

# L<sup>A</sup>T<sub>E</sub>X Tutorial

J. E. Rice

May 2010

## Abstract

The purpose of this document is to provide a simple example of how to use L<sup>A</sup>T<sub>E</sub>X. Examples of tables, figures, citations, references and math are shown, and some of the more commonly used document classes are discussed.

## 1 Introduction

L<sup>A</sup>T<sub>E</sub>X is a typesetting or mark-up language. That means you use symbols and mark-up notations to indicate how you want the document to look, and then some sort of compilation or processing must be used to generate the final document.

## 2 Running L<sup>A</sup>T<sub>E</sub>X

In L<sup>A</sup>T<sub>E</sub>X the processing may be done in either of two ways:

1. `pdflatex samplepaper.tex`

OR

1. `latex samplepaper.tex`
2. `dvips samplepaper.dvi`
3. `ps2pdf samplepaper.ps`

If you use the first option all your images must be in pdf format. If you use the second all your images must be in eps format. If you have citations in your paper you must add a few steps:

1. `pdflatex samplepaper.tex`
2. `bibtex samplepaper`
3. `pdflatex samplepaper.tex`

The first run of `pdflatex` (or `latex`) generates the list of citations that are needed to be found. Then `bibtex` finds them in the bibliography file, and the final run of `pdflatex` (or `latex`) includes them in the appropriate format to your document. Sometimes you even need to run `pdflatex` or `latex` again; it's a bit finicky.

There are a variety of front-ends available that may take care of some of these things for you; you'll have to search the web and see what's out there if you want to try that route.

### 3 The Format of a L<sup>A</sup>T<sub>E</sub>X document

You'll notice that the start of a L<sup>A</sup>T<sub>E</sub>X document can be a bit confusing. Often for journals or conferences a sample document is given; this is handy as you can use that as a starting point for your own work. Below is an example of a complete L<sup>A</sup>T<sub>E</sub>X document. It doesn't have much content, but it shows all the different pieces. Keep in mind that if you use a different document class the preamble may change – using the sample given is a good idea to keep track of what you will or will not need. See Section 13 for some information about the different types of document classes that are available.

```
\documentclass{article}
\usepackage{graphicx}

\title{title goes here}
\author{author goes here}
% leave the parentheses empty if you don't wish an author
\date{May 2010}
% if you leave out this line LaTeX will fill in today's date.
% leave the parentheses empty if you don't want a date

\begin{document}
\maketitle

\begin{abstract}
Abstract goes here, if desired. Abstracts are often optional.
\end{abstract}

\section{Introduction}
\label{sec: Introduction}
Introduction goes here.

% HERE IS THE BIBLIOGRAPHY. This is optional.
\bibliography{samplepaper}
\bibliographystyle{alpha}

\end{document}
```

### 4 Special L<sup>A</sup>T<sub>E</sub>X Symbols

There are certain symbols you cannot use as you normally would in a L<sup>A</sup>T<sub>E</sub>X document: # % \$ & ~ \_ ^ \ { }. So if you want to have a percent sign, you have to put a slash in front of it: \% will appear as %. The percent sign is normally used to indicate a comment, for example,

```
%FINISH THIS SECTION BEFORE SUBMITTING TO JACKIE!
```

The dollar sign is a short-cut to enter math mode; you'll see this in Section 7. The others are not that commonly used for their special purposes, you just have to remember to put the slash in front of them if you need them in your text.

### 5 Sections and creating References to them

You usually need to break your document into separate sections. These could be sections and subsections, as in a report or a paper, or they could be chapters, sections, and subsections, as in a book (*e.g.* your thesis). You need to look at the particular document class you are using to see what commands are appropriate. Usually,

as in this document, the command for a new section is `\section{Section Name}`, but some journals will give you a document class where this is changed to `\Section{Section Name}`. Numbering is done automatically by  $\LaTeX$ ; if you need to change it to roman numerals or alphanumeric you can do this. Look in one of Leslie Lamport's books, such as [Lam94].

