Brookhaven National Laboratory

M E M O R A N D U M

FROM: Eric Myers

DATE: 13 May 1992

SUBJECT: Basic TEX Rules

TEX is a computerized system for typesetting documents. It is especially well adapted to typesetting complex mathematical formulae, but it is also easy enough to use that it is useful for simpler documents. This memo is a condensed summary of the rules of TEX.

For a more complete description you should read *The TEXbook* by Donald Knuth, or *A Gentle Introduction to TEX* by Michael Doob.

1. The TEX program reads input files ("manuscript files") containing text and "control words" which tell it how to structure the output. The output is a file (called the "DVI" file) which contains instructions for a laser printer or typesetter which describe how to print the pages of the document.

   [TEX is not a WYSIWYG ("What You See Is What You Get") system, but there are previewing programs available on some machines which let you see your output on the screen before it is printed. TEX is more powerful than WYSIWYG systems.]

2. The control words which direct TEX begin with a a backslash followed by a string of letters. Uppercase and lowercase are different! Numbers can't be used in control word names. There are also some other special characters like `$`, `~`, `%` `{` and `}`. Example: the simplest command in TEX is `\relax`, which does nothing at all.

3. A blank line begins a new paragraph and is the same as saying `\par`. (Actually, `\par` or a blank line end the previous paragraph, but it's almost the same thing when you are getting started.)

4. Mathematics to be printed in-line in text must be enclosed between `$` ... `$`, which are called "math quotes". These must balance. Mathematics text is typeset in italics and is spaced differently from regular text. Example: to produce "$E = mc^2$" you would type `$E=mc^2$`.

5. Greek letters are printed by saying `\alpha`, `\beta`, `\gamma`, etc. This only works in math mode (i.e., between math quotes). Example: to produce "$\tau = \zeta_\rho$" you would type `$\tau=\zeta_\rho$`.

6. The summation symbol "$\sum$" is `\sum`, the integral symbol "$\int$" is `\int`, and infinity (∞) is `\infty`. $\hbar$ is `\hbar`. For the other special symbols see the lists in the back of *the TEXbook* (appendix F). These also generally work only in math mode.

7. To make a superscript use `^`, as in `$E=mc^2$` ($E = mc^2$). To make a subscript use the under-bar`_`, as in `$A_\mu$` (A_μ).

8. To make a "displayed" equation instead of an in-line equation enclose the equation in *double* math quotes, `$$` ... `$$`. Example: typing `$$E=mc^2$$` produces:
$$E = mc^2$$
As a matter of style it's good to put the `$$`'s on their own lines in the manuscript file so that the equation is set off from the text just as the equation will be set off from the text in the final output. This will make it easier for you to edit the manuscript file.

9. Grouping: `{` and `}` are used to group things together, as in subscripts and superscripts with more than one "token" (i.e. letters or control words). For example you have to say `$g_{\mu\nu}$` to get $g_{\mu\nu}$. If you say just `$g_\mu \nu$` you will get $g_\mu\nu$. If you change the "environment" (for example, the spacing between lines) within a group the change will go away when the group ends (and things go back to the way they were before the group was started).

10. Fractions are made using `\over` like this: `{` *top stuff* `\over` *bottom stuff* `}` Don't forget the brackets! Example: `$$ m= {F \over a}$$` produces:
$$m = {F \over a}$$

11. It is possible to create your own control words, which are called "macro" instructions. Example: suppose you use the complicated symbol $[Y_l(\theta)]^{ab}$ throughout your paper. You can make `\Yl` a shorthand definition for this by typing `\def\Yl{$[Y_l(\theta)]^{ab}$}` once at the beginning of the paper. Then whenever you use `\Yl` later in your paper it will produce $[Y_l(\theta)]^{ab}$. TEX's ability to define new macro instructions is one of it's most powerful features. It is possible to define an entire collection of specialized macros, which is called a "format." For example, TEXsis is a "format" for physics papers.

If you are completely new to TEX you may find it helpful to read through *A Gentle Introduction to TEX* by Michael Doob, and to do the exercises. TEX is documented completely in *The TEXbook* by Donald Knuth, and first-time users will find chapters 1-10 and 16-19 the most useful. Another source of information about TEX is the somewhat simpler manual by Spivak which accompanies PCTEX (which is an implementation of TEX for the IBM PC).