

Introduction to T_EX

for use in ST 8353 “Statistical Computations”

Fall, 2003

by

Jane L. Harvill*

T_EX (pronounced “tek”) is a typesetting language and compiler for creating beautiful manuscripts, particularly those containing mathematics. Most of the documents for this course, including this one, was created using T_EX, or a popular variant of T_EX. While there is a wide variety of programs for document preparation, T_EX has a number of advantages, including a) it is used all over the world on all types of computers, b) to use T_EX 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 T_EX are superior in quality than those created by other software.

In this report, I hope to introduce you to how T_EX can be used to produce mathematical manuscripts. The basic reference for the use of T_EX is the *T_EXbook* by Donald Knuth (the author of T_EX). Attached to this document is a listing of the ASCII file that was typed to produce it. In this manuscript, we discuss a number of features, while the listing in the Appendix gives a number of other details.

1. Introduction

T_EX 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. To produce a mathematical manuscript such as this one, the user must perform four steps which are analogous to compiling and linking a program using a language like Fortran or C++:

- 1) Use a word processor or text editor to produce a file that the T_EX program can process.
- 2) Use the T_EX program to process the T_EX file that was created in step (1). If there are no mistakes in the T_EX 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. If there are mistakes in the T_EX file, the T_EX program will describe what they are. The user must then return to step (1) to correct the mistakes.
- 3) Most computers have a previewer program that can be used to see a low resolution version of the final document on their monitor for a final screening. Once there are no more mistakes in the T_EX file, the user executes a “printer driver” program which converts the DVI file to another file which is in a form that is understandable to the particular printer that will be used for printing. In many cases, this printer driver program will also actually send the final file to the printer.
- 4) 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 (3) to the printer.

This report does not discuss any details of how these four steps are done on any particular computer system as they are dependent on what kind of computer and printer are being used. The reader should consult their local T_EX expert for details on how to do the steps on their machine.

Commands and Macros

To typeset ordinary text, the user merely types the text and T_EX determines line and page breaks automatically. To tell T_EX to do something other than just process text, one types what is called a ‘command’

* Adapated from Document by H. Joseph Newton, Department of Statistics, Texas A&M University

At any one time, T_EX 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, which is in fact what are used in this report. 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. It is also possible to define a font of a size other than the three standard sizes to be used for special situations. An example of this is the title of this report and the reader can consult the listing to see how this was done.

By default, T_EX leaves a one inch margin all around the page. This is controlled by the commands `\hsize`, `\vsize`, `\hoffset`, and `\voffset`, which are 6.5, 8.9, 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`.

By default, T_EX 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, set `\pageno` equal to a negative value. One can also place the page numbers somewhere else and get a running headline.

Hanging Indents

Often one needs to establish temporary indented margins so that a paragraph will be set off visually (such as in a bibliography for example). This paragraph is an example of such hanging indenting. The default paragraph indent is 20 points and is controlled by the command `\parindent`.

2. Examples of Typing Math

In this section, we produce a series of examples of commonly used mathematical expressions. See Appendix 1 for a list of all of the special mathematical symbols that T_EX can produce and how they are typed.

Subscripts and Superscripts

Subscripts and superscripts are easy to obtain, for example,

$$X_1^2, \dots, X_n^2 \quad \hat{\omega}_{\hat{p}} \quad e^{-x^2/2} \quad X_{n_i}$$

The basic rule is that the underscore and carat symbols are used to signal that what comes next is a sub or super script. If it is a single character, then one just types the character. Otherwise, a series of characters to be sub or super scripts are grouped within curly brackets.

