PHYS-4007/5007: Computational Physics

Tutorial #3

Learning the LaTeX Mark-Up Language

1 Introduction

LaTeX is a high-quality typesetting system; it includes features designed for the production of technical and scientific documentation. LaTeX is the de facto standard for the communication and publication of scientific documents. LaTeX is available as free software for vitually all operating systems.

LaTeX, which is pronounced '*Lah-tech*' or '*Lay-tech*,' is a document preparation system for high-quality typesetting. It is most often used for medium-to-large technical or scientific documents but it can be used for almost any form of publishing. Section III of the course notes goes into detail on the LaTeX Mark-Up Language.

LaTeX Features:

- Typesetting journal articles, technical reports, books, and slide presentations.
- Control over large documents containing sectioning, cross-references, tables and figures.
- Typesetting of complex mathematical formulas.
- Advanced typesetting of mathematics with AMS-LaTeX.
- Automatic generation of bibliographies and indexes.
- Multi-lingual typesetting.
- Inclusion of artwork, and process or spot colour.
- Using PostScript or Metafont fonts.

2 Getting Started with LaTeX

Log into your Linux account, open a terminal window, and change directory to your 'tex' subdirectory. Now open the web browser and go to the Computer Programming Tutorials Web

Page on the course web page:

https://faculty.etsu.edu/lutter/courses/phys4007/

Scroll down to the 'Programs and Supplemental Files for the Tutorials' table and click on the 'Tutorial03_LaTeX.tex' link in the data table on the Tutorial 3 row. Following this, open a new file in your tex subdirectory using the emacs editor with

```
> emacs Tutorial03_LaTeX.tex &
```

where the '>' symbol represents the Linux prompt (remember, don't type this symbol, it's only included above to indicate that this is a Linux command). Note that the ampersand symbol (&) puts the process in "background" mode so that you can get back the Linux command prompt in the terminal window to carry out additional commands. Now copy the text in your web browser to the emacs GUI window, then save this file.

Now go back to the terminal window and issue the command:

```
> latex Tutorial03_LaTeX
```

– do not include the filename suffix ".tex" in this command. You will see a variety of output printed to the screen indicating any issues that the LaTeX compiler may have found. Following this, issue the Linux command:

```
> xdvi Tutorial03_LaTeX &
```

and again, don't include the ".dvi" filename suffix. Click on the xdvi GUI and examine the document that was created.

3 Sections, Tables, and Lists

This shows how one uses the \section command to make new sections. In addition to a page counter and figure counter, LaTeX also has a variety of other counters, including the 'section' counter. Section titles are labeled with current numeric section counter. For instance, to create the section title above, which is the third section of this tutorial, I just issued the following command in this LaTeX document:

```
\section{Sections, Tables, and Lists}
```

There are a variety of ways to create tables in LaTeX. We will first look at the tabular environment. The following is an example of using the tabular environment, first the coding in the actual LaTeX file, then how this will look in the output file (*i.e.*, the dvi and pdf files).

In the LaTeX file (*e.g.*, Tutorial03_LaTeX.tex):

```
\begin{tabular}{rccl}
\hline
\textbf{Decimal} & \textbf{Binary} &  & \textbf{Name} \\
\textbf{Value\ \ } & \textbf{Value}  & \textbf{Symbol} & \textbf{Description} \\
\hline
  0\hspace*{6mm} & 00000000 & NUL & Null character \\
  1\hspace*{6mm} & 00000001 & SOH & Start of Heading \\
  2\hspace*{6mm} & 00000010 & STX & Start of Text \\
\end{tabular}
```

The resulting table in the dvi and pdf files:

Decimal	**Binary**		**Name**
Value	**Value**	**Symbol**	**Description**
0	00000000	NUL	Null character
1	00000001	SOH	Start of Heading
2	00000010	STX	Start of Text

For the tabular environment, the number of columns in the table is indicated in the \begin {tabular} statement with the {rccl} option. This tells LaTeX to make a table with 4 columns, the first being right-justified, the 2nd and 3rd columns being center-justified, and the 4th being left-justified. \hline tells LaTeX to draw a horizontal line across the table. \textbf{} tell LaTeX to put the text in between the curly braces in boldface font. \hspace tells LaTeX to add a horizontal space of 6 millimeters, where the asterisk (*) force these spaces to be included. The ampersand symbol (&) marks the column divisions and the double backslashs tells LaTeX this is the end of the row. This LaTeX tutorial will discuss how to change the appearance of tables a bit later.

