LaTeX and BibTeX Tutorials and Assignment

by Rebecca Underwood, Kevin Karplus, and Brian Levine

1 About this assignment

Students in the class often question why they are being taught LaTeX—after all, it is not a WYSIWYG (What You See is What You Get) page-layout system like PageMaker or QuarkXPress. There are several possible answers to this question:

- LaTeX is an *intentional*, rather then *extensional*, description technique. Rather than saying things like "place this here, make this font bigger, slide this over", you give more general statements of intent: "Start a new section with the following title, make a numbered list of the following items." Because it gives a very different perspective on typesetting, it can break you out of narrow mind set (if all you have is a hammer, pretty soon everything starts looking like a nail).
- LaTeX is distributed for free and is available on many different computers, independent of the operating system.
- LaTeX is the document-preparation system of choice for researchers in the sciences.
- LaTeX has by far the best capabilities for typesetting mathematics, which is difficult or impossible in most other typesetting systems.
- For preparing research papers in any field, LaTeX is an excellent tool—it allows the writer to concentrate on the content of the paper, and leave the formatting, cross-references, and citations to the program and the designer of the class or style file.
- Some of the algorithms (such as the hyphenation and line-breaking techniques) are far superior to the techniques used in other typesetting systems. This is somewhat surprising, since the techniques are not expensive to implement, and have been thoroughly documented in widely available publications.
- The BibTeX system provides one of the best ways of preparing bibliographies and reference lists, automatically generating correct cross-references, and allowing you to build databases of references that can be used in different papers.

Unfortunately, the TeX system was developed by a computer scientist for computer scientists, and so is primarily a programming language. The LaTeX macro package, which is written in the TeX language, hides much of the programming language from the casual user, by providing classes and style files with fairly simple commands to do the tasks envisioned by the style designer. Unfortunately, the programming skills needed for creating and modifying style files are well beyond the scope of this course, and so we will work only with existing style files—even when they do not exactly meet our needs.

This assignment does not call for original, creative work—it is simply a tutorial to get you familiar enough with LaTeX to prepare your research paper.

New this year: This year, I am giving students the option of using a different intentional specification—HTML (Hypertext Markup Language). I have not written a tutorial, but several are available on the World-wide Web. I have collected pointers to them under the CMP80K section of my homepage (`http://www.cse.ucsc.edu/~karplus`). I have also not written explicit directions about what to turn in—we will discuss that in class. The due dates for whatever projects we select will be the same as for the LaTeX tutorials.

This tutorial comes in three parts: first, a simple introduction to using LaTeX on a UNIX computer; second, an introduction to some of the commands you will need to typeset a research paper using LaTeX; and third, an introduction to the BibTeX program that works with LaTeX to produce properly cited reference lists in your papers automatically.

2 A LaTeX tutorial (not to turn in)

To use LaTeX, you will need to do three things: start up the machine, create a text file, then run the LaTeX program on your file. Often LaTeX will find errors in your file and you will need to "debug" (remove errors from) the file. That is, you will need to edit the file, fix any LaTeX command mistakes, then re-run LaTeX. This editing-LaTeXing process must be repeated until LaTeX finds no more errors.

After debugging, you can then preview the output on your screen and print the typeset file. Note that you will need a print card before you can pick up printouts from the laser printers (named `knuth` and `kepler`) in Applied Sciences 105. You can purchase print cards in $1 increments from the vending machine in AS 105.

2.1 Instructions for running LaTeX on UNIX

1. You need to have an account before you can work on an Athena Sparcstation. Any registered UCSC student is eligible for a free Athena account. If you already have one, proceed to the next step. If you don't have one yet, go to the Sparcstation lab in AS 215, sit down in front of a Sparcstation, type `register` in the box next to `username:`, press RETURN, type `athena` in the box next to `password:`, and press RETURN (notice that passwords never appear on the screen even after you've typed them). Follow the instructions to obtain an account. Choose your Athena username carefully; you will have it for your entire UCSC career. Note that it may take minutes or hours before your account is ready for you to use, and plan your work accordingly.

2. To log in, type your username and password in the appropriate blank boxes. The screen will go grey as X Windows initializes. X Windows is an interface that behaves much like that of the Macintosh. With X, you can have several login sessions running at once, one per window.

   In a few seconds, a few windows will pop up. A bar at the top of each window shows the title of the window and three special symbols: (on the left) a minus sign and (on the right) a dot and a square. Clicking on these symbols with the left mouse button lets you modify your windows. Clicking on the minus sign lets you shrink, move, expand, or restore a window to normal size. Clicking on the dot will iconify the window. (You can restore the window by clicking on the icon.) Clicking on the square will expand the window to cover the whole screen.

   To create a new window, click the right mouse button on the desktop (not in an existing window) then select `New window` and shortly a new window will appear. To activate an existing window, click the left mouse button anywhere inside it.

3. Activate one of the windows by clicking on it. Now you're ready to enter UNIX commands.

   Use `vi` (or another text editor, such as `emacs` or `jove`) to create a text file that contains the contents of Figure 2.1. For example, to create a file named `hw1.tex` you can type

   `vi hw1.tex` RETURN

   or, to use `emacs`, type

   `emacs hw1.tex` RETURN

   Note that LaTeX on the UNIX systems will not work correctly unless you make sure that your file's name has the characters `.tex` at the end *and* has no spaces. For example, `my-File.tex` and `yahoo.Foo.tex` are legal filenames, while `first assignment.tex` and `thisThing` are illegal filenames. (If you need help using `vi`, ask your TA.)

