TEXnical Typesetting

What is TEX?

TEX (pronounced "teck") is a computerized typesetting system developed by Donald Knuth and others at Stanford University. It is used to create high-quality documents, particularly those containing mathematics. The name TEX is an uppercase form of the Greek letters $\tau\epsilon\chi$, the first three letters of a Greek word meaning *art* as well as *technology*. The lowering of the "E" is a reminder that TEX is about typesetting, which can be thought of as the next stage beyond word processing. On devices where such lowering is difficult or impossible you may see TEX written as `TeX`.

Most word processors allow you to create and modify a document interactively — what you see on the screen is usually what your output will look like. TEX does *not* work in this way. Like other typesetting systems (such as SCRIBE and *troff*), TEX is known as a "document compiler". Using your favourite text editor you need to create a file containing the text of your manuscript along with the TEX typesetting commands. TEX gives you the ability to produce printed matter with a quality matching that found in books, depending on the output device. Adelaide University has an IMAGEN laser printer with a resolution of 240 dots per inch. This publication shows both the capabilities of TEX and the output quality of the laser printer.

Fonts

One of the more obvious advantages of TEX is the large range of fonts from which you can choose. A font is a collection of characters each having a similar size and style. Some of the fonts currently available include:

roman *slanted* *italic* **boldface** `typewriter` sans serif SMALL CAPS

Many of these also come in a variety of sizes:

from the very big, to the very small, to the ridiculous.

Apart from a large selection of mathematical symbols, many special characters and accents are available:

© £ § ¶ † ‡ ∘ ◯ ← → △ ♣ à é ç ô ü

TEX does a few subtle things automatically. Certain sequences of characters in your text will be replaced by *ligatures* in the printed output (consider the "`ffi`" in "difficult"), while other pairs of characters need to be *kerned* (e.g., the "o" and "x" in "box" look better if they are moved closer together). The range and quality of fonts available will continue to improve.

Mathematics

A major design goal of TEX was to simplify the task of typesetting mathematics — and to do it properly. Mathematicians will be pleasantly surprised at the ease with which formulae and expressions can be created; from simple in-line equations such as $e^{i\pi} = -1$ and $f_{n+2} = f_{n+1} + f_n$, to more extravagant displays:

$$\sum_{k\geq 1} \sqrt{x_k - \ln k} \quad \neq \quad \int_0^\infty \frac{e^{-x^3} + \sqrt{x}}{(123 - x)^3}\, dx$$

TEX looks after most of the nitty gritty details, such as spacing things correctly and choosing the right sizes for superscripts, parentheses, square root signs etc. (The discoverer of the above relation wishes to remain anonymous.)

Alignment

The preparation of tabular material such as in lists and matrices can be a tedious job for a person armed only with a typewriter and a bottle of correction fluid. With a little help from TEX, computers can make it so much easier:

Oldest players to represent England in a Test Match		
Name	Age	Versus
W.Rhodes	52y 165d	West Indies, 1930
W.G.Grace	50y 320d	Australia, 1899
G.Gunn	50y 303d	West Indies, 1929
J.Southerton*	49y 139d	Australia, 1877

* (This was actually his Test debut.)

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}$$

Other features

Space does not permit examples of all the things TEX can do. Here are some more features you might like to know about:

- Multi-column output can be generated.
- TEX has a very sophisticated paragraph building algorithm and rarely needs to resort to hyphenation. Paragraphs can be indented and shaped in many different ways.
- Automatic insertion of footnotes,† running heads, page numbers etc.
- TEX makes provision for generating a table of contents, a bibliography, even an index. Automatic section numbering and cross referencing are also possible.
- A powerful macro facility is built into TEX. This lets you do some very useful things, like creating an abbreviation for a commonly used phrase, or defining a new command that will have varying effects depending on the parameters it is given. A macro package can enhance TEX by making it much easier to generate a document in a predefined format.

What CAN'T TEX do?

Complex graphics such as diagrams and illustrations pose a major problem — at the moment you have to leave an appropriate amount of blank space and paste them in later. Graphic facilities are the subject of current research.

TEX and VAX/VMS

The TEX source file used to generate this document is available for inspection on any VAX node that has TEX— just type '`scroll tex_inputs:example.tex`'. A few steps are needed to print such a file:

(1) Type '`tex example`' to "compile" the file. (TEX looks for a `.tex` file by default. If it can't find the given file in your current directory it will look in `tex_inputs`.) Two new files will be created in your current directory: `example.dvi` and `example.lis`. The former is a device independent description of the document; the latter is simply a log of the TEX run.
(2) Type '`dvitovdu example`' to preview the document on a terminal screen. This program can be used to detect a variety of formatting problems, saving both time and paper.
(3) Type '`imprint example`' to print the document. (Note that the DVItoVDU and IMPRINT commands accept a `.dvi` file by default).

Detailed help on all these commands is available on-line — try typing '`help tex`' to get started.

† Here is a footnote.