Another useful environment for making tables is the tabbing environment. The tabbing environment provides a way to align text in columns. It works by setting tab stops and tabbing to them much the way you do with an ordinary typewriter. The next page shows an example of using the 'tabbing' environment. In the LaTeX file, the first few lines of this environment looks like this producing the output on the next page):

```
\begin{tabbing}
XXXXXXXXXXXXXXXXXXXXXXXXXXXX \=    \kill
                             \>        \\
                             \>        \\
PRINCIPAL INVESTIGATOR:      \>  \rule[-0.8mm]{60mm}{0.2mm} \\
                             \>  Type your name here \\
                             \>                             \\
```

and the final line of this environment is indicated with \end{tabbing}.

PRINCIPAL INVESTIGATOR:	Type your name here
NAME AND ADDRESS OF THE ORGANIZATION:	East Tennessee State University Department of Physics and Astronomy Johnson City, TN 37614
SUBMISSION DATE:	19 September 2025
CLASS STATUS AND MAJOR:	Senior in Physics
MACHINE ARCHITECTURE:	Intel Core 2, 64-bit CPU
OPERATING SYSTEM:	Ubuntu Linux
PROGRAMMING LANGUAGE:	Fortran 77 and Python
GRANT OFFICIAL SIGNATURE:	Dr. Donald G. Luttermoser East Tennessee State University

In addition to tables, one often needs to display lists. The following shows the use of the `itemize` environment:

- High-dispersion spectra of the Mg II h and k resonance lines are a very sensitive diagnostic of temperature reversals in the atmospheres of stars due to their high opacity in relatively low-temperature regimes (∼5000 Å).
- The Fe II multiplets are a major source of cooling in these types of stellar atmospheres (Judge & Neff 1990).
- The line strengths of the C II] UV0.01 multiplet are sensitive to the electron density of the emitting region (Stencel *et al.* 1981; Lennon *et al.* 1985).

At this point in this LaTeX tutorial, change this itemized lists from a "bulleted" list to a numeric list. To do this, change the word "`itemize`" in the `\begin{itemize}` and `\end{itemize}` commands to "`enumerate`" so that these commands read: `\begin{enumerate}` and `\end{enumerate}`. Recompile your LaTeX document to see how this itemized list changes in appearance (as shown here):

1. High-dispersion spectra of the Mg II *h* and *k* resonance lines are a very sensitive diagnostic of temperature reversals in the atmospheres of stars due to their high opacity in relatively low-temperature regimes (∼5000 Å).
2. The Fe II multiplets are a major source of cooling in these types of stellar atmospheres (Judge & Neff 1990).
3. The line strengths of the C II] UV0.01 multiplet are sensitive to the electron density of the emitting region (Stencel *et al.* 1981; Lennon *et al.* 1985).

4 Equations

One of the things that makes LaTeX so useful to physicists, astronomers, and mathematicians is its ability to create equations and formulae in papers in a relatively easy way. There are a variety of ways to display mathematics in papers which we will work on here. If you need to include an equation in the main body of a paragraph, then use the dollar sign as delimiters around the equation, for instance \$F = m a\$ produces $F = ma$. Let's say that we want a little more space between the 'm' and the 'a' in this equation. We would then insert a small space with the '\,' marker, producing $F = m\,a$.

To make a numbered, center-justified equations, one uses the equation environment:

$$F_{\rm tot} = \sum_{i=1}^{N} F\left[t, r(t), v(t)\right] = ma = m\frac{dv(t)}{dt} = m\frac{d^2r(t)}{dt^2}\ . \tag{1}$$

If instead, we wanted an unnumbered center-justified equation, we would use the displaymath environment:

$$F_{\rm tot} = \sum_{i=1}^{N} F\left[t, r(t), v(t)\right] = ma = m\frac{dv(t)}{dt} = m\frac{d^2r(t)}{dt^2}\ .$$