   When done entering the data, save your file and exit to the UNIX prompt.

```
\documentclass[12pt]{article}
\title{No time for art}
\author{Jill F.~Charboneau}
\date{January 1989}

\begin{document}
\maketitle

The VCR may become an increasingly important funding source for the
arts by art lovers who want to see live performances and art shows,
but who do not have enough free time.
These are the findings of Lou Harris and Associates Inc.\ in a study
called `America and the Arts V'.

The study found that between 1984 and 1987, the amount of people going
to arts events fell 12\%.
More than 75\% of all VCR owners, which include 42\% of all adults,
said it was probable or certain they would rent or purchase
videocassettes of hit musical comedy or theater shows.
Videocassettes of new plays on Broadway or in London would be of
interest to nearly two-thirds of the VCR owners.
There is also interest in videocassettes of recent concerts, art
shows, operas, and symphony concerts.

\end{document}
```

Figure 2.1: Text taken from "No time for art" by Jill F. Charboneau in *American Demographics*, volume 11, number 1, January 1989, p. 16. Use it for the LaTeX tutorial.

4. To run LaTeX, exit `vi` and type

   `latex` *name*`.tex` RETURN

   For example, if you created a file named `hw1.tex` then you would type

   `latex hw1.tex` RETURN

   to run LaTeX on your file.

   As LaTeX processes your file, it will display messages on your screen. Each bracketed number (for example, `[1]`) corresponds to a pageful of processed material. If LaTeX finds any errors, it will display an error message and stop processing your file. If this happens, you must type `end` RETURN and go back to Step 3 to correct any typos or LaTeX command mistakes.

5. When LaTeX runs through without errors, it will create three new files having `.aux`, `.log`, and `.dvi` suffixes, and UNIX will give you its prompt. The file with the `.dvi` suffix is your output file; it contains the information needed to preview or print the output. You can now run the page-previewer, `xdvi`, on the new `.dvi` file. Type

   `xdvi` *name*`.dvi` RETURN

   If you've been working with a file named `hw1.tex` then to preview the LaTeXed output you would type

   `xdvi hw1.dvi` RETURN

   (The `xdvi` program runs only Sparcstations, not on Macintoshes.) Scan the preview carefully. If you discover typos while proofreading the preview screen, then note where the problems are and return to Step 3 to correct your mistakes. (For experienced UNIX users, you can start the `xdvi` program in the background with `xdvi hw1.dvi &` RETURN, edit your `.tex` file, re-run LaTeX, and then see the new result just by clicking on the `xdvi` window.)

6. When the preview looks exactly the way you want, you can print the output file. Type
   `dvips` *name*`.dvi -h -Pknuth` RETURN
   and promptly pick up your printout from the `knuth`laser printer in AS 105. You can also use the `kepler` printer in AS 105, or any other laser printer available to students. (Warning: I have not checked the names of the printers in Applied Sciences this year—you might want to check before you issue the command.)

2.2 What to notice about what you've typed

- The document class determines the default margin and font sizes. With special commands, you can change these within a document; see p. 176 of the LaTeX manual or ask your TA for information. (Different document classes have different default margin and font size settings.)
- All LaTeX files must have a `\documentclass` command, a `\begin{document}`, and an `\end{document}`.
- LaTeX fills lines to the margins. It doesn't matter how short the lines are or how many spaces you place between words in your input file. LaTeX uses a sophisticated line-breaking algorithm to place the right number of words on each line. We recommend typing a RETURN after each sentence, to make subsequent editing easier.
- Certain characters, including %, $, #, and &, have special meaning to LaTeX. If you want to produce such characters in your output file, you need to precede them with a backslash (\) in the input.
- Open and close quotation marks must be specified with different characters. To obtain an open quotation mark, you need to type two ‘ marks. (The single quote marks used around `America and the Arts` in the example are not standard American typesetting. Double quotes or, better, italics would be more appropriate.)
- Some periods are not ends of sentences. To keep LaTeX from spacing them as if they were ends of sentences, you need to do one of two things: put a backslash before the space (as was done after `Inc.`) or change the space to a tilde (~), which is used in LaTeX to specify a space that cannot be used as a line break.

3 More advanced use of LaTeX

The first part of this assignment showed how to typeset a document using the `article` document class, having a title page, but having no sections. Many other document styles exist: two examples by Prof. Karplus are `ucsc-report`, used to typeset UCSC theses, conference papers, and technical reports, and `handout`, used to typeset handouts like this one.

In this part of the assignment, you'll learn how to use the `article` document class, how to typeset tables and figures, how to make section headers, how to use special fonts, and how to make a table of contents. More details on how to use the `ucsc-report` style are in the technical report *Using the* `ucsc-report` LaTeX *style file for UCSC technical reports* [UCSC-CRL-8090-25]. However, this style file has not been updated for LaTeX2e on CATS yet (though it worked ok in Switzerland in summer 1995).

