

<table>
<thead>
<tr>
<th>Negated relational symbols</th>
<th>\not&lt;</th>
<th>\not\leq</th>
<th>\not\prec</th>
<th>\not\preceq</th>
<th>\not\subseteq</th>
<th>\not\sqsubset</th>
<th>\not\subset</th>
</tr>
</thead>
<tbody>
<tr>
<td>\not=</td>
<td>\not\geq</td>
<td>\not\succ</td>
<td>\not\succ</td>
<td>\not\sqsupseteq</td>
<td>\not\supseteq</td>
<td>\not\supset</td>
<td></td>
</tr>
<tr>
<td>\not\models</td>
<td>\not\models</td>
<td>\models</td>
<td>\models</td>
<td>\models</td>
<td>\models</td>
<td>\models</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Brackets</th>
<th>{</th>
<th>}</th>
<th>\lfloor</th>
<th>\rfloor</th>
<th>\lceil</th>
<th>\rceil</th>
</tr>
</thead>
<tbody>
<tr>
<td>{</td>
<td>}</td>
<td>\langle</td>
<td>\rangle</td>
<td>\uparrow</td>
<td>\uparrow</td>
<td></td>
</tr>
<tr>
<td>\uparrow</td>
<td>\downarrow</td>
<td>\downarrow</td>
<td>\downarrow</td>
<td>\updownarrow</td>
<td>\updownarrow</td>
<td></td>
</tr>
</tbody>
</table>

36
Summary tables and figures

Binary operation symbols*

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>±</td>
<td>\pm</td>
</tr>
<tr>
<td>≠</td>
<td>\neq</td>
</tr>
<tr>
<td>⊕</td>
<td>\oplus</td>
</tr>
<tr>
<td>×</td>
<td>\times</td>
</tr>
<tr>
<td>÷</td>
<td>\div</td>
</tr>
<tr>
<td>⋅</td>
<td>\cdot</td>
</tr>
<tr>
<td>⋊</td>
<td>\amalg</td>
</tr>
<tr>
<td>⋓</td>
<td>\sqcup</td>
</tr>
<tr>
<td>⋐</td>
<td>\sqcap</td>
</tr>
<tr>
<td>⊆</td>
<td>\subseteq</td>
</tr>
<tr>
<td>⊇</td>
<td>\supseteq</td>
</tr>
<tr>
<td>⊂</td>
<td>\subset</td>
</tr>
<tr>
<td>⊃</td>
<td>\supset</td>
</tr>
<tr>
<td>⊊</td>
<td>\not\subseteq</td>
</tr>
<tr>
<td>⊋</td>
<td>\not\supset</td>
</tr>
</tbody>
</table>

Relational symbols*

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>≤</td>
<td>\leq</td>
</tr>
<tr>
<td>≥</td>
<td>\geq</td>
</tr>
<tr>
<td>≦</td>
<td>\equiv</td>
</tr>
<tr>
<td>⊆</td>
<td>\subseteq</td>
</tr>
<tr>
<td>⊇</td>
<td>\supseteq</td>
</tr>
<tr>
<td>⊊</td>
<td>\not\subseteq</td>
</tr>
<tr>
<td>⊋</td>
<td>\not\supset</td>
</tr>
<tr>
<td>≜</td>
<td>\models</td>
</tr>
</tbody>
</table>

Negated relational symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>¬&lt;</td>
<td>\not&lt;</td>
</tr>
<tr>
<td>¬\leq</td>
<td>\not\leq</td>
</tr>
<tr>
<td>¬\prec</td>
<td>\not\prec</td>
</tr>
<tr>
<td>¬\preceq</td>
<td>\not\preceq</td>
</tr>
<tr>
<td>¬\subset</td>
<td>\not\subset</td>
</tr>
<tr>
<td>¬\supset</td>
<td>\not\supset</td>
</tr>
<tr>
<td>¬\subseteq</td>
<td>\not\subseteq</td>
</tr>
<tr>
<td>¬\supseteq</td>
<td>\not\supseteq</td>
</tr>
<tr>
<td>¬\in</td>
<td>\not\in</td>
</tr>
</tbody>
</table>

Brackets

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>(</td>
<td>\left(</td>
</tr>
<tr>
<td>)</td>
<td>\right)</td>
</tr>
<tr>
<td>[</td>
<td>\left[</td>
</tr>
<tr>
<td>]</td>
<td>\right]</td>
</tr>
<tr>
<td>{}</td>
<td>{}</td>
</tr>
<tr>
<td>}</td>
<td>}}</td>
</tr>
<tr>
<td>\backslash</td>
<td>\backslash</td>
</tr>
</tbody>
</table>

*The underlined commands can only be used with packages latexsym or amsfonts.
AMS Greek and Hebrew letters

\[ \digamma \quad \kappa \quad \varkappa \]
\[ \beth \quad \tau \quad \daleth \quad \gimel \]

AMS delimiters

\[ \llcorner \quad \lrcorner \quad \llcorner \quad \lrcorner \]

