CHAPTER 9

An Introduction to TEX

TEX is a computer program for creating beautiful manuscripts, particularly those containing mathematics. This report was created using TEX, while many new books are being created using TEX. All publications of the American Mathematical Society are now being produced using TEX. While there are many programs for document preparation, TEX has a number of advantages, including a) it is used all over the world on all types of computers, b) to use TEX the user creates a file containing no 'special characters' (usually referred to as an 'ASCII file'), which allows easy communication of documents via electronic means (such as e-mail), and c) most observers feel that documents created using TEX are superior in quality than those created by other software.

In this chapter, we introduce the reader to how TEX can be used to produce mathematical manuscripts. The basic reference for the use of TEX is the *TEXbook* by Donald Knuth (the author of TEX) which is widely available in book stores.

1. Introduction

TEX is not a WYSIWYG (what you see is what you get) program, that is, the user does not see what the paper will look like until it is printed or previewed on the computer screen. One must first produce a file that the TEX program can process. This file is referred to as a TEX file and it should have `.tex` as its extension. On an X-Station, one would have an emacs window open to type in the TEX file. An example of such a file is

```
The pdf $f$ and cdf $F$ of an exponential distribution
having parameter $\lambda$ are given by
$$
f(x)=\lambda e^{-\lambda x},\qquad F(x)=1-e^{-\lambda x},\qquad x>0.
$$

\bye
```

which produces the output

The pdf f and cdf F of an exponential distribution having parameter λ are given by

$$f(x)=\lambda e^{-\lambda x},\qquad F(x)=1-e^{-\lambda x},\qquad x>0.$$

After producing a TEX file, the user must do the following steps to produce a printed document:

1. Use the TEX program to process the TEX file that was created in step (1). This is done by the command `tex filename` where `filename` need not include the `.tex` extension. If there are no mistakes in the TEX file, this step will produce a file called a 'DVI' file since it is 'device independent', that is, it will be the same no matter what computer one is using. The DVI file will have the same name as the TEX file except it will have extension `.dvi`. Using the TEX program will also create a 'log file' having the same

name as the TEX file except extension `.log`. If there are mistakes in the TEX file, the TEX program will describe what they are. The user must then correct the mistakes in the TEX file.

2. Once there are no more mistakes in the TEX file, the user executes a 'printer driver' program which will convert the DVI file to another file which is in a form that is understandable to the particular printer that is connected to the user's computer. In many cases, this printer driver program will also actually send the final file to the printer. Also, many computers have a program called a 'screen previewer' program that can be used to see a low resolution version of the final document on their screen. On many X-Stations, the printer driver programs are called `dvihp` for Laserjet printers and `dvips` for postscript printers, while the screen previewer is called `xdvi`.

3. If the printer driver does not send the final output to the printer, the user executes a printing program to send the file produced in step (2) to the printer.

For more information on this process, see your local system administrator.

Commands and Macros

To typeset ordinary text, the user merely types the text and TEX determines line and page breaks automatically. To tell TEX to do something other than just process text, one types what is called a 'command' (also called a macro), that is, a backslash (like this: \), followed by the name of the command, followed by a space. This space is needed so that TEX knows that the name of the command has ended. For example, typing `\TeX` produces TEX. The names of macros contain only latin letters (no numerical digits for example). Thus if the name of a macro is followed by a character that is not a letter, the usually required space following the name is not required. Note that throughout this report, we will indicate something that is to be typed by the user by having it appear in typewriter font (`like this`).

To tell TEX to end the current paragraph, one issues the command `\par` or leaves a blank line in the file.

TEX comes with about 300 commands built in and one can create their own set of commands. Almost all computers that have TEX installed also have several files containing sets of macros that users have created. These files are usually referred to as 'style files', and usually also include commands specifying what margins and fonts are to be used. The best thing to do is to ask around and find out what style files there are and what they contain.