If we need to display a set of equations that are numbered, we use the eqnarray environment:

$$v = \frac{dr(t)}{dt} = \dot{r} \qquad \text{(for velocity)} \tag{2}$$

$$a = \frac{dv(t)}{dt} = \frac{d^2r(t)}{dt^2} = \ddot{r} \qquad \text{(for acceleration),} \tag{3}$$

or if we would rather not have numbers associated with these equations, we would use the 'eqnarray$*$' environment:

$$\begin{eqnarray*}
v &=& \frac{dr(t)}{dt} = \dot{r} \qquad \text{(for velocity)} \\
a &=& \frac{dv(t)}{dt} = \frac{d^2r(t)}{dt^2} = \ddot{r} \qquad \text{(for acceleration),}
\end{eqnarray*}$$

Please note the use of the ampersand symbol (&) in the equation array environment to separate the left-hand side of the equation (typically with only one variable present), the center column containing the equal sign, and the right-hand side of the equation. LaTeX will make sure the center column with the equal signs line up in the same vertical column on the page.

5 Including Figures

Figures can be included in the LaTeX mark-up language as well. We can either have a figure by itself or in a figure environment with a figure number and a caption. There are a variety of ways to add figures to a manuscript. I will describe two here. The first I will call the "**Classical**" way using the \special command, and the other, I will call the "**Modern**" way using the graphicx package.

5.1 The "Classical" Way of Adding Figures

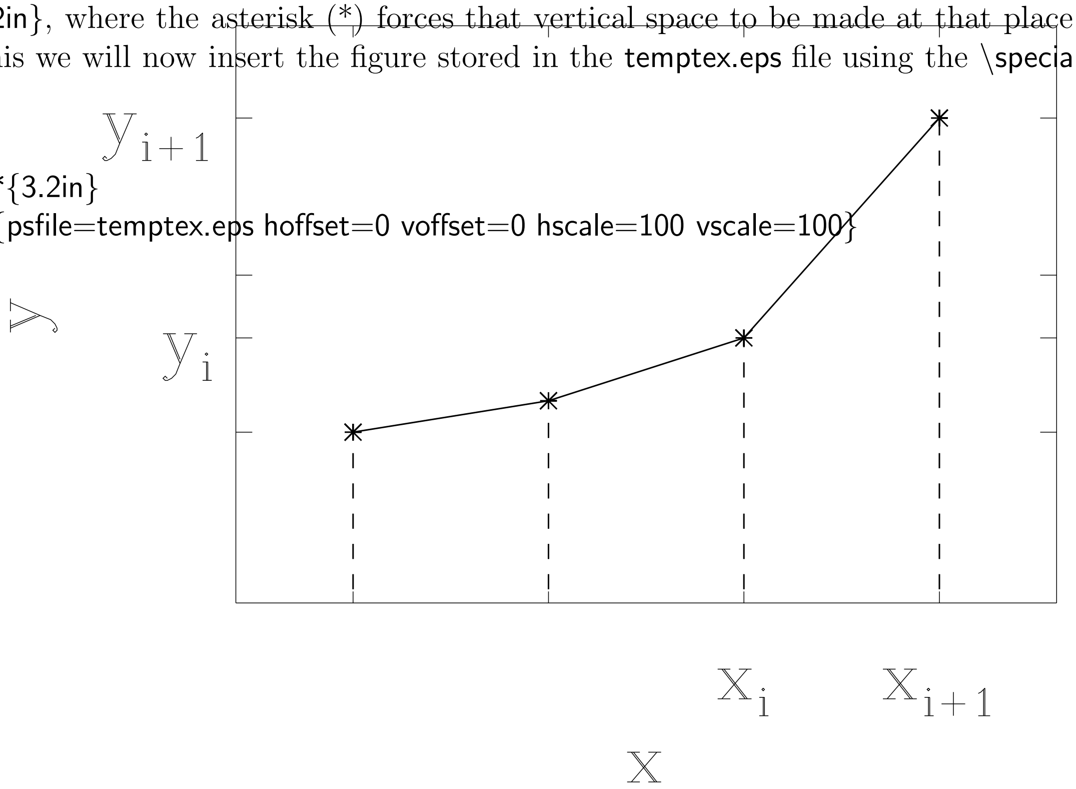
The \special command in LaTeX is a low-level command used to send instructions to the device driver (the program that converts the DVI output from LaTeX into a displayable or printable format like PDF or PostScript). In "classical" LaTeX, the \special command is often used to insert figures, typically stored in *encapsulated postscript* files (with a filename suffix of '.eps'). We will show how to use the \special command here.