What is very handy in  $\LaTeX$  is the ability to refer to a particular chapter or section without remembering its number. As you write your thesis you'll find you want to rearrange things a lot. In fact, you might even want to put each chapter into a separate file, then you won't need to cut and paste if you need to move chapter 10 to chapter 3, for instance. You can ask me how to do that later. For now, let's say you need to remind me (or your reader) that you discussed something back in Section 1. You might write this: "An example of computing the autocorrelation function is given in Section 1." The  $\LaTeX$  code looks like this:

```
'An example of computing the autocorrelation
function is given in Section~\ref{sec: Introduction}.'
```

Back in the Introduction I had to put in a label that looked like this:

```
\label{sec: Introduction}.
```

This works for figures, tables, chapters, sections, and subsections. The label command can go anywhere and  $\LaTeX$  will figure out which section you need to refer to.

## 6 Lists

Lists are very easy in  $\LaTeX$ . There are two types: enumerated lists (*i.e.* numbered lists) and unordered lists (*i.e.* bulleted lists).

```
\begin{enumerate}
\item number one
\item number two
\end{enumerate}
```

will look like this:

1. number one
2. number two

```
\begin{itemize}
\item number one
\item number two
\end{itemize}
```

will look like this:

- number one
- number two

Other types of lists, for instance descriptive lists, are also available, and you can nest lists as well. See the Lamport book or look on the web for more info.

## 7 Math

You can do anything in  $\LaTeX$  that you will ever need to when it comes to math. Matrices, formulas, equations, *anything*. You start math mode with `\begin{displaymath}` or `\begin{equation}` and end it

with `\end{displaymath}` or `\end{equation}`. These both start a separate, centered area for your math, for example

$$B(u) = \sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus u)$$

is written as

```
\begin{displaymath}
B(u) = \sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus u)
\end{displaymath}
```

The equation environment puts in a number, for example

$$B(u) = \sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus u) \tag{1}$$

is written as

```
\begin{equation}
B(u) = \sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus u)
\end{equation}
```

If you want an inline equation, for instance  $x^2$  then use the dollar signs to start and end the math as shown below.

For the actual math formatting there is far too much for me to show you in a short tutorial, so looking on the web or reading the Lamport book [Lam94] will be essential as you put together your reports or theses. Some simple things that are commonly used are:

- subscripts:  $x_1$  is written as `$x_1$`.
- superscripts:  $x^2$  is written as `$x^2$`.
- fractions:  $\frac{1}{2}$  is written as `$$\frac{1}{2}$$`.
- summations:  $\sum_{v=0}^n$  is written as `$$\sum_{v=0}^n$$`.
- greek letters:  $\alpha, \beta, \gamma, \rho, \pi$  etc. are written as `$\alpha`, `\beta`, `\gamma`, `\rho`, `\pi`.

## 8 Citations

L<sup>A</sup>T<sub>E</sub>X makes it very easy to keep track of your citations and also to change citation styles. You can also keep your references in a tool such as EndNote and export them to a L<sup>A</sup>T<sub>E</sub>X style bibliography. Citations are referenced as follows: `\cite{Lamport94}`. A matching bibliography item must be in your bibliography file; it might look like this:

```
@book{Lamport94,
title={\latex: A Document Preparation System},
author={Leslie Lamport},
publisher={Addison-Wesley},
year={1994}
};
```

Also at the end of your L<sup>A</sup>T<sub>E</sub>X document you have to tell the system where to find that file and what style of citations you need:

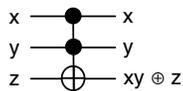


Figure 1: A Toffoli gate.

column 1	column 2	column 3
1	2	3
3	2	1

Table 1: An example table.

```
\bibliography{samplepaper}
\bibliographystyle{alpha}
```

I’ve used this all through-out this paper. A note on citations is useful at this point. A citation should appear immediately after the information being cited, with one space separation. In  $\text{\LaTeX}$  we often “join” the citation to the material with a `~`. This creates a space but does not allow a line break. Thus I might say “The material from this tutorial is based on the famous Lamport book [Lam94]”, for which the  $\text{\LaTeX}$  code is

```
‘‘The material from this tutorial is based
on the famous Lamport book~\cite{Lamport94}’’
```

More information about bibliographies, bibtex, and generally using  $\text{\LaTeX}$  for bibliographies can be found at this website: <http://amath.colorado.edu/documentation/LaTeX/reference/faq/bibstyles.html>