Sizes and Fonts

1. By default, TEX leaves a one inch margin all around the page. This is controlled by the parameters `\hsize`, `\vsize`, `\hoffset`, and `\voffset`, which are 6.5, 9.0, 0.0, and 0.0 by default. To have a 5 by 7 inch page, still centered on the paper, one would use `\hsize=5in`, `\vsize=7in`, `\hoffset=0.75in`, `\voffset=1in`.

2. By default, TEX numbers the pages of a document starting with the number 1 and places the numbers centered at the bottom of each page. One can have the numbers start with something other than 1 by issuing the command (to start with 10, for example) `\pageno=10`. To get Roman numerals for page numbers, set `\pageno` equal to a negative value. One can also place the page numbers somewhere else and get a running headline.

3. The nonmathematical characters that are produced are in what is called ten point Computer Modern Roman font unless the user does something to change this default setting. This font is called ten point because the height of the tallest character (which is a parenthesis) is ten points high (1 inch is 72.27 points). Each line of text has what is called a baseline (the bottom of the letters that don't have 'descenders'), and by default these baselines are twelve points apart. This can be changed to 14 points for example by the command `\baselineskip=14pt`. When typing math (see below), latin characters are in what is called math italics ($like\ this$). Note that there is no concept of 'doublespaced' in TEX. In fact, there is nothing like 'lines' on a typewriter. How much space is left between lines of what is printed is determined by the `baselineskip`.

4. Several other types of fonts are available, such as italics (*like this*), or bold face (**like this**), or any combination of Roman, *italics*, **bold face**, `typewriter`, *slanted*, or SMALL CAPS, by typing for example

```
or any combination of Roman, {\it italics}, {\bf bold face},
{\tt typewriter}, {\sl slanted}, or {\smc small caps}
```

   This illustrates the very important concept of 'grouping.' Note that we have surrounded each of the words in the various fonts by left and right curly brackets. This makes it so that the font only lasts inside of the brackets. There are a number of situations when you will want something to apply only temporarily and using the brackets will allow this.

5. At any one time, TEX uses fonts of three sizes. One size is for regular characters, while the other two sizes are for sub or superscripts and sub or superscripts applied to sub or superscripts. The default sizes are 10, 7, and 5 points. For example, in X_{n_i}, the default size for the X is ten points, the n is seven points, and the i is five points. Usually a computer installation will have established a style file that will give other sizes since many people find 10, 7, and 5 points to be too small.

6. One can magnify all sizes in a document by five different factors called 'magsteps' which are half, 1, 2, 3, and 4, for example if you include the line

```
\magnification=\magstep1
```

   at the beginning of your document, then all sizes will be multiplied by 1.2. The five factors are $\sqrt{1.2}$, 1.2, 1.2^2, 1.2^3, and 1.2^4. You can only apply a magnification once in a document and it applies throughout the document. One is tempted to use magnification to get fonts of sizes other the default 10, 7, and 5 points (for producing overheads for example), but it is important to understand that fonts do not come in all sizes. To define a 12 point bold face font called `bfone`, one includes the line

```
\font\bfone=cmbx10 scaled\magstep1
```

   in the TEX file prior to using the font and then invoking the font by saying its name in a way similar to using `\bf` in item 4 above. The important part of this line above is the font name `cmbx10`. The other names are `cmti10` for *italics*, `cmtt10` for `typewriter`, `cmsl10` for *slanted*, `cmcsc` for SMALL CAPS. There are many other families of fonts as well.

7. The first line of each paragraph is indented by an amount controlled by the parameter `\parindent`, the default value of which is 20 pts. You can avoid the indenting of the first line by typing `\noindent`.

Vertical and Horizontal Skips

To skip space vertically one types (to skip two millimeters for example; there are also `in`, `cm`, and `pt` for inches, centimeters, and points) `\vskip 2mm`. Note that doing this will end the current paragraph. To skip space horizontally, one uses `hskip` instead of `vskip`. For a variety of important reasons, these skips are not obeyed at the beginning of a page or the beginning of a line, respectively. To get some blank space at the top of a page, one can type `\vtop{ }\vskip 4mm`.