	Default	Default margins (in inches)			
Document style	font size	Right	Left	Top	Bottom
`article`	10pt	1-3/4	1-7/8	2	1-7/8
`ucsc-report` (odd pages)	11pt	1	1-1/2	1-1/8	7/8
(even pages)		1-1/2	1	1-1/8	7/8

Table 3.1: Default font and margin sizes for the `article` and `ucsc-report` document styles. Note that `ucsc-report` margins depend on whether the page is odd- or even-numbered. The top margin is larger than the bottom, because it includes a header line. The space above the header in ucsc-report is only 7/8".

3.1 An example

This handout shows both the raw input `.tex` file and the typeset output `.dvi` file. The input file is shown in Section 3.3, pages 8–12 in `typewriter` font. You don't need to read these pages carefully—instead, refer to them to find LaTeX commands for the homework.

The following sections explain the LaTeX commands found in the example input file.

3.2 Explanation of commands

This section explains each of the different commands in order of their appearance in the example. Note: the example document has no bibliography, though normally a formal report such as this one would.

`\documentclass[10pt]{article}` This selects the `article` style with the option `[10pt]`. The option tells LaTeX to use the 10-point Computer Modern font family.

Recall that the document style determines the default margin and font sizes. Some default settings are shown in Table 3.1. The gutter margins in `ucsc-report` are large to compensate for stapled bindings, but the reports distributed by UCSC use tape binding, and so the gutter and outer margins should really be modified. With a 6" line width, the extra 2.5" should probably be split into .833" and 1.667" for the gutter and outer margins.

`\def \extradown { ... }` **and** `\def \extraup{ ... }` The command `\def` is a TeX command for defining macros. LaTeX will substitute what's in the curly braces after the command name wherever the command name appears in the `.tex` file. The commands defined here are named `\extraup` and `\extradown`. These particular commands make a line of text take up more space vertically by adding an invisible (zero-width) vertical line. They are used in `tabular` environments to keep the table's rules and data appropriately spaced. The `\extraup` command adds space above the line, and so is appropriate for the first line of a table after a horizontal rule in the table. The `\extradown` command adds space below the line, and so is appropriate for the last line of a table before a horizontal rule.

This command starts the text of the document. Any commands that put out text must come after this command.

`\title{{\protect \sc a selection from }\\ ... }` This title command has a couple new features: the `{\protect\sc ... }` forces the text inside the curly braces to print in the small-caps font. Outside of special environments (like `\title` or `\caption`), font changes can be done without the `\protect`. See Table 3.2 for a list of fonts available in LaTeX and the commands to access them.

LaTeX Command	Font
`\rm`	Roman (the default)
`\bf`	**Bold**
`\it`	*Italic*
`\sl`	*Slanted* (oblique)
`\sf`	Sans serif
`\sc`	SMALL CAPS
`\tt`	`Typewriter` (like Courier)

Table 3.2: To use these fonts, type an open curly brace, the font command, the text you want typeset in that font, then a final closing brace. Example: `{\tt This is in typewriter.}`

`\author{ ... }` and `\date{ ... }` These are the usual fields for specifying author and date of the document. If there are multiple authors, separate their names with `\and`, so that the style file can format them appropriately as a horizontal or vertical list.

`\maketitle` This produces a separate title page using all the fields defined up to this point. The `\maketitle` command does actual formatting, and so must come after the `\begin{document}` command. The title, date, and author will all print on this page. By default, the `article` class puts the title and author at the top of the first page of text, with the body immediately following. To create a a title page and avoid the default, the next command was used.

`\vspace {2in}` This command specifies a vertical white space between lines. It was used here to place the abstract farther down the page.

`\begin {abstract }` ... `\end{abstract}` The abstract for the article goes within these delimiters. If your paper has no abstract, leave out this command. An abstract is normally about 50–200 words giving a brief summary of the main results of the paper. Almost all conferences and journals require abstracts, so that their readers can figure out quickly whether or not to read a particular article. They are also very useful in databases like INSPEC, COMP, MEDLINE, and MAGS on MELVYL.

`\bf keywords` ... Here are listed the article's main topics—added to help people using computerized indexes find your paper. It is not usually necessary to duplicate words that appear in the title. Although this is not a command (only a boldface command was used), in many styles, such as the `ucsc-report`, it is included as part of the maketitle. If you don't want to list keywords, leave out this line.

`\clearpage` leaves the current page and start typesetting on a new one. A `\pagebreak` will do this too, but it has the (usually) undesirable side effect of making LaTeX spread out the text on the previous page to fill extra space to the bottom margin.

`\pagestyle{headings}` This defines the format of the header and footer of the document. `headings` indicates that the page number, section name, and section number will be put in the header. Two other options are `plain`, which is just a page number in the header, and `empty`, which is no header or footer at all. See page 89 of the LaTeX manual for more information, including how to create your own pagestyle.

`\tableofcontents` This command produces a table of contents. To get this to work right, you must run Latex twice in a row—the first time to find out where the sections start, the second time to fill in the page numbers in the table of contents. The % character makes the rest of the line a comment, and so running the example unmodified will not generate a table of contents.

`\section{ ... }` To make a section heading, place the text of the heading between this command's curly braces. Subsection and subsubsection headings are made the same way, with the appropriate command name. Subsubsubsections and so forth are not supported.

`When {\it justifying\/}` ... This italicizes the word "justifying" and uses the italic correction `\/` afterwards.