Log-Like Functions

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

<code>\arccos</code>	<code>\cos</code>	<code>\csc</code>	<code>\exp</code>	<code>\ker</code>	<code>\limsup</code>	<code>\min</code>	<code>\sinh</code>
<code>\arcsin</code>	<code>\cosh</code>	<code>\deg</code>	<code>\gcd</code>	<code>\lg</code>	<code>\ln</code>	<code>\Pr</code>	<code>\sup</code>
<code>\arctan</code>	<code>\cot</code>	<code>\det</code>	<code>\hom</code>	<code>\lim</code>	<code>\log</code>	<code>\sec</code>	<code>\tan</code>
<code>\arg</code>	<code>\coth</code>	<code>\dim</code>	<code>\inf</code>	<code>\liminf</code>	<code>\max</code>	<code>\sin</code>	<code>\tanh</code>

We can easily obtain similar effects for expressions not in the above list, for example, $\sigma^2 = \sqrt{\text{Var}(X)}$ rather than $\sigma^2 = \sqrt{\text{Var}(\bar{X})}$. Some of the expressions in the table given above will achieve special effects when they have subscripts, for example:

$$\lim_{n \rightarrow \infty} a_n = 0, \quad \min_{n \leq k} X_n = 10.$$

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: $\tan(\pi(U - \frac{1}{2}))$	∞	∞
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 $\begin{cases} \ln(2U), & U \leq .5 \\ -\ln(2(1 - U)), & U > .5 \end{cases}$	0	2

Aligning Expressions

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

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

$$F(x) = \int_{-\infty}^x f(y) dy,$$

or

$$X(t) = \sum_{k=1}^n a_k \cos \frac{2\pi(t-1)}{n} + b_k \sin \frac{2\pi(t-1)}{n} \quad (2)$$

$$= \sum_{k=1}^n c_k \cos \left(\frac{2\pi(t-1)}{n} - \phi_k \right).$$

In this last expression, the outer parentheses are larger than the inner parentheses. If you look at the code that generated this expression, you will see that typing `\left` and `\right` before the outer left and right parentheses told T_EX 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

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

Matrices

One of the best features of T_EX is how simple it is to produce matrices such as

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

3. Other Special Effects

There are a whole host of other special things that one can use T_EX to do. In this section we illustrate some of them.

Insertions

Many times one wants to insert tables or figures into a paper and have them appear either on a page by themselves or at the top of a page. T_EX has macros called `pageinsert` and `topinsert` respectively for these purposes. For example, at the top of one of the pages near here in the report is a table that appears in my time series book.

Tables

Creating tables is an important and challenging part of using T_EX. There are really two ways to create tables, but the easiest to use is the method that uses the `halign` command in T_EX. In this method, one first creates a template for each column of the table, and then enters each row with ampersands separating the elements of the row.

Theorems and Such

We often need the ability to do things such as

Theorem 1. *If X_1, \dots, X_n is a random sample from a distribution having mean μ and variance σ^2 , then*

$$\frac{\sqrt{n}(\bar{X} - \mu)}{\sigma} \xrightarrow{\mathcal{D}} N(0, 1).$$

Forms

One often needs to create special forms as in

MEMORANDUM

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

or

John Doe

Fred Smith

KNUTH, DONALD E. (1970) *The T_EXbook*. Addison Wesley Publishing Company, Reading Massachusetts.

Appendix 1: List of Special Symbols

Lower Case Greek Letters

<code>\alpha</code>	α	<code>\iota</code>	ι	<code>\varrho</code>	ϱ
<code>\beta</code>	β	<code>\kappa</code>	κ	<code>\sigma</code>	σ
<code>\gamma</code>	γ	<code>\lambda</code>	λ	<code>\varsigma</code>	ς
<code>\delta</code>	δ	<code>\mu</code>	μ	<code>\tau</code>	τ
<code>\epsilon</code>	ϵ	<code>\nu</code>	ν	<code>\upsilon</code>	υ
<code>\varepsilon</code>	ε	<code>\xi</code>	ξ	<code>\phi</code>	ϕ
<code>\zeta</code>	ζ	<code>\omicron</code>	\omicron	<code>\varphi</code>	φ
<code>\eta</code>	η	<code>\pi</code>	π	<code>\chi</code>	χ
<code>\theta</code>	θ	<code>\varpi</code>	ϖ	<code>\psi</code>	ψ
<code>\vartheta</code>	ϑ	<code>\rho</code>	ρ	<code>\omega</code>	ω