Using Comments in the TEX File

Anything following a percent sign on a line in the TEX file is treated as a comment and is ignored by the TEX program (see the last page of this chapter for an example of using comments). As with any other programming language, it is important to put comments into the file. This is particularly important in TEX as TEX files are notoriously difficult to read and you will use previous files as templates for future documents and you will need to be able to find useful examples. Further, while you are 'debugging' your TEX file (getting all the syntax errors out of it) you will need to be able to read the file easily. One very useful way to do this is to use blank lines liberally between paragraphs.

Some Special Characters

The characters $, %, &, #, _, and \ have special meaning to TEX and must be typed in a special way:

```
\$, \%, \&, \#, \_, and $\backslash$
```

There are three kinds of dashes: 1) the hyphen as in the phrase ‘math-in-text’, 2) the en-dash as in the phrase ‘read sections 10–12’, and 3) the em-dash as in the phrase ‘now is the time—I mean right now’. These three dashes are produced by typing one, two, or three minus signs, respectively.

You should note the quote marks in this document (‘like this’). The one on the right is familiar to everyone, but pay particular attention to the one on the left. It is different than the one on the right and there is a key on all keyboards that produces this left quote.

One can apply accents to any letter as in ò, ó, ô, ö, õ, ō, ȯ, ŏ, ǒ, ő, o͡o, o̧, ọ, o̲, and there are the special characters œ, Œ, æ, Æ, å, Å, ø, Ø, ł, Ł, ß. These are obtained by typing

```
\`o \'o \^o \"o \~o \=o \.o \u o \v o \H o \t oo \c o \d o \b o
```

and

```
\oe  \OE  \ae  \AE  \aa  \AA  \o  \O  \l  \L  \ss
```

The Item and Itemitem Macros

Several times in this chapter we have displayed lists of items (see the first page of the chapter for example). The items are characterized by the fact that all lines in an item are indented by the current value of the parameter `\parindent` and the first line of an item is preceded by an identifier (such as the 1, 2, and so on). For example,

1. Item 1.
   a. Subitem a.
   b. Subitem b.
2. Item 2.
3. Item 3.

is obtained by typing

```
\vskip 2mm\item{1.} Item 1.
\vskip 2mm\itemitem{a.} Subitem a.
\vskip 2mm\itemitem{b.} Subitem b.
\vskip 2mm\item{2.} Item 2.
\vskip 2mm\item{3.} Item 3.
```

Note that an item ‘stays active’ until a new paragraph is encountered (by a blank line, a `\vskip`, or another `item` or `itemitem`.

Typing Math

There are two ways that mathematical symbols appear in a manuscript. The first is when they are mixed into a regular line of text such as in ‘Let $S = X_1 + \cdots + X_n$, where $X_1, \ldots, X_n$ is a random sample from the distribution having pdf $f(x) = \lambda e^{-\lambda x}$ for $x > 0$,’ which is obtained by typing

```
`Let $S=X_1+\cdots+X_n$, where $X_1,\ldots,X_n$ is a
random sample from the distribution having pdf
$f(x)=\lambda e^{-\lambda x}$ for $x>0$,'
```

Thus 'math in text' is obtained by surrounding the math with single dollar signs. The second way math is used is when an expression is displayed on a separate line, that is, 'display math,' such as in

$$F(x)=\int_{-\infty}^x f(y)dy, \qquad \bar X=\sum_{i=1}^nX_i, \qquad \tilde X=\hbox{median}\{X_i\}, \qquad \hat\beta=S_{XY}/S_{XX},$$

which is obtained by surrounding the math in 'double dollar signs':

```
$$
F(x)=\int_{-\infty}^x f(y)dy, \qquad
\bar X=\sum_{i=1}^nX_i, \qquad
\tilde X=\hbox{\rm median}\{X_i\}, \qquad
\hat\beta=S_{XY}/S_{XX},
$$
```

These examples illustrate a number of important features of typing math:

1. They illustrate how to get subscripts (using the underscore character) and superscripts (using a caret). The important thing to notice is that if a subscript or a superscript is a single character, then one types the underscore or caret followed by the single character. If it is more than a single character, then the characters must be placed inside curly brackets following the underscore or caret (another example of grouping via the curly brackets). This makes it possible to get extremely complicated expressions very simply.