`Figure \ref {leadingExample}` ... `}` The `\ref` command refers to a table later in the document. The table is labeled with "leadingExample", as you will see when we get to it. The ~ character is used to put an unbreakable space between the word *Figure* and the number of the figure.

`\begin{figure}` starts a figure environment. LaTeX will place figures and tables where it sees fit, and not necessarily in the exact location where you typed it in. Objects that are not fixed with respect to the text (like figures and tables) are called *floats*, while similar objects that cannot be moved with respect to the text are called *displays*. Usually floats (figures and tables) are placed at the top of the current or next page. Because the float can move, it must be referred to by name in the main body text.

Figure environments are similar to tabular, verbatim, and document environments. Every `\begin` must be matched by an `\end`.

`\vspace{3.5in}` places 3.5 inches of vertical space in the typeset document. Later an illustration may be drawn, taped, or pasted in here. There are also ways to import drawings made with `xfig` and encapsulated postscript files.

`\hrule` creates a horizontal rule from the left to the right margin.

`\caption {` ... `}` Part of the `figure` environment, the `\caption` command allows you to place a figure caption within the curly braces. All figures should have captions, and captions are properly one- or two-sentence explanations of the figure, not just titles. Most people will flip through an article looking at the figures and reading their captions before reading the article, and so it helps enormously if the captions explain the main points, rather than just giving titles to the figures.

`\label{leadingExample}` labels the current environment with the name typed between the curly braces (here, "leadingExample"). The `\label` command records the most recently assigned number (from a `\caption`, `\section`, `\subsection`, or similar command), and so the `\label` command must come after the `\caption` command. The `\ref` command used earlier expands to the number recorded in the `\label` command.

`\end {figure}` As mentioned previously, environments require both `begin` and `end`.

`\subsection{Justification of lines}` This follows the same format as regular sections.

`\begin{table}[t]` The start of a table environment is marked with this command. The optional argument `[t]` means that the table will appear typeset at the top of a page. You can specify four places on the page where you want the table to be typeset:

`[h]`	at the current position
`[t]`	at the top of a text page (LaTeX decides which page)
`[b]`	at the bottom of a text page
`[p]`	on a separate page containing no text, only figures and tables

`\begin{center}` The `center` environment will center the table horizontally.

`\begin{tabular}{|r|r|r|r|}` starts a tabular environment having four columns and five vertical rules. Each column is indicated by an `r`, and each vertical rule by a \|. The `r` means that the data within the column should be aligned right. A tabular environment can exist outside of a table environment (to make a displayed table, for example). The table environment produces a float, and has a `\caption` command to label the float, while the tabular environment produces an unlabeled table. It is quite standard to have the nested environments `table`, `center`, and `tabular`.

`\hline` creates a horizontal rule across the width of the table.

`\extraup` Remember those new commands defined previously with `\def`? Here is where they get used. The `\extraup` will put a little extra vertical space between `\hline` and following text. The `height` parameter specifies the minimum separation between the baseline of the text and the `\hline`—in this case, 2.5 times the ex-height.

`{\bf Line}& {\bf Set }& {\bf 1-Point}& {\bf 2-Point }\\` translates to a row in the table. Ampersands (`&`) separate column entries. The `\\` at the end indicates the end of the row of entries. Each column entry must be boldfaced separately.

`\extradown` puts some vertical space between the current table row and the next `\hline`. The `depth` parameter in the definition of `\extradown` gives the minimum separation between the baseline of the text and the `\hline`.

`\end{tabular}` ends the tabular environment.

`\caption{\label{myTable}` ... `}` The format for a table caption is the same as for figures. The `\label` command here labels the current table as `myTable`. A label command can be put inside curly braces of a caption or after the end of the caption, but not before the `\caption` command, since it needs to record the figure or table number generated by the `\caption` command.

`\copyright` generates a copyright symbol ('c' in a circle).

`\end{document}` ends the document.

3.3 The example input file

Text for this input file was paraphrased from Richard Rubinstein's *Digital Typography*, Chapter 5, pages 153–194, Addison-Wesley Publishing Company, Menlo Park, 1988. We are asking you to retype it, rather than giving it to you electronically, for several reasons:

- To give you some practice using the `vi` or `emacs` editors (your choice).
- So that you look very carefully at the commands, and don't just glance at them. As in all computer languages, punctuation and small details matter very much to LaTeX.
- So that you can experience debugging using the error messages LaTeX gives you when you have a typo, with a correct example to check your input against. (The alternative is to practice debugging LaTeX for the first time when you're not sure what the correct input looks like.)

```
\documentclass[10pt]{article}

% defines special commands to put extra space around table entries
\def\extradown{\vrule width 0pt depth 1.2ex}
\def\extraup{\vrule width 0pt height 2.5ex}

\begin{document}
\title{{\protect\sc a selection from} \\ \bf Space and putting things in it}
\author{by Richard Rubinstein}
\date{January 1988}

\maketitle

\vspace{2in}
```

```
\begin{abstract}
Typographers attend to space, the part of the page
{\it not\/} occupied by images.
Text layout facilitates visual parsing of pages by readers, helping
them find information and avoid confusion.
Typewriter habits account for some defects in typeset material.
Extra space after punctuation, wide lines with little interline space,
and extra space between paragraphs are all contrary to good
typesetting.
\end{abstract}

\vspace{1in}

{\bf Keywords:} typography, layout, graphic design, use of space

\clearpage
\pagestyle{headings}

\setcounter{figure}{0}
\setcounter{table}{0}

% Remove the % from the next line to get a table of contents
% \tableofcontents\clearpage


\begin{figure}
\vspace{2.5in}
\hrule
\caption{The mood of text varies with the leading.
The left sample (a) might be appropriate for a scholarly monograph,
economical of paper and tightly reasoned.
A sumptuous art book might be leaded loosely as in (b).}
\label{leadingExample}
\end{figure}

Like other artists, typographers frequently focus their attention on
the space occupied by the objects that they create, what is normally
background; this is a useful exercise in graphic design.
This report considers the space occupied by text and pictures.
The objective is to understand the relationship of the images to each
other and to the pages on which they appear.


\section{Views of space}

What is a good way to allocate space within a line when typesetting a
line of text?
When {\it justifying\/} lines so that the right and left edges of a
block of text are straight, this simple question actually has a simple
answer: The ``extra'' space can be distributed evenly between the
words.
Extra space between letters within the words can degrade the
legibility of print.
```

```
The term {\it color} is also used by typographers to describe the
overall tone of gray that text presents on a page.
Interline spacing affects the look of a block of text, its texture and
typographic color, and is an important component of a typographer's
design for a book or other printed work.
Without extra leading, blocks of text may be very economical of space.
Depending on the face design, such print may look too dense on the
page, and may be harder to read than type spaced more loosely.
Conversely, widely spaced type may look sparse, and may be hard to
read because the eye is unaccustomed to finding the next line at such
a vertical distance.
Figure~\ref{leadingExample} shows how the same text can give different
overall impressions depending on the leading used.


\subsection{Justification of lines}

Justification of lines of print is an essential typesetting function.
It is simply placing lines of text in some specified relation to the
margins.
Given a sequence of symbols that are to appear on a particular line,
the system must decide where to place them.
Since their order is fixed, and spacing between letters within words
is generally not modifiable, justification comes down to deciding how
much space should be placed between words and at the left and right.

\subsection{Some layout guidelines}

Here are some general principles and guidelines that apply to document
layout.
They come primarily from typographic practice, but are consistent with
psychological evidence and the nature of automatic typesetting.

\subsubsection*{Don't impose typewriter habits}

Typewriter habits, intended for a monospaced environment, should not
be followed when typesetting with proportionally spaced typefaces.

\subsubsection*{Typesetting ``typed'' material}

When a ``typed'' look is required by a document, consider using a
proportionally spaced typeface reminiscent of typewriter styles, such
as ITC Typewriter, or simply a typeface that contrasts with the other
faces used.

\subsubsection*{Set narrow lines ragged right}

When indented itemized lines are set, the resulting lines may be very
narrow.
If such lines are justified, large interword spaces are inevitable.
The style of many typeset documents is, therefore, to set these narrow
lines ragged right.
```

```
This is done even when the surrounding text is justified.

\subsubsection*{Margins}

Ample margins may facilitate reading by helping the reader avoid
distractions from objects near the text, and by preventing a crowded
look on the page.
Bear in mind that books, folders, reports, and other documents that
open to reveal two pages have a double margin at the center.
Frequently, somewhat less space is used in this ``gutter.''

\subsubsection*{Use a grid}

The value of gridding for electronic publishing is that it can provide
a useful user-interface specification for layout.
Only a few current desktop-publishing systems offer grids (PageMaker
is one).

\subsubsection*{Provide redundant parsing clues}

Multiple differences in page design make it easier for people to
understand information.
For example, footnotes can be smaller, at the bottom of the page, and
more narrowly spaced.


\section{Leading, line width, and type}

Formatting programs frequently take as givens the type size and line
width, which depend greatly on the kind of document the user wants to
produce, the size of the paper, and the sizes of type available on the
printer.
If good values for leading could be computed from a formula or
algorithm, then formatting programs could produce results that would
require little or no adjustment by the user.

~\ldots~In six-thousand-subject study, psychologist Miles Tinker
determined optimal line widths and leading for each type size he
used (Table~\ref{myTable}).
His data are somewhat hard to use prescriptively, because he quotes
the amount of leading {\it added\/} to the interline white space.
The typeface used was Excelsior~\ldots

\begin{table}[t]
\begin{center}
\begin{tabular}{|r|r|r|r|} \hline
\extraup
{\bf Line}  & {\bf Set}   & {\bf 1-Point} & {\bf 2-Point} \\
{\bf Width} & {\bf Solid} & {\bf Leading} & {\bf Leading} \extradown \\
\hline
\extraup 9 & -9.3 & -6.0 & -5.3  \\
 14 & -4.5 & -0.6 & -0.3  \\
 19 & -5.0 & -5.1 & {\bf 0.0}\extradown\\
```

```
\hline
\end{tabular}
\end{center}
\caption{\label{myTable} Optimal combinations of line widths (in picas) and
leading for 10-point type.
Figures are the percent reduction in speed relative to
a chosen setting (shown in {\bf bold face}).
``Set solid'' indicates no added leading.
(Miles A.~Tinker, {\it Legibility of Print}.
\copyright 1963.
Reprinted
courtesy of Iowa State University Press.) }
\end{table}
\end{document}
```

4 The LaTeX exercise (due Wednesday 9 February)

Create a dummy paper containing at least two sections and two floats: a centered table and a figure. Use the `article` document class and set the article in 10-point font.