Upper Case Greek Letters

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

Calligraphic Capitals

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

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

Miscellaneous Symbols of Type Ord

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

Large Operators

<code>\sum</code>	\sum	<code>\bigcap</code>	\bigcap	<code>\bigodot</code>	\bigodot
<code>\prod</code>	\prod	<code>\bigcup</code>	\bigcup	<code>\bigotimes</code>	\bigotimes
<code>\coprod</code>	\coprod	<code>\bigsqcup</code>	\bigsqcup	<code>\bigoplus</code>	\bigoplus
<code>\int</code>	\int	<code>\bigvee</code>	\bigvee	<code>\biguplus</code>	\biguplus
<code>\oint</code>	\oint	<code>\bigwedge</code>	\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

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

Binary Operations

<code>\pm</code>	±	<code>\cap</code>	∩	<code>\vee</code>	∨
<code>\mp</code>	∓	<code>\cup</code>	∪	<code>\wedge</code>	∧
<code>\setminus</code>	\	<code>\uplus</code>	⊕	<code>\oplus</code>	⊕
<code>\cdot</code>	·	<code>\sqcap</code>	⊓	<code>\ominus</code>	⊖
<code>\times</code>	×	<code>\sqcup</code>	⊔	<code>\otimes</code>	⊗
<code>\ast</code>	*	<code>\triangleleft</code>	◁	<code>\oslash</code>	⊘
<code>\star</code>	*	<code>\triangleright</code>	▷	<code>\odot</code>	⊙
<code>\diamond</code>	◇	<code>\wr</code>	ℳ	<code>\dagger</code>	†
<code>\circ</code>	○	<code>\bigcirc</code>	◯	<code>\ddagger</code>	‡
<code>\bullet</code>	•	<code>\bigtriangleup</code>	△	<code>\amalg</code>	∐
<code>\div</code>	÷	<code>\bigtriangledown</code>	▽		

Relations

<code>\leq</code>	≤	<code>\geq</code>	≥	<code>\equiv</code>	≡
<code>\prec</code>	≺	<code>\succ</code>	≻	<code>\sim</code>	≈
<code>\preceq</code>	≼	<code>\succeq</code>	≽	<code>\simeq</code>	≈
<code>\ll</code>	≪	<code>\gg</code>	≫	<code>\asymp</code>	≈
<code>\subset</code>	⊂	<code>\supset</code>	⊃	<code>\approx</code>	≈
<code>\subseteq</code>	⊆	<code>\supseteq</code>	⊇	<code>\cong</code>	≅
<code>\sqsubseteq</code>	⊓	<code>\sqsupseteq</code>	⊔	<code>\bowtie</code>	⋈
<code>\in</code>	∈	<code>\ni</code>	∋	<code>\propto</code>	∝
<code>\vdash</code>	⊢	<code>\dashv</code>	⊥	<code>\models</code>	⊨
<code>\smile</code>	⤴	<code>\mid</code>		<code>\doteq</code>	⋮
<code>\frown</code>	⤵	<code>\parallel</code>	∥	<code>\perp</code>	⊥