2. Integrals and summations are obtained similarly, with the lower and upper limits being obtained in the same way as subscripts and superscripts.

3. Notice the difference between `\ldots` and `\cdots` and when they are used.

4. Notice how $\bar X$, $\tilde X$, and $\hat\beta$ are typed.

5. Most mathematical functions (such as 'median') are typed in Roman font while all characters are by default produced in math italics (*like this*). Notice how we typed 'median.'

6. We were able to display in a readable way what we typed to get the displayed equation because of the fact that spaces that you type while in the math mode (except for those required after macros) are ignored by TEX. In order to get spaces, you have to tell TEX to put them in. For example, the macro `\qquad` leaves a 'qquad' of space, where a 'quad' is the width of a capital M and a 'qquad' is two quads.

Alignments

Many times one needs to line up a series of expressions, such as in

$$\begin{aligned}&f(x)=\dfrac{1}{\sqrt{2\pi}}e^{-x^2/2}\\ &F(x)=\int_{-\infty}^xf(y)dy,\end{aligned}$$

which is obtained from

```
$$
\eqalign{
   &f(x)=\dfrac{1}{\sqrt{2\pi}}e^{-x^2/2}\cr
   &F(x)=\int_{-\infty}^xf(y)dy,\cr
}
$$
```

or the same kind of thing but with equation numbers:

$$
\begin{aligned}
X(t) &= \sum_{k=1}^{n} a_k \cos \frac{2\pi(t-1)}{n} + b_k \sin \frac{2\pi(t-1)}{n} &\quad (2)\\
&= \sum_{k=1}^{n} c_k \cos \left(\frac{2\pi(t-1)}{n} - \phi_k\right).
\end{aligned}
$$

which one gets from

```
$$
\eqalignno{
  X(t)&=\sum_{k=1}^na_k\cos\dfrac{2\pi(t-1)}{n}+
                       b_k\sin\dfrac{2\pi(t-1)}{n}&(2)\cr
      &=\sum_{k=1}^nc_k\cos\left(\dfrac{2\pi(t-1)}{n}-\phi_k\right).
\cr}
$$
```

All you have to do is put an & before the characters you want lined up and then put a `\cr` after each line. Note that `eqalignno` is the same as `eqalign` except it allows equation numbers. There is an `leqaligno` too that puts equation numbers on the left.

In the last expression above, the outer parentheses are larger than the inner parentheses. Typing `\left` and `\right` before the outer left and right parentheses told TEX to figure out how large they should be. This can also be done with square and curly brackets.

Typing Cases

Often we have to define functions whose values vary according to different conditions as in

$$
(x+y)^n = \begin{cases} x^2+2xy+y^2, & \text{if } n=2 \\ x^3+3x^2y+3xy^2+y^3, & \text{if } n=3 \\ \sum_{j=0}^{n} \binom{n}{j} x^j y^{n-j}, & \text{in general.} \end{cases}
$$

which is obtained by typing

```
$$
(x+y)^n=\cases{
                x^2+2xy+y^2,&if $n=2$\cr
                x^3+3x^2y+3xy^2+y^3,&if $n=3$\cr
                \sum_{j=0}^n{n\choose j}x^jy^{n-j},&in general.
               }
$$
```

This is the `cases` command. Note that it is very similar to `eqalign` in that you put ampersands before the conditions for each case and a `\cr` at the end of each line.

