An Introduction to LaTeX

LaTeXis an extremely powerful tool for creating beautiful mathematical documents. This set of notes is a starting point for LaTeX – it is easy to get overwhelmed at first, but we'll break things down into some small topics to get you started.

LaTeX is a *markup language*, like HTML. This means that typing a document in LaTeX is just text, with some extra commands to tell LaTeX how to format the text.

1 Setting everything up

We'll go over three ways to set up LaTeX on your computer:

1. Use an online editor (easiest to set up, but requires a constant internet connection)
2. Install LaTeX on your computer
3. Install LaTeX on a USB flash drive

When I say "set up LaTeX ", you should be aware that we are really installing two programs that work together: an *editor* where you will do all of your typing, and a *compiler* that translates the ugly LaTeX code into a nice PDF document. The compiler works entirely in the background, and if you're lucky, you'll only set it up once and never worry about it again. The editor is what you will be working with.

1.1 Use an online editor

There are two well-known online LaTeX editors: ShareLaTeX (`http://www.sharelatex.com`) and WriteLaTeX (`http://www.writelatex.com`). Both are good, they each have their own pros and cons. Both offer free accounts, and you'll want to make an account right away in order to store your files online.

The good news is that other than creating an account, there is nothing to set up! You can just begin typing. If you are not computer savvy, this is definitely the way to go. The downside is that you have to be online and logged into ShareLaTeX or WriteLaTeX in order to type your documents. If your internet connection is down or just slow, you will have a hard time. Finally, using an online editor means you have a little less control over your documents.

1.2 Installing LaTeX on your computer

LaTeX is freely available on any computer. Windows users will want to use MikTeX (`http://www.miktex.org`); Mac users will use MacTeX (`http://www.tug.org/mactex/`).

I don't use a Mac, so I can only help with Windows. I'm told that setting up LaTeX on a Mac pretty straightforward, though.

On Windows, you will want to download and install the "Basic MikTeX installer". You can choose all of the default options, with the exception that you may want to choose a default page size of "Letter" rather than "A4." Make sure that you choose the option to "install packages on the fly." The program will install *both* the compiler (MikTeX) and an editor, called TeXworks. TeXworks is the program

you will actually be using to type your documents, in theory you should never have to touch MikTeX itself again.

One thing you should be aware of: you will often make use of *packages* which extend the capabilities of LaTeX . The first time you use a new package, you will need to make sure that you are connected to the internet so that MikTeX can download the new package. Once the package has been downloaded, you can happily compile your documents whether or not you are connected to the internet.

1.3 Installing LaTeX on a USB flash drive

This is pretty much the same as installing LaTeX on your computer- just make sure to choose the "portable" installer at MikTeX's website. Unfortunately, this is Windows-only, there is no portable version of MacTex. When you run the program from your flash drive, a small icon will appear in the right side of your Windows taskbar. Right click on that icon to access TeXworks and other options.

The same caveat applies with regard to packages: make sure you are connected to the internet the first time you use a new package!

2 *Your first LaTeX documents*

The easiest way to get started is by looking at some examples, so let's get started!

Example 1: Suppose we want to type the message, "Hello world!"

We can start by just typing,

```
Hello World!
```

which is a fine text document, but LaTeX has no idea what kind of formatting we want to put onto this text. In order to give LaTeX some idea as to how to format it, we need a minimal set of commands. First, we need to tell LaTeX what kind of document we are writing, using a `\documentclass` command. Second, we need to tell LaTeX where the document begins and ends. To do so, we'll use `\begin{document}` and `\end{document}`. Notice that LaTeX commands always begin with a backslash (\).

Putting it all together, we get:

```
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```

Go ahead and type this into your editor, and compile it. You should get a nice pdf file with the words, "Hello World!"

If you are using TeXworks, you will want to set up a couple of options to make your life easier. First, enable the spell checker under Edit→Spelling→English - United States. Next, turn on context formatting by choosing Format→Syntax Coloring→LaTeX. This will make it easier to spot LaTeX commands by highlighting them in different colors.

Without knowing anything else at all, you can now type documents using LaTeX– All you need to do is to type anything you like between the `\begin{document}` and `\end{document}` statements.

Let's look at another, meatier example:

This example has a much more extensive *preamble* (see below). I hardly ever write a preamble from scratch, I just cut-and-paste the preamble that works for me into every new document I write. This has the advantage of getting everything set up exactly the way I like it every time.

Example 2:

```
\documentclass[12pt]{article} % 12 point fonts
\usepackage[margin=1in,letterpaper]{geometry} % 1 in. margins
\linespread{1.1} % slightly larger line spread
\usepackage{mathpazo} % nicer fonts
\usepackage{latexsym} % Extra symbols if we need them
\usepackage{amsmath} % extra math commands
\usepackage{color} % allows color commands
\usepackage{fancyhdr} % nice headers
\usepackage[shortlabels]{enumitem} % nicer lists
\setlist[itemize]{leftmargin=*,itemindent=*} % adjust list labels
\setlist[description]{leftmargin=*,itemindent=*}
\setlist[enumerate]{leftmargin=*,itemindent=*}
\usepackage{graphicx} % import graphics
\usepackage{tikz} % draw graphics
\usepackage{lipsum} % filler text

\begin{document}

The variables $x$, $y$, and $z$ are defined such that
\[x^2-y^2=z^2\]

Here are \emph{three} things you should \emph{know}:
\begin{enumerate}[{Thing }1:]
\item This thing
\item That thing
\item The other thing
\end{enumerate}

\lipsum[1-2]