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

<code>\leftarrow</code>	←	<code>\longleftarrow</code>	⇐	<code>\uparrow</code>	↑
<code>\Leftarrow</code>	⇐	<code>\Lleftarrow</code>	⇐	<code>\Uparrow</code>	⇑
<code>\rightarrow</code>	→	<code>\longrightarrow</code>	⇒	<code>\downarrow</code>	↓
<code>\Rightarrow</code>	⇒	<code>\Longrightarrow</code>	⇒	<code>\Downarrow</code>	⇓
<code>\leftrightarrow</code>	↔	<code>\longleftrightarrow</code>	↔	<code>\updownarrow</code>	↕
<code>\Leftrightarrow</code>	⇔	<code>\Longleftrightarrow</code>	⇔	<code>\Updownarrow</code>	⇕
<code>\mapsto</code>	↦	<code>\longmapsto</code>	↦	<code>\nearrow</code>	↗
<code>\hookrightarrow</code>	↪	<code>\hookrightarrow</code>	↪	<code>\searrow</code>	↘
<code>\leftharpoonup</code>	↵	<code>\rightharpoonup</code>	↶	<code>\swarrow</code>	↙
<code>\leftharpoondown</code>	↶	<code>\rightharpoondown</code>	↷	<code>\nwarrow</code>	↖
<code>\rightleftharpoons</code>	⇌				

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

```

%-----
%
% intro.tex: TeX file containing
%           ``Introduction to TeX (September, 1992)''
%           by H. Joseph Newton.
%
% Adapted and adopted for ST8353 by Jane L. Harvill
% August, 2000
%
% Modified: 08/21/2003 JLH
%-----
%
% Note that anything after a % on a line is ignored and thus
% we can insert comments into a TeX file.
%-----
%
% Define some fonts in special sizes for special uses:
% (The others are cmr10 (Roman), cmti10 (Italics), cmsl10 (Slanted))
%-----
%
\font\bfone=cmbx10 scaled\magstep1 % used in sechd macro
\font\bftwo=cmbx10 scaled\magstep2 % used in title
\font\tteight=cmtt8                % used in listing macro
\font\smc=cmcsc10                  % small caps
%-----
%
% Some macros:
%
% These have no arguments. Whenever TeX sees one of these macros
% it will replace the name (e.g. \IE) by what it is defined to be.
%
% Warnings: 1. macro names can only contain a-z and A-Z.
%           2. try not to use the same name as something in the
%              TeXbook (look in its index)
%           3. One is tempted to define lots and lots of macros
%              and ends up forgetting what they are called.
%-----
%
\def\IE{that is,\ }
\def\EG{for example,\ }
\def\tbs{${\backslash$}
\def\tbsk{\noalign{\vskip 2mm}}
%-----
%
% Here's how to get the page numbers and headline at the top of the
% page. If you just want pagenumbers centered at the bottom of the pages,
% just leave this stuff out. If you don't want any pagenumbers, leave this
% stuff out and put the command \nopagenumbers at the beginning of the
% TeX file.
%-----
%
\def\blank{\hfil{ }\hfil}
\headline={\ifnum\pageno=1\blank \else\ifodd\pageno\rhl

```

```

\else\lhl\fi\fi}
\footline={\blank}
\def\lhl{\tenrm\folio\hfil {\smc Introduction to \TeX}\hfil { }}
\def\rhl{\tenrm { }}\hfil {\smc Introduction to \TeX}\hfil\folio}

```

```

%-----
%
% Here are some macros that have an argument(s):
%
% For example, when you type \dfrac{x}{n}, whatever is in the
% first (second) pair of curly brackets replaces #1 (#2).
%
%-----

```

```

\def\sechd#1{\vskip 4mm\noindent{\bfone#1}\vskip 3mm}
\def\sbhd#1{\vskip 4mm\noindent{\bf#1}\vskip 3mm}
\def\tsl#1{{\tt#1}}
\def\dspl#1{\vskip 4mm{\tt#1}\vskip 4mm\noindent\ignorespaces}
\def\dfrac#1#2{{\#1}\over{\#2}}

```

```

%-----
%
% START THE REPORT:
%
%-----

```

```

\centerline{\bftwo Introduction to \TeX}
\vskip 4mm
\centerline{\bfone for use in ST 8353 ``Statistical Computations''}
\vskip 4mm
\centerline{\bfone Fall, 2003}

\vskip 6mm
\centerline{by}
\vskip 4mm