The figure should be a crisply rendered (easily identifiable) line drawing or black-and-white photograph of your choosing. It should be appropriate in subject matter and in intensity (not too dark but not too light).

The best easy way to produce such a figure with LaTeX is to leave enough space for it in your LaTeX document using `\vspace` inside the `figure` environment. then tape or paste the illustration onto the laserprinted output and xerox that to produce a new camera-ready original. Try to make the tape or paste inconspicuous, and be sure to align the illustration. Alternatively, if you trust your artistic skill, draw an illustration with a black pen directly on the laserprinted output.

Although the desktop-publishing books would have you think otherwise, this piecemeal process is similar to what happens in the magazine industry: artwork is photostatted (photographed and usually reduced or, less commonly, expanded in size), then the photostats are pasted up on a galley next to typeset text. This helps preserve the resolution of the art.

You will not be given extra credit for using `xfig` or programs that import PostScript figures into LaTeX files, although the TA will admire your tenacity.

If you can't think of any tables that you really want to format, try reproducing Table 5.1 on page 179 of Rubinstein's *Digital Typography*. If that seems too easy, try seeing how close you can come to the horizon table in Figure 7.6 on page 263, which is extremely difficult to typeset well.

Table 4.3 is one attempt at duplicating the style of the table. Getting the column of differences to be vertically offset between the lines was rather tricky in LaTeX, and required using the optional arguments of the `\raisebox` command (see LaTeX manual). This table also uses the `\hline`, `\multicolumn`, invisible `\vrule`, `\noalign`, and @-sign formatting commands for tables. See pages 182–185 of Lamport's *LaTeX: A Document Preparation System* for more information on these commands. (For `\noalign`, you'll have to look in Knuth's *The TeXbook*, since Lamport doesn't believe you need to know about it. If you are willing to have the horizontal rules go all the way to the vertical rules, then you can get by with using `\cline` and not need `\noalign`.)

5 BibTeX tutorial (due Monday 14 November)

Reference lists (bibliographies) and citations are what distinguish scholarly writing from almost all other forms. The worst sin in academia is to use someone else's work without giving proper

Table des inclinaisons de l'horizon visual avec l'horizon vrai.												
Elev. au-dess. de la mer.		Inclinaison de l'horizon.			Elev. au-dess. de la mer.	Inclinaison de l'horizon.			Elev. au-dess. de la mer.	Inclinaison de l'horizon.		
pi.	po.	m.	s.	d.	pieds.	m.	s.	d.	pieds.	m.	s.	d.
0.	6	0.	44	17	28	5.	26	11	94	9.	58	12
1.	0	1.	1	14	30	5.	37	11	98	10.	10	12
1.	6	1.	15	12	32	5.	48	10	102	10.	22	11
2.	0	1.	27	11	34	5.	58	11	106	10.	33	12
2.	6	1.	38	9	36	6.	9	10	110	10.	45	12
3.	0	1.	47	16	38	6.	19	10	114	10.	57	11
4.	0	2.	3	15	40	6.	29	10	118	11.	8	12
5.	0	2.	18	13	42	6.	39	9	122	11.	20	11
6.	0	2.	31	12	44	6.	48	9	126	11.	31	12
7.	0	2.	43		46	6.	57		130	11.	43	

Table 4.3: Excerpt from a table of horizons from François Callet's *Tables Portatives de Logarithmes* published in 1795, re-typeset using LaTeX.

credit. References (or resources) are books, articles, magazines, proceedings, manuals, Ph.D. theses and so forth; citations are within-the-text references to those resources (for example, [Smith88] and [17] are citations). For most disciplines, the best reference on correct citation style is Mary-Claire van Leunen's book *A Handbook for Scholars*.

There are two ways to make reference lists in LaTeX: by typesetting the reference list yourself, or by creating an additional references file and letting BibTeX typeset a bibliography from it. If you are writing many articles that cite many of the same resources, then BibTeX can conveniently make those reference lists for you; all you need to do is cite the appropriate resources in your `.tex` article files. Even for single-use citations, BibTeX is handy in that it manages the formatting of the reference list—you need only provide the information in the database, not worry about how to format it.

BibTeX is a program separate from LaTeX, and does two things. First, it takes bibliographic information that you put in a special database file (a `.bib` file) and creates a nice-looking reference list that prints at the end of your regular document. Second, BibTeX keeps track of and produces appropriate citations to the reference list.

To use BibTeX, you will need to do two things: add some commands to a standard `.tex` file (these allow BibTeX to make the citations), and create or edit a `.bib` file (this allows BibTeX to make the reference list).