Typing Matrices

Typing matrices is done similarly, for example,

$$
\mathbf{X} = \begin{bmatrix} X_{11} & X_{12} & \dots & X_{1p} \\ X_{21} & X_{22} & \dots & X_{2p} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ X_{n1} & X_{n2} & \dots & X_{np} \end{bmatrix}, \mathbf{L} = \begin{bmatrix} L_{11} & & & & \\ L_{21} & L_{22} & & & \\ \vdots & \vdots & \ddots & & \\ \vdots & \vdots & & \ddots & \\ L_{p1} & L_{p2} & \dots & \dots & L_{pp} \end{bmatrix}.
$$

This is obtained by

```
$$
\hbox{{\bf X}}=\left[\matrix{X_{11}&X_{12}&\ldots&X_{1p}\cr
                             X_{21}&X_{22}&\ldots&X_{2p}\cr
                             \vdots&\vdots&      &\vdots\cr
                             \vdots&\vdots&      &\vdots\cr
                             X_{n1}&X_{n2}&\ldots&X_{np}\cr}\right],
\hbox{{\bf L}}=\left[\matrix{L_{11}\cr
                             L_{21}&L_{22}\cr
                             \vdots&\vdots&\ddots\cr
                             \vdots&\vdots&      &\ddots\cr
                      L_{p1}&L_{p2}&\ldots&\ldots&L_{pp}\cr}\right].
$$
```

Thus you type each row of the matrix with ampersands between the elements and a `\cr` at the end of each line.

Log-Like Functions

All latin characters in the math modes are formed in math italics. It is conventional that mathematical functions such as log should be formed in Roman font. Thus TEX has the following macros for producing the desired Roman font for commonly used expressions:

`\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`

We can easily obtain similar effects for expressions not in the above list, for example, $\sigma^2=\sqrt{\mathrm{Var}(X)}$, which is obtained by typing

```
$\sigma^2=\sqrt{\hbox{\rm Var}(X)}$
```

rather than $\sigma^2=\sqrt{Var(X)}$. Some of the expressions in the table given above will achieve special effects when they have subscripts, for example:

$$\lim_{n\to\infty}a_n=0,\quad\min_{n\le k}X_n=10$$

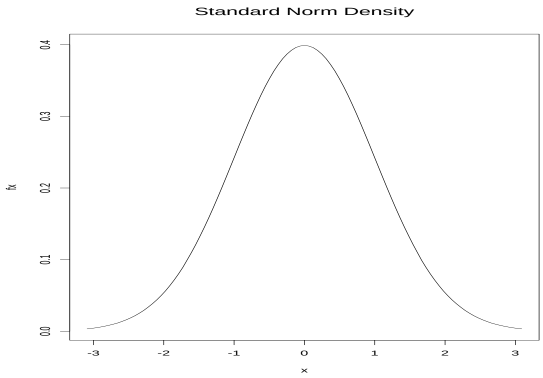
which is obtained from

```
$$
\lim_{n\to\infty}a_n=0,\quad\min_{n\le k}X_n=10
$$
```

Typing Tables

One of the more challenging parts of TEX is typing tables. For example, at the end of this chapter on a separate page is a table and the code that produced the table. Note that to get things on a separate page, one need only put `\pageinsert` at the beginning of the material and `\endinsert` at the end. There is also a `\topinsert` and `\midinsert`. Note the comment lines in the code.

Tables are formed using the `\halign` command. One creates a 'template' describing how each element of each line in the table is to appear in the table. In the template you have to say whether the elements of a column are to be left justified (by typing `#\hfil`; note that the `\hfil` being on the right of the `#` 'pushes' the entry (which is represented by the `#`) to the left), or right justified (by typing `\hfil#`) or centered (by typing `\hfil#\hfil`). Then each row of the table is entered with ampersands separating the entries and a `\cr` at the end of each row. While in the middle of the alignment process, one can use `\noalign` to put in horizontal lines and other effects (see the example).



Inserting Postscript Graphs

To insert postscript graphs into a document (such as the one at the top of this or the next page—with inserts one doesn't always know where the graph will go), one must insert the line

```
\include psfig
```

at the beginning of the file, and then lines such as

```
\topinsert
\centerline{\psfig{file=example.ps,height=2in,width=3in}}
\endinsert
\vskip 4mm
```

in the file.