```

```

%-----
%
% Notice how footnotes are done:
%
%-----

```

```

\centerline{Jane L.~Harvill\footnote*{Adaped from Document by
H.~Joseph Newton, Department of Statistics, Texas A\&M University}}
\vskip 10mm

```

\TeX (pronounced ``tek'') is a typesetting language and compiler for creating beautiful manuscripts, particularly those containing mathematics. Most of the documents for this course, including this one, was created using \TeX , or a popular variant of \TeX . While there is a wide variety of 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 report, I hope to introduce you to how `\TeX` can be used to produce mathematical manuscripts. The basic reference for the use of `\TeX` is the `{\it \TeX book}` by Donald Knuth (the author of `\TeX`). Attached to this document is a listing of the ASCII file that was typed to produce it. In this manuscript, we discuss a number of features, while the listing in the Appendix gives a number of other details.

`\sechd{1. Introduction}`

`\TeX` is not a WYSIWYG (what you see is what you get) program, `\IE` the user does not see what the paper will look like until it is printed or previewed on the computer screen. To produce a mathematical manuscript such as this one, the user must perform four steps which are analogous to compiling and linking a program using a language like Fortran or C++:

```
%-----  
%  
%   There is an \itemitem macro too which lets you put items within items.  
%  
%-----
```

`\vskip 4mm`

`\item{1}`

Use a word processor or text editor to produce a file that the `\TeX` program can process.

`\item{2}`

Use the `\TeX` program to process the `\TeX` file that was created in step (1). If there are no mistakes in the `\TeX` file, this step will produce a file called a 'DVI' file since it is 'device independent', `\IE` it will be the same no matter what computer one is using. If there are mistakes in the `\TeX` file, the `\TeX` program will describe what they are. The user must then return to step (1) to correct the mistakes.

`\item{3}`

Most computers have a previewer program that can be used to see a low resolution version of the final document on their monitor for a final screening. Once there are no more mistakes in the `\TeX` file, the user executes a 'printer driver' program which converts the DVI file to another file which is in a form that is understandable to the particular printer that will be used for printing. In many cases, this printer driver program will also actually send the final file to the printer.

`\item{4}`

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 (3) to the printer.

`\vskip 4mm`

This report does not discuss any details of how these four steps are done on any particular computer system as they are dependent on what kind of computer and printer are being used. The reader should consult their local `\TeX` expert for details on how to do the steps on their machine.

`\sbhd{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), `\IE` a backslash (like this: `\backslash`), 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 `\tsl{\backslashTeX}` produces `\TeX`. The names of macros contain only latin letters (no numbers 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 (`\tsl{like this}`).
`\par` To tell `\TeX` to end the current paragraph, one issues the command `\tsl{\backslashpar}` 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.

`\sbhd{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 'the exponential distribution has pdf $f(x) = \lambda e^{-\lambda x}$ for $x > 0$.' The second way is when an expression is displayed on a separate line, such as in

$$F(x) = \int_{-\infty}^x f(y) dy.$$

These two ways are called the 'math-in-text' and 'displayed math' modes respectively. To produce such mathematical symbols, one must signal to `\TeX` that these modes are desired by enclosing math within text in single dollar signs and displayed math in double dollar signs. Section 2 of this report contains a series of examples of typing math.

`\sbhd{Some Special Characters}`

The characters `\$, \% , \& , \# , _ ,` and `\backslash` have special meaning to `\TeX` and must be typed in a special way. 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'. One can apply accents to any letter as in `\`o , \'o , \^o , \~o , \=o , \.o , \u o , \v o , \H o , \t oo , \c o , \d o , \b o`, and there are the special characters `\oe , \OE , \ae , \AE , \aa , \AA , \o , \O , \l , \L , \ss`.