First, from the Computer Programming Tutorial web page, download the file temptex.eps from the 'Programs and Supplemental Files for the Tutorials' table to the directory you are currently working (*i.e.*, the /tex subdirectory). Now go to your LaTeX file in the emacs GUI. Scroll down until you find the first instance of the \special command.

Note that we need to make some blank space in the document before inserting the figure. I typically assume a vertical size of the figure anywhere between 2.5 inches to 3.5 inches. Prior to the \special command, let's create a vertical space of size 3.2 inches with the command

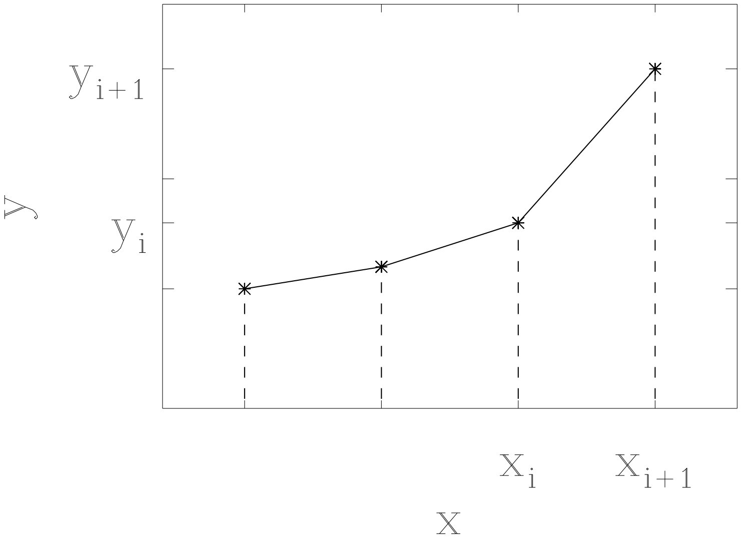
\vspace*{3.2in}, where the asterisk (*) forces that vertical space to be made at that place. Following this we will now insert the figure stored in the temptex.eps file using the \special command:

```
\vspace*{3.2in}
\special{psfile=temptex.eps hoffset=0 voffset=0 hscale=100 vscale=100}
```



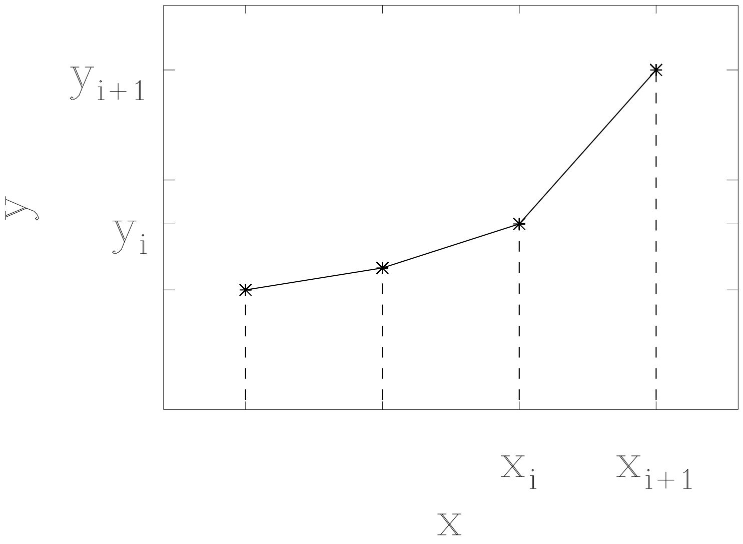
We see that this figure is a little too big and that it is not centered in the space that we have allowed for it. To correct this, let's first scale the figure down to 70% of its current size by setting hscale (horizontal scale) and vscale (vertical scale) from 100 to 70 and recompile:

```
\vspace*{3.2in}
\special{psfile=temptex.eps hoffset=0 voffset=0 hscale=70 vscale=70}
```



The figure now fits in the space we allowed for it, but it is not centered. Now we will reposition this figure so that it is centered in the space that we made for it. First we need to shift it downward — as such, change voffset (vertical offset) from 0 to -30 (where these numbers have units of 'points,' note that there are 72 points per inch on the document). Now we must center it horizontally. As can be seen, we need to shift this picture a bit to the right — as such, change hoffset (horizontal offset) from 0 to 30.

```
\vspace*{3.2in}
\special{psfile=temptex.eps hoffset=30 voffset=-30 hscale=70 vscale=70}
```



At this point, the figure is well placed for the space we have allowed for it.