Table G.24 AMS relational symbols

\[ \leqq \quad \leqslant \quad \leqssim \]
\[ \leqslantless \quad \lesssim \quad \approx \]
\[ \lessapprox \quad \approxeq \quad \lesssim \]
\[ \lesseqqurr \quad \lesseqqrr \]
\[ \lesseqqrrtr \quad \doteqdot \quad \fallingdotseq \]
\[ \backsimeq \quad \backsimeq \quad \preccurlyeq \]
\[ \preccurlyeq \quad \preccurlyeq \quad \prec \]
\[ \precapprox \quad \vartriangleleft \quad \triangledown \]
\[ \vdash \quad \triangleleft \quad \smallsmile \]
\[ \smallfrown \quad \bump \quad \geq \]
\[ \geqslant \quad \geqslantgtr \quad \gtrsim \]
\[ \gtrapprox \quad \gtrapprox \quad \gg \]
\[ \gtreqqless \quad \gtreqqless \quad \approx \]
\[ \approx \quad \approx \quad \cong \]
\[ \cong \quad \cong \quad \succ \]
\[ \succ \quad \succ \quad \succ \]
\[ \succapprox \quad \vartriangleleft \quad \triangleleft \]
\[ \vdash \quad \shortparallel \quad \pitchfork \]
\[ \therefore \quad \because \]

\[ \blacktriangleright \quad \because \]
Summary tables and figures

Table G.25 $\mathcal{AMS}$ negated relational symbols

<table>
<thead>
<tr>
<th>\textbackslash{less}</th>
<th>\textbackslash{leq}</th>
</tr>
</thead>
<tbody>
<tr>
<td>\textbackslash{leqslant}</td>
<td>\textbackslash{leqq}</td>
</tr>
<tr>
<td>\textbackslash{leq}</td>
<td>\textbackslash{leq}</td>
</tr>
<tr>
<td>\textbackslash{vertneqq}</td>
<td>\textbackslash{nsim}</td>
</tr>
<tr>
<td>\textbackslash{inapprox}</td>
<td>\textbackslash{nprec}</td>
</tr>
<tr>
<td>\textbackslash{nprec}</td>
<td>\textbackslash{nprecnsim}</td>
</tr>
<tr>
<td>\textbackslash{precapprox}</td>
<td>\textbackslash{nsim}</td>
</tr>
<tr>
<td>\textbackslash{nshortmid}</td>
<td>\textbackslash{nmid}</td>
</tr>
<tr>
<td>\textbackslash{nvDash}</td>
<td>\textbackslash{nvDash}</td>
</tr>
<tr>
<td>\textbackslash{ntriangledownleft}</td>
<td>\textbackslash{ntriangledownlefteq}</td>
</tr>
<tr>
<td>\textbackslash{nsupseteq}</td>
<td>\textbackslash{nsupseteq}</td>
</tr>
<tr>
<td>\textbackslash{varsupseteq}</td>
<td>\textbackslash{varsupseteq}</td>
</tr>
<tr>
<td>\textbackslash{varsupseteq}</td>
<td>\textbackslash{ngtr}</td>
</tr>
<tr>
<td>\textbackslash{ngeq}</td>
<td>\textbackslash{ngtr}</td>
</tr>
<tr>
<td>\textbackslash{ngtr}</td>
<td>\textbackslash{ngtr}</td>
</tr>
<tr>
<td>\textbackslash{gnsim}</td>
<td>\textbackslash{gnapprox}</td>
</tr>
<tr>
<td>\textbackslash{nsucc}</td>
<td>\textbackslash{nsucc}</td>
</tr>
<tr>
<td>\textbackslash{nsucc}</td>
<td>\textbackslash{nsuccnsim}</td>
</tr>
<tr>
<td>\textbackslash{succapprox}</td>
<td>\textbackslash{ncong}</td>
</tr>
<tr>
<td>\textbackslash{nshortparallel}</td>
<td>\textbackslash{nparallel}</td>
</tr>
<tr>
<td>\textbackslash{nvDash}</td>
<td>\textbackslash{nvDash}</td>
</tr>
<tr>
<td>\textbackslash{ntriangledownright}</td>
<td>\textbackslash{ntriangledownrighteq}</td>
</tr>
<tr>
<td>\textbackslash{nsupseteq}</td>
<td>\textbackslash{nsupseteq}</td>
</tr>
<tr>
<td>\textbackslash{supseteq}</td>
<td>\textbackslash{varsupseteq}</td>
</tr>
<tr>
<td>\textbackslash{varsupseteq}</td>
<td>\textbackslash{varsupseteq}</td>
</tr>
</tbody>
</table>

Miscellaneous $\mathcal{AMS}$ symbols

| $\hbar$ | $\textbackslash{hslash}$ |
| $\vartriangle$ | $\textbackslash{triangledown}$ |
| $\square$ | $\lozenge$ |
| $\circledS$ | $\angle$ |
| $\measuredangle$ | $\exists$ |
| $\mathcal{M}$ | $\textbackslash{Finv}$ |
| $\mathcal{O}$ | $\text{Bbbk}$ |
| $\backprime$ | $\text{varnothing}$ |
| $\blacktriangle$ | $\blacktriangledown$ |
| $\blacksquare$ | $\blacklozenge$ |
| $\blacklozenge$ | $\text{eth}$ |
| $\bigstar$ | $\text{eth}$ |
| $\complement$ | $\text{eth}$ |
| $\diagup$ | $\text{eth}$ |
References