`\sbhd{Fonts, Font Sizes, Page Sizes, and Page Numbers}`

The characters that are formed in the text mode are in what is called ten point 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 `\tsl{\backslashbaselineskip=14pt}`. In the math modes, latin characters are in what is called math italics (`$like\ this$`), and all blank spaces in the input file are ignored. In the text mode, more than one space in the input file is treated as a single space. One can get spaces in the math mode; for example:

```
$$
single\ space\ between\ words,
\ medium\quad space,
\ or\qquad larger\qquad space.
$$
```

We can also easily obtain italics in text (`{\it like this}`), or bold face within text (`{\bf like this}`), or any combination of Roman, `{\it italics}`, `{\bf bold face}`, `{\tt typewriter}`, `{\sl slanted}`, or `{\smc small caps}`. We can also get text within math as in the word 'for' (which would ordinarily be in math italics and look lousy) in

```
%
%-----
%
% \eqno will put an equation number on the right. \leqno
% will put it at the left. If you don't want an equation
% numbered just don't put in the \eqno or \leqno.
%
%-----
%
$$
F(x)=\int_{-\infty}^{\infty}xf(y)dy,\quad\hbox{for }x\in(-\infty,\infty).\leqno(1)
$$
```

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, which is in fact what are used in this report. 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. It is also possible to define a font of a size other than the three standard sizes to be used for special situations. An example of this is the title of this report and the reader can consult the listing to see how this was done.

By default, `\TeX` leaves a one inch margin all around the page. This is controlled by the commands `{\tt \tbs hsize, \tbs vsize, \tbs hoffset,}` and `{\tt \tbs voffset}`, which are 6.5, 8.9, 0.0, and 0.0 by default. To have a 5 by 7 inch page, still centered on the paper, one would use `{\tt \tbs hsize=5in, \tbs vsize=7in, \tbs hoffset=0.75in, \tbs voffset=1in}`.

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) `{\tt \tbs pageno=10}`. To get Roman numerals, set `{\tt \tbs pageno}` equal to a negative value. One can also place the page numbers somewhere else and get a running headline.

`\sbhd{Hanging Indents}`

```
%-----
%
% the indenting starts at the line after whatever is specified by
% \hangafter and continues until a new paragraph begins. Note
% that a \vskip acts like a \par. You can do references by
```

```
% letting \hangafter=1.
```

```
%
```

```
%-----
```

```
\hangindent 60pt\hangafter=0
```

Often one needs to establish temporary indented margins so that a paragraph will be set off visually (such as in a bibliography for example). This paragraph is an example of such hanging indenting. The default paragraph indent is 20 points and is controlled by the command `{\tt \tbs parindent}`.

```
\sechd{2. Examples of Typing Math}
```

In this section, we produce a series of examples of commonly used mathematical expressions. See Appendix 1 for a list of all of the special mathematical symbols that `\TeX` can produce and how they are typed.

```
\sbhd{Subscripts and Superscripts}
```

Subscripts and superscripts are easy to obtain, `\EG`

```
$$
```

```
X_1^2, \ldots, X_n^2 \qquad \hat{\omega}_p \qquad
```

```
e^{-x^2/2} \qquad X_{n_i}
```

```
$$
```

The basic rule is that the underscore and carat symbols are used to signal that what comes next is a sub or super script. If it is a single character, then one just types the character. Otherwise, a series of characters to be sub or super scripts are grouped within curly brackets.

```
\sbhd{Log-Like Functions}
```

All latin characters are formed in italics while in the math mode. 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:

```
\vskip 4mm
```

```
%-----
```

```
%
```

```
% This is an example of how a table is formed. There's a better
```

```
% example below.
```

```
%
```

```
%-----
```

```
\tabskip=1em plus2em minus.5em
```

```
\halign to\hsize{\tsl{\backslash$#}\hfil&\tsl{\backslash$#}\hfil
```

```
&\tsl{\backslash$#}\hfil&\tsl{\backslash$#}\hfil
```

```
&\tsl{\backslash$#}\hfil&\tsl{\backslash$#}\hfil
```

```
&\tsl{\backslash$#}\hfil&\tsl{\backslash$#}\hfil\cr
```

```
arccos&cos&csc&exp&ker&limsup&min&sinh\cr
```

```
arcsin&cosh&deg&gcd&lg&ln&Pr&sup\cr
```

```
arctan&cot&det&hom&lim&log&sec&tan\cr
```

```
arg&coth&dim&inf&liminf&max&sin&tanh\cr}
```

```
\vskip 4mm
```