## 9 Figures

In order to include figures I use the package called `graphicx`. Figure 1 is included using the following  $\text{\LaTeX}$  code:

```
\begin{figure}
\center
\includegraphics[width=3cm]{sample}
\caption{\label{fig: sample}A Toffoli gate.}
\end{figure}
```

For this figure and because I am using `pdflatex` the file called “sample.pdf” has to be in the same directory as the  $\text{\LaTeX}$  file I am working with. If I was using `latex` then I would need a file called “sample.eps”. It is worth noting that postscript (eps) files often appear clearer and of better quality. The width command inside the `[ ]` is optional, but it can be used to resize your figure. This is very useful for making figures smaller for two-column articles. The center command ensures that the figure is centered, and you’ll note that the caption command has a label command inside it. This isn’t essential, but I find it ensures that the correct figure number is used when I reference the figure. I referenced the figure using this code:

```
Figure~\ref{fig: sample}
```

## 10 Tables

Tables are created using the `table` command and the `tabular` command. Table 1 shows an example. The  $\text{\LaTeX}$  code to create this table is shown below.

```

\begin{table}
\center
\begin{tabular}{c | c | c}
column 1 & column 2 & column 3 \\ \hline
1 & 2 & 3 \\
3 & 2 & 1
\end{tabular}
\caption{\label{tbl: sample}An example table.}
\end{table}

```

You'll notice the use of the center command and the caption command; these are used in the same way that we used them for figures. The tabular command sets up the structure of the table, while the table command sets up how the caption will appear and the numbering for the table. When referring to a table I used this L<sup>A</sup>T<sub>E</sub>X code:

```
Table~\ref{tbl: sample}.
```

You'll also notice that L<sup>A</sup>T<sub>E</sub>X likes to put tables and figures at the top of the page. If you don't like this you can use the [h] (for here) option when you create the table (or figure) like this:

```

\begin{table}[h]
\center
\begin{tabular}{c | c | c}
column 1 & column 2 & column 3 \\ \hline
99 & 29 & 39 \\
87 & 56 & 123
\end{tabular}
\caption{\label{tbl: sample2}A table that is not at the top of the page.}
\end{table}

```

The resulting table is shown as Table 2. Other options are [t] for top, [b] for bottom, or you can combine them *e.g.* [hb] to let L<sup>A</sup>T<sub>E</sub>X decide which of those two places it likes best.

column 1	column 2	column 3
99	29	39
87	56	123

Table 2: A table that is not at the top of the page.

## 11 Fonts

We don't change fonts a lot in L<sup>A</sup>T<sub>E</sub>X. The most common thing you might want to do is have a word in *italics*. Here's the L<sup>A</sup>T<sub>E</sub>X code for this: `{\em italics}`. **Bold font** is sometimes used, and here's the code for that: `{\bf Bold font}`.

You sometimes might want to make fonts large or really large, or small or tiny. The list below shows how each of these are done.

- `{\large large}` appears as large
- `{\Large really large}` appears as really large
- `{\huge huge}` appears as huge
- `{\Huge really huge}` appears as really huge

- `{\small small}` appears as small
- `{\tiny tiny}` appears as tiny

## 12 Tips and Tricks

### 12.1 Adding (or Removing) Extra Space

Sometimes things just won't appear the way you want. Maybe there is too much space, or too little space. You can force extra horizontal space using `\hspace*{2cm}`. Either cm or in can be used as the units. Here's what this command will do in between the words "start" and "finish" below:

start                      finish

Here's what the  $\LaTeX$  code looked like:

```
start \hspace*{2cm} finish \\\
```

Forcing vertical space is sometimes handy too. Keep in mind negative numbers can be used to remove space, but the result is sometimes unpredictable. If I put `\vspace*{3cm}` between the words "start" and "finish" below here's what will happen:

start

finish

Here's the  $\LaTeX$  code:

```
\noindent start\\
\vspace*{3cm}\\
\noindent finish\\
```

### 12.2 Line Breaks and Page Breaks

Note that in the above examples I forced line breaks by using `\\` at the end of each line. In my draft version I also moved the page break where I wanted it to occur by using `\newpage`, although in my final version I took it out. *Normally we let  $\LaTeX$  deal with page and line breaks!!!*  $\LaTeX$  was designed to put page breaks and line breaks in the most logical place. However sometimes we need special formatting and in this case `vspace`, `hspace`, `newpage` and line breaks can come in handy.