5.2 The "Modern" Way of Adding Figures

As you can see, inserting figures in the "classical" way is not an *exact science* — there's a lot of experimentation to get the figure 'just right.' Due to the difficulties in handling graphics, figures, and pictures in LaTeX and TeX, the CTAN (Comprehensive TeX Archive Network) group have developed packages that can be used in either TeX or LaTeX documents. The three most popular are the following:

- The color package: LaTeX's color support is built around the idea of a system of *color models*. The color models supported by a driver may vary, but typically include:
  - rgb: Red Green Blue. A comma separated list of three real numbers between 0 and 1 giving the components of the color.
  - cmyk: Cyan Magenta Yellow, [K]Black. A comma separated list of four real numbers between 0 and 1 giving the components of the color according to the additive model used in most printers.
  - grey: grayscale. A single real number between 0 (black) and 1 (white).
  - named: Colors accessed by name selected from a list of predefined colors is used, *e.g.*, 'LightBlue'.
- The graphics package: The 'standard' graphics package.
- The graphicx package: The 'extended' or 'enhanced' graphics package.

One loads any of these packages in the *Preamble* of the LaTeX document, typically right after the \documentclass statement, *e.g.*,

```
\usepackage{graphicx}
```

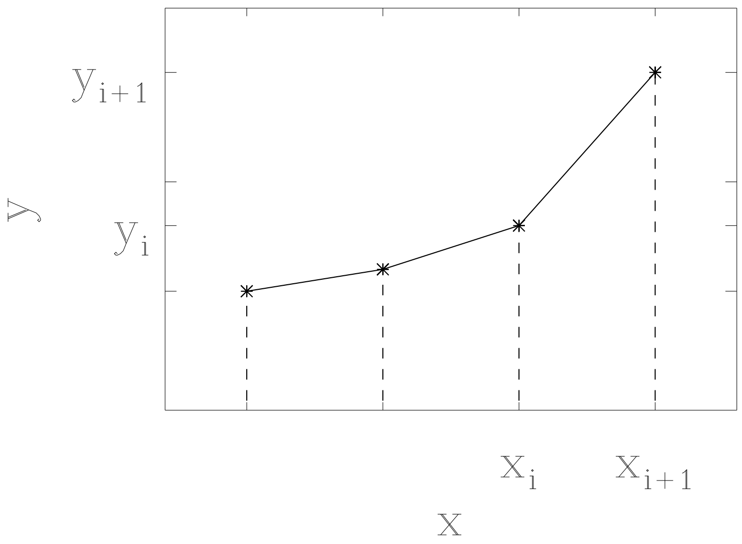
There are many options you can use with these packages which are too numerous to mention here. One can download a "*Guide to Graphics in LaTeX*" at the following website:

https://ctan.org/pkg/graphicx

To demonstrate the ease at including a figure with the graphicx package, all we have to do is issue one command in your LaTeX file:

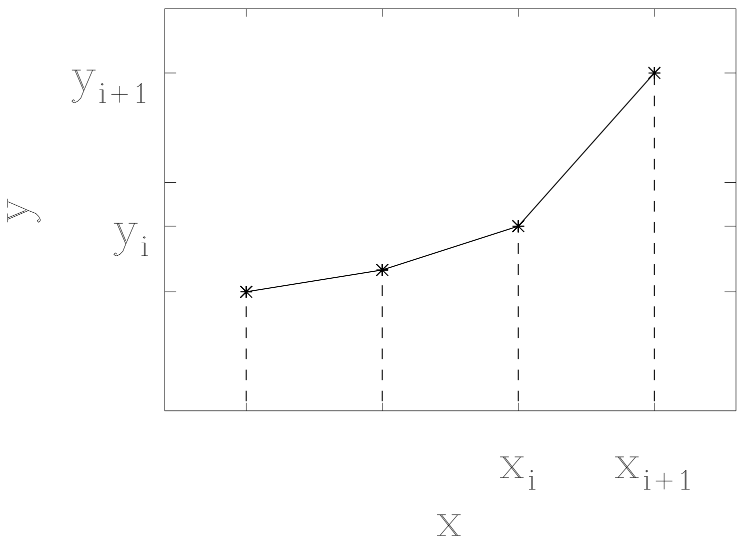
```
\includegraphics[width=0.75\textwidth]{temptex.eps}
```

where here we are using the ‘width’ option to scale the size of the figure to be 75% of the \textwidth of the document. Note that the vertical scale (height) will also be set to 75%. Here is the result of this command:



Note that the figure is not quite centered horizontally. We can fix this by including \includegraphics command in a ‘center’ environment:

```
\begin{center} \includegraphics[width=0.75\textwidth]{temptex.eps} \end{center}
```



Finally note that whereas the \special command typically requires the figure files to be Postscript files, the graphics and graphicx packages supports virtually all image file types (*e.g.*, JPEG [.jpg filename extension], GIF, TIFF, BMP, etc.) as well as Postscript files.

5.3 Adding Figures in a Figure Environment

What we have been doing so far is just inserting figures into the text of the manuscript. However, in textbooks and scientific manuscripts, figures are included with Figure numbers and captions. This is handled in LaTeX with the 'figure' environment. This environment is what is known as a *floating* environment, meaning that LaTeX will insert the figure where ever it 'feels' it looks the best. We can override this by placing positioning options with the \begin{figure} statement (t = top of page, b = bottom of page, h = current location in the text, p = placed on a separate page at the end of the manuscript).

For instance, the following lines of LaTeX code will produce the following figure environment in the paper:

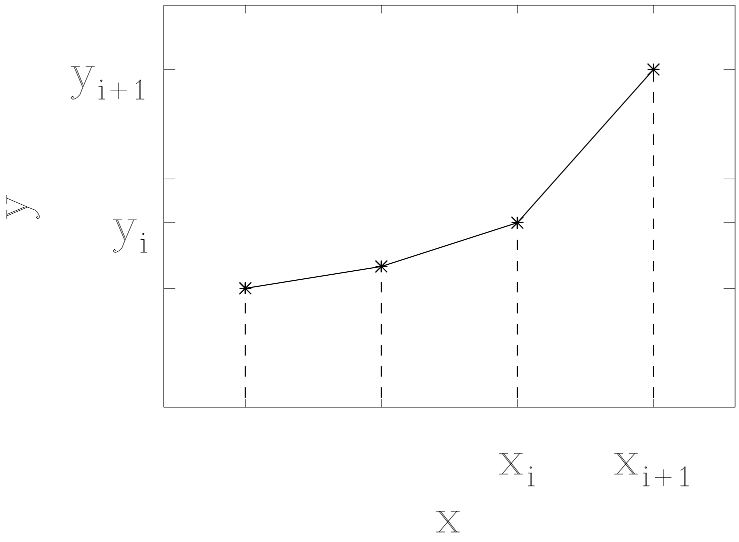


Figure 1: An Encapsulated Postscript Figure.

```
\begin{figure}[h]
\hspace*{10mm} \includegraphics[width=0.75\textwidth]{temptex.eps}
\caption{An Encapsulated Postscript Figure.}
\end{figure}
```

Note that instead of using the 'centering' environment for the figure placement, I forced a horizontal space of 10 millimeters in front of the figure, giving a better placement of the figure with respect to the figure caption.

5.4 Adding Picture Files

At this point, once again go to the Computer Programming Tutorial web page and now download the file temptex.bmp from the 'Programs and Supplemental Files for the Tutorials' table to the directory you are currently working (*i.e.*, /tex subdirectory)

As mentioned above, the graphics and graphicx packages allow a user to use any type of image, picture, and photograph files (*e.g.*, JPEG, GIF, TIFF, and BMP).

Note, however, that these picture file types do not include *size* information that LaTeX requires. Instead, one has to specify the size of the picture in the \includegraphics command. There are a variety of ways of handling this, including:

- Specify the size in units of measure (*e.g.*, mm, cm, inch):

  \includegraphics[height=60mm, width=100mm]{temptex.bmp}

- Specify the 'bounding box' locations in points (= 1/72 inches):

  \includegraphics[bb=0 0 320 240]{temptex.bmp}

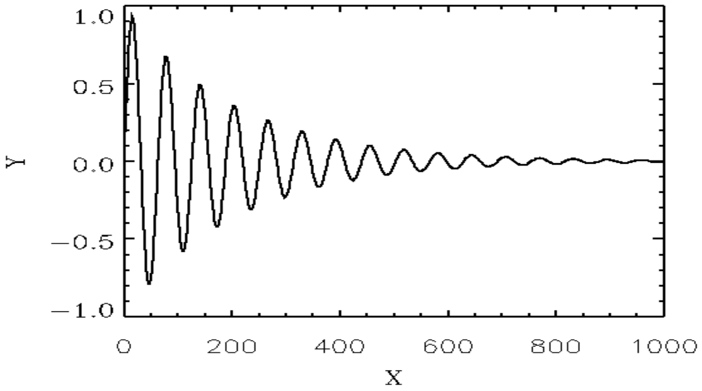


Figure 2: The above is a bitmap image.