We can easily obtain similar effects for expressions not in the above list, `\EG` $\sigma^2 = \sqrt{\text{Var}(X)}$ rather than $\sigma^2 = \sqrt{\text{Var}(X)}$. Some of the expressions in the table given above will achieve special effects when they have subscripts, for example:

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

```
\sbhd{Aligning Expressions}
```

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

```
%
%-----
%
% All you have to do is put an & before the characters you want
% lined up and then put a \cr after each line.
%
% \eqalignno is the same as \eqalign except it has equation
% numbers. There is an \leqalignno too.
%
%-----
$$
```

```
\eqalign{
  &f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \cr
  &F(x) = \int_{-\infty}^x f(y) dy, \cr
}
$$
or
$$
\eqalignno{
  X(t) &= \sum_{k=1}^n a_k \cos \frac{2\pi(t-1)}{n} +
        b_k \sin \frac{2\pi(t-1)}{n} &(2) \cr
  &= \sum_{k=1}^n c_k \cos \left( \frac{2\pi(t-1)}{n} - \phi_k \right). \cr
}
$$
```

In this last expression, the outer parentheses are larger than the inner parentheses. If you look at the code that generated this expression, you will see that typing `\tsl{\tbs left}` and `\tsl{\tbs 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.

```
\sbhd{Typing Cases}
```

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

```
%
%-----
%
% Here you put ampersands before the conditions and they will
% line up. Note that the conditions are in the text mode.
%
%-----
$$
(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.
}

```

\$\$

```

%-----
%
% Here's an example of typing a table (as well as \topinsert
% and \endinsert).
%
%-----
%
\topinsert
%
%-----
%
% The title of the table:
%
%-----
%
\centerline{{\bf Table 1.1.} Distributions in the \tsl{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. Just
% use this same line and it should look OK.
%
%-----
%
\tabskip=1em plus2em minus.5em
%
%-----
%
% The next line is the template. After the \halign to \hsize
% (the \hsize is the width of a page) one describes the
% appearance of each element of a row of the table, separated
% by ampersands. The first element is \hfil#\hfil. Putting an
% \hfil on each side of the # tells TeX to center the elements
% in the first column. The second element is left justified
% (pushed to the left) which is accomplished by putting an \hfil
% only on the right of the #.
%
% At the end of the template is a \cr
%
% Then the elements of each row of the table are entered, separated
% by ampersands, and ending with a \cr.
% Note that \tbsk is a macro defined in intromacs.tex and is
% used to get a little more space between rows than normal.
% Note also the line \noalign{\hrule\vskip 2mm} which draws
% the second horizontal line (rule) in the table.
%
%

```

```

% When the whole table has been typed, the } which matches the
% { after \halign to \hsize must be typed. Finally, \hrule is
% used to draw the horizontal rule at the bottom of the table.
%
%-----
%
\halign to \hsize{\hfil#\hfil&#\hfil&\hfil#\hfil&\hfil#\hfil\cr
dist&Distribution&Mean&Variance\cr \tbsk
\noalign{\hrule\vskip 2mm}
1 & $N(0,1):\ Z$& 0 & 1\cr \tbsk
2 & $U(0,1):\ U$& $1/2$& $1/12$\cr \tbsk
3 & Exponential: $-\ln(1-U)$& 1 & 1\cr \tbsk
4 & Logistic: $\ln(U/1-U)$& 0 & $\pi^2/3=3.28987$\cr \tbsk
5 & Cauchy: $\tan(\pi(U-\dfrac{1}{2}))$ & $\infty$ & $\infty$ \cr\tbsk
6 & Extreme Value: $\ln(-\ln(1-U))$ & $-\gamma=-.5772$
& $\pi^2/6=1.6449$ \cr\tbsk
7 & Lognormal: $e^Z$ & $e^{1/2}=1.6487$ & $e^2-e=4.67077$ \cr\tbsk
8 & Double Exponential & 0 & 2\cr\tbsk
& $\cases{\ln(2U),\& $ U\le .5$\cr-\ln(2(1-U)),\& $U>.5$\cr}$ & & \cr
}
\vskip 2mm\hrule
\endinsert