The additions consist of

- A `\bibliographystyle{}` line anywhere in your `.tex` file. What you type inside the curly braces (the argument) of this command determines what style of abbreviation and punctuation BibTeX will use for your citations and bibliography entries.
- As many `\cite{}` commands in the text of your regular `.tex` file as appropriate. BibTeX and LaTeX will turn these into actual citations (for example, [23] or [Smith91]) when you typeset

```
\documentstyle{article}
\title{Connectionist Music Composition}
\author{Michael C.~Mozer}
\date{15 April 1991}

\begin{document}
\maketitle

Algorithmic music composition involves using heuristics in the
generation of melodies.
One simple but interesting technique is to select notes sequentially
according to a transition table that specifies the probability
of the next note as a function of the previous context.

I describe an extension of this transition table approach using a
recurrent connectionist network called {\sc Concert}.
{\sc Concert} is trained on a set of melodies written in a certain
style and then is able to compose new melodies in the same style.
A central ingredient of {\sc Concert} is the incorporation of a
psychologically grounded representation of pitch.~\ldots~

\end{document}
```

Figure 5.2: For the tutorial, use this text from M. Mozer's article, "Connectionist music composition based on melodic, stylistic, and psychophysical constraints."

your regular file. Normally BibTeX will include in your reference list only the resources that you actually cite in your paper.

- As many `\nocite{}` commands in the text of your regular `.tex` file as needed. This is how you force BibTeX to include in the reference list resources that you don't actually *cite*. It is generally considered extremely poor style to include in the reference list a resource that is not cited in the text—you should rarely, if ever, need the `\nocite` command.
- A `\bibliography{}` line in your `.tex` file where you want the reference list to appear. The argument of this command contains the name of the `.bib` file that holds the bibliographic information. You can have several `.bib` files; just list them all inside the curly braces after `\bibliography`, and separate the filenames with commas.

The *database* (`.bib`) file holds bibliographic information such as titles, authors, dates, page numbers, names of publishers, and so forth. The format for entries is well described in Appendix B of the LaTeX manual. You can use `vi` (or any other text editor) on a UNIX system to create the database file. We'll see how to do this in detail in the following sections.

5.1 Create a standard `.tex` file

Create a simple LaTeX file called `connect.tex` for this tutorial. Type in Figure 5.2 using `vi`.

5.2 Add the bibliography commands

What you put in curly braces after `\bibliographystyle` affects the style of both citations and bibliography entries. Each bibliography entry is a summary of publication information about some

Citation style	Reference-entry style	Style to use
number in square brackets	entries ordered alphabetically	`plain`
	entries listed in order cited	`unsrt`
	entries abbreviated and listed in alphabetical order	`abbrv`
abbreviated names & dates	entries ordered alphabetically	`alpha`

Table 5.4: Standard citation styles available for BibTeX.

resource. The following example is a BibTeX `alpha`-style entry:

> [Kar89] Kevin Karplus. Using if-then-else DAGs for multi-level logic minimization. In Charles L. Seitz, editor, *Advanced Research in VLSI: Proceedings of the Decennial Caltech Conference on VLSI*, pages 101–118, Pasadena, CA, 20–22 March 1989.

Table 5.4 lists the four most common of BibTeX's styles. You can find other styles by looking for `.bst` files (in `/usr/lib/tex/macros` on CATS machines or `/usr/local/TeX/lib/inputs` on Baskin Center machines). Other BibTeX styles include `aaai-named`, `acm`, `apalike`, `ieeetr`, and `siam`.

The four BibTeX styles in Table 5.4 allow two formats for typeset citations: a number in square brackets like [93] or an abbreviated name and year like [Luu91]. Other style files provide other choices (for example, `aaai-named` uses full author name and year, rather than abbreviated ones).

In the reference list, entries can be listed in alphabetical order by the author's last name or in the order of the corresponding citations' appearances in the text. The reference list can give full names of the authors and journals, or can do some abbreviation (using the authors' initials for example).

- Returning to our tutorial, assume we want to use the `alpha` style. At the end of the document, after `~\ldots~`, type `\bibliographystyle{alpha}`.
- After that line, type `\bibliography{music1}`. This command uses the name of an as-yet-nonexistent database file (`music1.bib`) as its argument. Later, BibTeX will insert the typeset bibliography in the printed document exactly where you typed this line. Note that you need not type the `.bib` suffix.

Your document should look like this:

```
\documentstyle{article}
\title{Connectionist Music Composition}
... representation of pitch.~\ldots~
\bibliographystyle{alpha}
\bibliography{music1}
\end{document}
```

5.3 Create the `.bib` file

The *database* file holds bibliographic information (titles, authors, names of publishers, and so forth). The name of any bibliographic database file must have a `.bib` suffix. Inside the `.bib` file, you need to put entries for each article or book you will be citing. See pages 140–147 in the LaTeX manual for full explanations of the different entry formats (`inbook`, `misc` and so forth).

Note: You can save yourself trouble if you write a complete `.bib` entry for each article or book as you are doing research for the paper. That way, if you need to cite the resource later, you don't have to find it again to check the spelling of the author's name or the page number in the journal. Try to make your `.bib` file entries as complete as possible; it is a common problem for students (and some professional scholars) to provide useless, incomplete references. It is far better to be redundant than to be incomplete.

- Start editing your database file by typing `vi music1.bib`. The name of the file must match what you put in the `\bibliography` command, and must have a `.bst` extension.
- Type the data from Figure 5.3 into your currently blank `music1.bib` file.

5.4 Add `\cite` commands to the `.tex` file

To get good typesetting of citations, you should precede `\cite` commands with an unbreakable space (indicated by a `~`, a tilde, in LaTeX). The citations should occur outside the closing quotation mark of a quotation, or at the end of the phrase, sentence, or paragraph containing paraphrased material.