Writing Your Own Macros

There are two main reasons to write your own macros. First, if you find yourself typing the same long character string over and over again, you can define a macro and then only type the macro. For example, a time series book would include the phrase 'covariance stationary time series' repeatedly. If you include the line

```
\def\CSTS{covariance stationary time series}
```

somewhere in your file, then you can type `\CSTS` anytime after you defined it. Perhaps the most important use of macros is when they have arguments. Here are three examples:

```
\def\sechd#1{\vskip 4mm\noindent{\bf#1}\vskip 4mm}
\def\dfrac#1#2{\displaystyle{{#1}\over{#2}}}
\def\binom#1#2{{{#1}\choose{#2}}}
```

The first example defines a section heading by skipping four millimeters but not indenting, then put the 'argument' of the macro in bold face, and then skip four more millimeters. The macro would be used by typing

```
\sechd{Writing Your Own Macros}
```

for example. Note how the argument is defined using the `#1`. The second example makes it easy to produce fractions, and the third produces binomial coefficients, for example, typing

```
$$
M_{nk}=\binom{n}{k/2},\qquad N_{nk}=\dfrac{M_{nk}}{n^3}
$$
```

gives

$$M_{nk} = \binom{n}{k/2}, \qquad N_{nk} = \frac{M_{nk}}{n^3}$$

Lining Things Up

One often needs to create special forms where things are aligned in unusual ways. One obtains

MEMORANDUM

TO:	John Doe
FROM:	Fred Smith
SUBJECT:	Making Forms

John Doe Fred Smith

by typing

```
\def\hitem#1#2#3{\line{\hbox to #1{#2\hfil}#3\hfil}}
\noindent MEMORANDUM\vskip 2mm
\hitem{1in}{TO:}{John Doe}
\hitem{1in}{FROM:}{Fred Smith}
\hitem{1in}{SUBJECT:}{Making Forms}

\vskip 6mm\noindent
\line{\hbox to 2in{\hrulefill}\hfil\hbox to 2in{\hrulefill}}
\line{\hbox to 2in{\hfil John Doe\hfil}\hfil
      \hbox to 2in{\hfil Fred Smith\hfil}}
```

Running Headers

Here is what I used in this report to do the running headline:

```
\newcount\pgno

\headline={\ifnum\pageno=\pgno\blank \else\ifodd\pageno\rhl
 \else\lhl\fi\fi}

\footline={\ifnum\pageno=\pgno\hss\tenrm\folio\hss
  \else\nopagenumbers\fi}

\def\blank{\tenrm\hfil {}\hfil}
\def\lhl{\tenrm\folio\hfil {\smc Intro to \TeX}\hfil {\smc ch.} ??}
\def\rhl{\tenrm {\smc sec} ??\hfil {\smc Intro to \TeX}\hfil\folio}
```

List of Special Symbols

There are a host of special symbols that TEX can produce. These symbols fall into several categories and are listed below.

Lower Case Greek Letters

`\alpha`	α	`\iota`	ι	`\varrho`	ϱ
`\beta`	β	`\kappa`	κ	`\sigma`	σ
`\gamma`	γ	`\lambda`	λ	`\varsigma`	ς

`\delta`	δ	`\mu`	μ	`\tau`	τ
`\epsilon`	ϵ	`\nu`	ν	`\upsilon`	υ
`\varepsilon`	ε	`\xi`	ξ	`\phi`	ϕ
`\zeta`	ζ	`o`	o	`\varphi`	φ
`\eta`	η	`\pi`	π	`\chi`	χ
`\theta`	θ	`\varpi`	ϖ	`\psi`	ψ
`\vartheta`	ϑ	`\rho`	ρ	`\omega`	ω