```

```
\sbhd{Matrices}
```

One of the best features of \TeX is how simple it is to produce matrices such as

```

%
%-----
%
% To form a matrix, list the elements of each row separated
% by ampersands and put a \cr at the end of each row.
% Note how the \left[ and \right] are used to get the border
% of the matrix.
%
% Note that \hbox{\bf X} gets the boldface Roman X. If you did
% \def\BX{\hbox{\bf X}} you could just type \BX instead.
%
%-----
%
$$
\hbox{\bf X}=\left[\matrix{X_{11}&X_{12}&\ldots&X_{1p}\cr
X_{21}&X_{22}&\ldots&X_{2p}\cr
\vdots&\vdots&&\vdots\cr
\vdots&\vdots&&\vdots\cr
X_{n1}&X_{n2}&\ldots&X_{np}\cr}\right],
\hbox{\bf L}=\left[\matrix{L_{11}\cr
L_{21}&L_{22}\cr
\vdots&\vdots&\ddots\cr
\vdots&\vdots&&\ddots\cr
L_{p1}&L_{p2}&\ldots&\ldots&L_{pp}\cr}\right].
$$

```

```
\sechd{3. Other Special Effects}
```

There are a whole host of other special things that one can use \TeX to do.

In this section we illustrate some of them.

```
\sbhd{Insertions}
```

Many times one wants to insert tables or figures into a paper and have them appear either on a page by themselves or at the top of a page. `\TeX` has macros called `\tsl{pageinsert}` and `\tsl{topinsert}` respectively for these purposes. For example, at the top of one of the pages near here in the report is a table that appears in my time series book.

```
\sbhd{Tables}
```

Creating tables is an important and challenging part of using `\TeX`. There are really two ways to create tables, but the easiest to use is the method that uses the `\tsl{halign}` command in `\TeX`. In this method, one first creates a template for each column of the table, and then enters each row with ampersands separating the elements of the row.

```
\sbhd{Theorems and Such}
```

We often need the ability to do things such as

```
\proclaim Theorem 1. If  $X_1, \dots, X_n$  is a random sample from a
distribution having mean  $\mu$  and variance  $\sigma^2$ , then

$$\frac{\sqrt{n}(\bar{X} - \mu)}{\sigma} \xrightarrow{D} N(0,1).$$

```

```
\sbhd{Forms}
```

```
\def\hitem#1#2#3{\line{\hbox to #1{#2\hfil}\hfil}}
```

One often needs to create special forms as in

```
\vskip 10mm\noindent MEMORANDUM\vskip 2mm
\hitem{1in}{TO:}{John Doe}
\hitem{1in}{FROM:}{Fred Smith}
\hitem{1in}{SUBJECT:}{Making Forms}
\vskip 6mm\noindent
or
\vskip 6mm
\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}}
```

```
%-----
%
% Here's how to eject the current page and go to the top of the next:
%
%-----
\line{\hbox to 6.5in{\hrulefill}}
\noindent
{\smc Knuth, Donald E.} (1970) {\it The \TeX book}. Addison Wesley
```

Publishing Company, Reading Massachusetts.

`\vfil\ejct`

```
%-----  
%  
%   Now get Appendix 1 from INTROAPP.TEX:  
%  
%-----
```

`\input introapp`

`\bye`