- In your `connect.tex` file, insert four `\cite{}` commands as shown:
- ... `heuristics in the generation of melodies.~\cite{dodge}`
- ... `as a function of the previous context.~\cite{fraisse,dodge}`
- ... `in the same style.~\cite{fraisse}`
- ... `psychologically grounded representation~\cite{auntjem}`

Note that since we didn't cite Smolensky, his article will not appear in the printed reference list for the final document. Also note how we cited two authors simultaneously. Now let's see how the output looks.

5.5 Run BibTeX

To create a final document that includes a typeset bibliography, do the following steps:

- Run LaTeX, by typing `latex connect` to the UNIX prompt. LaTeX will find the `\cite` commands and put requests for citations into `connect.aux`. It will complain that your citations are undefined—just ignore that message for now.
- Run BibTeX. It will look up the citations in the database and collect information for the reference list. On UNIX systems, type the command `bibtex connect` to do this. You should see something like this:

  ```
  % bibtex connect
  This is BibTeX, C Version 0.99c
  The top-level auxiliary file: connect.aux
  The style file: alpha.bst
  Database file #1: music1.bib
  ```

- Run LaTeX again so it can typeset the reference list and add the labels for the entries to `connect.aux`. You should see something like this:

```
@book(dodge,
  author = "Dodge, Carla and Jerse, Timothy A.",
  title = "Computer Music: Synthesis, Composition, and Performance",
  publisher = "Shirmer Books",
  year = "1985", address = "New York", edition = "first"
  )
@article(smolensky,
  author = "Smolensky, Pat",
  title="On the Proper Treatment of Connectionism",
  journal = "Behavioral and Brain Sciences",
  year="1988", month="December",
  volume="27", number="11", pages="1--74"
  )
@inbook(fraisse,
  author = "Fraisse, Petra",
  title = "The Psychology of Music",
  chapter = "Rhythm and tempo", pages = "149--180",
  publisher = "Academic Press",
  year = "1982", address = "New York"
  )
@misc(auntjem,
  author = "Cricket, Jemima",
  howpublished = "Electronic mail correspondence", year="1989",
  )
```

Figure 5.3: Data for `music1.bib`, the bibliography file for the tutorial sample document.

```
...
LaTeX Warning: Citation ‘fraisse’ on page 1 undefined.
LaTeX Warning: Citation ‘dodge’ on page 1 undefined.
LaTeX Warning: Citation ‘auntjem’ on page 1 undefined.
[1] (connect.aux)
LaTeX Warning: Label(s) may have changed. Rerun to get
cross-references right.
...
```

- Run LaTeX one last time, so LaTeX can insert and typeset the correct labels where you put the `\cite` commands. You should see no error messages:

```
... (connect.aux) (connect.bbl) [1] (connect.aux)
Output written on connect.dvi (1 page, 1930 bytes).
Transcript written on connect.log.
```

5.6 Take a look

- Run your previewer as usual: `xdvi connect`.

5.7 Laserprint the tutorial (optional)

Once you've proofread your work and checked that the citations came out all right, you're ready to print.

- From the Sparcstations, type `dvips connect.dvi -h -Pknuth` as usual.

5.8 Experiment

- Try editing the `connect.tex` file so the bibliography style is no longer `alpha` but `plain`, `abbrv`, `unsrt`, or some other style. Save your changes and repeat the process from Section 5.5 on. Compare the outputs of documents using different bibliography styles.
- Insert a `\nocite{smolensky}` command somewhere in `connect.tex`. Save changes and repeat the process from Section 5.5 on. Observe that the Smolensky resource will appear on the reference list even though you didn't cite it in the paper.

5.9 Start preparing your database for your research paper

The main point of this tutorial is to get you familiar enough with LaTeX and BibTeX to prepare a research paper using them. Enter at least five of the references that you might use in your final paper in a `.bib` file, and write a very short `.tex` file that cites each of them—if you want, you can place the citations in the dummy paper you used in Section 4.

6 What to turn in

- A one-page cover memo explaining any problems you encountered, ideas for possibly fixing them, and other reflections on LaTeX and BibTeX. As usual, list names and contributions of people who proofread, pointed out errors and unclear passages, commented on formatting, or made suggestions relating to your work.
- The typeset printout of your dummy papers, demonstrating your command of sections, tables, and citations, as described in Sections 4 and 5.9. As mentioned in the latter section, you may turn in a single paper if it has the following features (minimum):
  1. two sections (using `\section` commands),
  2. one figure (using the `figure` environment),
  3. one table (using the `tabular` environment),
  4. 10-pt body text set using the `article` document style, and
  5. five citations (using the `\cite` command) for references listed in your accompanying `.bib` bibliography file.
- A printout of your paper's `.bib` file, containing at least five references, as created in Section 5.9. Don't typeset it, just print it: type `enscript -2rG -h -Pknuth paper.bib` to print a `.bib` file called `paper.bib` to the `knuth` laser printer. Since the document is not typeset, printing to a cheaper printer using `lpr` instead of `enscript` is also acceptable. Enscript with the specified options gets two pages of text onto each sheet of paper, and so there may be no savings using lpr.
- (Optional) The laser-printed tutorial examples, from Sections 2 and 3.3. This is not an extra-credit option, but a fallback option for those who are having trouble, and want to show that they managed to do at least part of the assignment.
- (Optional) A sheet voicing any comments or suggestions you have relating to this tutorial. You can submit these anonymously if you like.