\end{document}
```

Cut and paste this into your editor and compile it. Notice all the fancy formatting!

A couple things to notice:

1. The *preamble* is everything that comes before the `\begin{document}` command. This is where "global" formatting commands go.
2. I added in some comments using the % symbol. This allows me to put messages in the file that do not appear in the final document. It's useful to put in reminders of what commands do.

3. Everything that you *begin* in LaTeX must also *end*. Every `\begin` must also `\end`, every `{` needs a `}`, etc. This is the *most* important rule to remember!

We can make formatting changes that apply to the entire document by putting them in the preamble.

Example 3: Add the line below to the preamble of your previous example, and see what happens:

```
\let\emph\textbf
```

All the emphasized text is changed from italics to bold! This allows you to easily make changes to formatting throughout the entire document.

3 *A more advanced example*

Let's look at another example with a lot of math:

Don't try to read through this example, just cut and paste it into your editor and see what happens. This example is just meant to give you some idea of the math commands that LaTeX can use. When you see a symbol you want to use, you can look at the code and figure out what command gave the symbol you are interested in.

Example 4: Copy and paste this into your editor, and compile it:

```
\documentclass[12pt]{article}

\usepackage[margin=1in]{geometry}
\usepackage{amsmath,amssymb}
\usepackage[shortlabels]{enumitem}

\begin{document}
Do union and intersection distribute over set difference?
In other words, is it true that:
\begin{enumerate}[1.]
\item $A \cup (B \backslash C)=(A \cup B) \backslash (A \cup C)$ and
\item $A \cap (B \backslash C) = (A \cap B)\backslash (A \cap C)$
\end{enumerate}
\begin{enumerate}[{Answer to }1.]
\item No, set difference does not distribute over union.
Here is one counter example: Let $U=\{1,2,3,4,5\}$, $A=\{1,2,3\}$,
$B=\{2,3,4\}$, and $C=\{4\}$. Then $A \cup (B \backslash C)=\{1,2,3\}$
 and  $(A \cup B) \backslash (A \cup C)=\{1\}$. Since these sets
are not equal, we have a counterexample and
 $A \cup (B \backslash C) \neq (A \cup B) \backslash (A \cup C)$.

\item Yes, set difference does distribute over intersection.
\textbf{Proof:} (This is only an outline. You can fill in
the details.) Let $A$, $B$ and $C$ be sets. Then:
\begin {align*}
x \in A \cap (B \backslash C) &\Leftrightarrow x \in A
\text { and } x \in B \backslash C \\
&\Leftrightarrow x \in A \text{ and } x \in
B \text { and } x \not \in C \\
&\Leftrightarrow x \in A \cap B \text { and }
x \not \in A \cap C \\
&\Leftrightarrow x \in (A \cap B) \backslash
(A \cap C) \\
\end{align*}
\end{enumerate}
\end{document}
```

There is a lot going on in this example! If you compare the LaTeX code to the pdf file generated by the compiler, you'll notice that many

of the mathematical symbols have somewhat intuitive commands, like `\backslash` for $\backslash$, `\cup` for $\cup$, and so on. Also notice that we are using the *enumerate* environment for making nice lists, with some custom labels. Here we are taking advantage of the *enumitem* package, which allows for those nice custom labels. The *align** environment is very useful for typing long chunks of mathematical notation- here we are using logical symbols, but it works nicely for algebraic manipulations as well. Finally, the `\text{}` command allows us to insert small chunks of text into math mode. The `\text{}` command is part of the *amsmath* package.

4 *Graphics*

Finally, let's look at graphs and graphics. The easiest way to include graphs (or other graphic) is to create a .png or .jpg file, and have LaTeX import it into the document. First, we need to create the graph. You can create the graph in any mathematical graphing program, such as Microsoft Mathematics (for Windows), or through online websites like GeoGebra (www.geogebra.org) or Desmos (www.desmos.com). No matter how you create the graph, you want to save the graph file as a .png or .jpg file in the same folder where you keep your .tex file. If you are using an online LaTeX editor like WriteLaTeX or ShareLaTeX, you will need to upload the .png or .jpg file to the website.

Another option is to use LaTeX commands to create your graphic within LaTeX itself. This is accomplished using the *tikz* package. It's quite complicated, but it is useful because then you don't have to worry about an extra .png or .jpg file to keep track of. I won't go over it here, but you can find lots of examples online.

Once you have all that done, it's time to tell LaTeX to insert the picture. First, we'll need to make sure that the *graphicx* package is in the preamble:

```
\usepackage{graphicx}
```

Then, you'll need to put your graph in the document. If your graph is named "filename.png", you will use the command

```
\includegraphics[width=\textwidth]{filename.png}
```

Make sure to include a blank line above and below the `\includegraphics` command, so that it does not become part of one of your other paragraphs. You can replace the `\textwidth` command with a length in inches to make the graph smaller or larger; the `\textwidth` command makes the graph the width of the document, within the margins.

5 *More resources*

This should get you started. There are plenty of resources available online to help. Some of the most helpful are

1. The Not So Short Introduction to LaTeX (http://tobi.oetiker.ch/lshort/lshort.pdf): This is a quite lengthy manual containing all sort of information about different commands, environments, and packages that you may find useful. It's not meant to be digested all at once, but you can use it as a handy reference.
2. DeTeXify (http://detexify.kirelabs.org/classify.html): This website allows you to draw symbols with your mouse, and will (usually) look up the LaTeX command that gives you that symbol. It will even tell you what packages you need to use!
3. Online LaTeX Equation Editor (http://www.codecogs.com/latex/eqneditor.php): Here, you can try out small snippets of math mode code to make sure it works the way you want it to.
4. TeX at StackExchange (http://tex.stackexchange.com/): StackExchange is a Q&A site where a large community can help with any questions about LaTeX that you may have. Most basic questions already have answers that you can search for.

Good luck! I think you will find that within a few weeks, LaTeX will feel much easier to use.