The following lines of LaTeX produced the above figure (Fig. 2).

```
\begin{figure}[ht]
\centering
\includegraphics[height=60mm, width=100mm]{temptex.bmp}
\caption{The above is a bitmap image.}
\end{figure}
```

6 The Table Environment

We will next work with the table environment. Let's use the table we printed on Page 3 of this Tutorial and insert it in the table environment as shown below.

Decimal	**Binary**		**Name**
Value	**Value**	**Symbol**	**Description**
0	00000000	NUL	Null character
1	00000001	SOH	Start of Heading
2	00000010	STX	Start of Text

Now before the \begin{tabular}{rccl} line, insert the line \begin{table}, and after the \end{tabular}, insert the line \end{table}.

Next, let's center the table on the page. We do this by inserting the line \begin{center} after the \begin{table} line and enter \end{center} after the \end{tabular} line (and before the \end{table} line) and recompile the code at the Linux prompt. Click on the xdvi GUI to see how this has changed your document.

We now see that this table has move to the top of the following page. This occurred because the 'table' environment is *floating* just like the 'figure' environment. We can force the table to appear at a specific location by using the position options mentioned for the 'figure' environment.

We next want to add some vertical lines in this table to separate each column with a line and completely enclose the table in a box. To do this, input the '|' character in the \begin{tabular} line so that it reads:

```
\begin{tabular}{|r|c|c|l|}
```

then insert an \hline command just before the \end{tabular} line. Recompile and look at the output in the xdvi GUI.

Finally, let's include a caption for this table. In scientific journals, table captions typically come at the top of a table. After the \begin{center} line, insert the following line:

```
\caption{Portion of the ASCII Set of Symbols}
```

Recompile and examine the resulting dvi file. When all is said and done, the data table should appear as shown in Table 1.

Table 1: Portion of the ASCII Set of Symbols

Decimal Value	**Binary Value**	**Symbol**	**Name Description**
0	00000000	NUL	Null character
1	00000001	SOH	Start of Heading
2	00000010	STX	Start of Text

7 Special Fonts and Characters

At this point, go ahead and try different things with this LaTeX program, for instance, making some of the text boldface with the \textbf{ } environment or italics with the \textit{ } environment. Note that there is also a sans-serif font. The command for this is \textsf{ }. Go through this LaTeX document and change some of the phrases and/or sentences to each of these 3 fonts.

When a student first learns to create a manuscript by writing a program in LaTeX, one of the most difficult topics is realizing that you can't just clip and paste text from one document into the LaTeX file. This is because a LaTeX program must be written with standard ASCII characters (see §I.E of the notes). For instance, if you were to copy text from an external file with the pi symbol ($\pi$), it might show this symbol in the emacs GUI, but when you compile your LaTeX file, you will note that the $\pi$ symbol is not shown in the DVI or PDF file. Instead, we need to make the $\pi$ symbol in math mode: `$\pi$`.

In addition, some of the characters in standard ASCII (*i.e.*, the keys on an American keyboard) have special meanings in LaTeX. For instance the dollar sign, which put text following this symbol into math mode (as shown with the pi symbol above). To actually print the dollar sign, we must put a 'backslash' in front of the dollar symbol (\$). There are a significant number of these special characters in LaTeX which require using the backslash to print out these characters. The most common are $, #, %, {, }, _, and &. Also, the tilde symbol ($\sim$) is used to stop LaTeX from separating two words in a document at the end of a line. To actually print this symbol, you have to go into math mode and use the \sim command.

As I'm sure most of you have noticed, LaTeX (and TeX) commands begin with the '*backslash*' character (\\). You may be wondering how I am printing out this symbol in this tutorial, since LaTeX assumes 'backslash' indicates a command (or function). This is done by issuing the following command in math mode (*i.e.*, between the dollar sign, $, markers): `$\backslash$`.