Upper Case Greek Letters

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

Calligraphic Capitals

In the math mode, typing `\cal A`, `\cal B`, and so on gives

$$\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D},\mathcal{E},\mathcal{F},\mathcal{G},\mathcal{H},\mathcal{I},\mathcal{J},\mathcal{K},\mathcal{L},\mathcal{M},\mathcal{N},\mathcal{O},\mathcal{P},\mathcal{Q},\mathcal{R},\mathcal{S},\mathcal{T},\mathcal{U},\mathcal{V},\mathcal{W},\mathcal{X},\mathcal{Y},\mathcal{Z}.$$

Miscellaneous Symbols of Type Ord

`\aleph`	$\aleph$	`\prime`	$\prime$	`\forall`	$\forall$
`\hbar`	$\hbar$	`\emptyset`	$\emptyset$	`\exists`	$\exists$
`\imath`	$\imath$	`\nabla`	∇	`\neg`	$\neg$
`\jmath`	$\jmath$	`\surd`	$\surd$	`\flat`	$\flat$
`\ell`	ℓ	`\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`	$\bigwedge$		

These symbols are bigger in the display math mode than in the math within text mode, and the limits on them are treated differently, for example, $\sum_{j=1}^n$ and $\int_0^1$ versus

$$\sum_{j=1}^n \qquad \int_0^1 .$$

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	$\perp$

Many of these symbols can be negated by typing `\not` in front of the command for a symbol, for example, typing `\not\leq` gives $\not\leq$.

Arrows

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

By typing `\buildrel\alpha \beta \over \longrightarrow` one gets $\stackrel{\alpha\beta}{\longrightarrow}$.

Table 1.1. Distributions in the `WN` Command

dist	Distribution	Mean	Variance
1	$N(0,1):\ Z$	0	1
2	$U(0,1):\ U$	$1/2$	$1/12$
3	Exponential: $-\ln(1-U)$	1	1
4	Logistic: $\ln(U/1-U)$	0	$\pi^2/3=3.28987$
5	Cauchy: $\tan(\pi(U-\dfrac{1}{2}))$	∞	∞
6	Extreme Value: $\ln(-\ln(1-U))$	$-\gamma=-.5772$	$\pi^2/6=1.6449$
7	Lognormal: e^Z	$e^{1/2}=1.6487$	$e^2-e=4.67077$
8	Double Exponential	0	2
	$\begin{cases}\ln(2U), & U\le .5\\ -\ln(2(1-U)), & U>.5\end{cases}$		

```
\centerline{{\bf Table 1.1.} Distributions in the {\tt WN} Command}
%
%   Skip a little space, draw the dark line and skip a little more:
%
\vskip 2mm\hrule height1pt\vskip 2mm
%
%   The next line sets up space between entries in table.
%
\tabskip=1em plus2em minus.5em
%
%   Template (1st, 3rd, 4th centered, 2nd left justified):
%
\halign to \hsize{\hfil#\hfil&#\hfil&\hfil#\hfil&\hfil#\hfil\cr
%
%   The column headings and the line under the heading:
%
dist&Distribution&Mean&Variance\cr
\noalign{\vskip 2mm\hrule\vskip 2mm}
%
%   Now enter each line of the table:
%
1 & $N(0,1):\ Z$& 0 & 1\cr
2 & $U(0,1):\ U$& $1/2$& $1/12$\cr
3 & Exponential: $-\ln(1-U)$& 1 &1\cr
4 & Logistic: $\ln(U/1-U)$& 0 & $\pi^2/3=3.28987$\cr
5 & Cauchy: $\tan(\pi(U-\dfrac12))$ & $\infty$ & $\infty$ \cr
6 & Extreme Value: $\ln(-\ln(1-U))$ & $-\gamma=-.5772$
& $\pi^2/6=1.6449$ \cr
7 & Lognormal: $e^Z$ & $e^{1/2}=1.6487$ & $e^2-e=4.67077$ \cr
8 & Double Exponential & 0 & 2\cr
  & $\cases{\ln(2U),&$ U\le.5$\cr-\ln(2(1-U)),&$U>.5$\cr}$ & &\cr
%
%   End the \halign with the }; draw the line under the table:
%
}
\vskip 2mm\hrule
```