### 12.3 Paragraphs in $\LaTeX$

Paragraphs occur when  $\LaTeX$  sees an empty line in the text. So this:

Blah blah blah

New paragraph

creates a new paragraph, while this

Blah blah blah

continues from above

does not. Try it for yourself.

## 12.4 Including text Verbatim

In really special cases we want L<sup>A</sup>T<sub>E</sub>X to ignore all the normal rules and just put the text into the document EXACTLY the way we typed it. This is particularly useful for pseudocode. In this case the verbatim environment is used. A short form of the verbatim environment is the `verb` command; I've used that all the way through to show you my L<sup>A</sup>T<sub>E</sub>X code. Figure 2 shows an example of the verbatim environment. You don't have to put it in a figure, but of course you can if you want.

```
\begin{verbatim}
  If I wrote a poem *
      that I wanted indented @
  and with &
      special line breaks ^
  and symbols %
      I might use the _
  verbatim environment \
\end{verbatim}
```

Figure 2: An example of the verbatim environment.

Here's an example of the `verb` command:

```
\verb^symbols you can't normally use: $ & \^
```

You'll notice that the `verb` command uses two symbols to start and end the stuff you want included verbatim; these can be any two symbols as long as they match.

## 13 Document Classes and Packages

### 13.1 Document Classes

There are a number of built-in document classes, including `article`, `report`, `book`, and `slides`. I nearly always use `article` or `report` for mine. The standard layout as shown in Section 3 should apply to most of these.

For each of these document classes there are options that you may use. For instance you can change the font size for the document, the paper size, or the number of columns used. Options are given in square brackets before the name of the class, for example

```
\documentclass[twocolumn]{article}
```

Try including this option on this document and see what happens.

As I've mentioned before, journals and conferences will often supply their own documentclass to use. If that is the case then there may be different options that are available, and documentation is always supplied.

### 13.2 Packages

Lots of people have added functionality to L<sup>A</sup>T<sub>E</sub>X and these additions are usually available in the form of packages. For instance, you'll notice that there are two packages used in this document:

```
\usepackage{graphicx}
\usepackage{fullpage}
```

The `graphicx` package deals with including graphics, usually for figures, while the `fullpage` package causes L<sup>A</sup>T<sub>E</sub>X to use more of the page and have smaller margins. There are millions of packages out there; some are part of the standard L<sup>A</sup>T<sub>E</sub>X installation (such as `graphicx` and `fullpage`) and some need to be installed separately if you want to use them.

There is more information on a variety of classes and packages at this website:  
<http://www.math.uiuc.edu/~hildebr/tex/packages.html>

## 14 Writing in L<sup>A</sup>T<sub>E</sub>X

Writing a document in L<sup>A</sup>T<sub>E</sub>X is a lot like programming:

- planning is essential – I often write the section headings first, then go back and fill in the text
- save often
- compile (*i.e.* run L<sup>A</sup>T<sub>E</sub>X) often – that way you can fix errors before you get too far along
- worrying about formatting last; just get the content written to start with, then go back and add line breaks, center your figures, fix the math, and so on.

## 15 Conclusion

L<sup>A</sup>T<sub>E</sub>X is an essential tool for writing any sort of document that needs to be professional and contains a lot of math. Both of these are common requirements in Computer Science. I use a simple text editor (vi) to edit my L<sup>A</sup>T<sub>E</sub>X documents, but one can also download from the web any number of L<sup>A</sup>T<sub>E</sub>X front ends that will help you with editing your L<sup>A</sup>T<sub>E</sub>X files. The L<sup>A</sup>T<sub>E</sub>X software is available on our linux machines, but you can also install it on your pc or mac if you wish; I suggest searching the web for help on this.

This document will get you started in L<sup>A</sup>T<sub>E</sub>X but you will need more help as you develop more complicated documents. There is lots of information on the web, but I strongly recommend a reference book to refer to, such as the Lamport book [Lam94].

## References

[Lam94] Leslie Lamport. *L<sup>A</sup>T<sub>E</sub>X: A Document Preparation System*. Addison-Wesley, 1994.