Learning LaTeX: The Basics

The best way to learn LaTeX is by trial and error, with a lot of experimenting, and using other people's .tex files as a model. Google is also a good source: for example, googling "latex integral" immediately gives a bunch of links explaining how to do integrals in LaTeX.

Even though learning on your own is the best, there are certain basic things that come up so often that it might be useful just to read about them. This document is a compendium of some of these things.

Basic structure of a LaTeX file

LaTeX files begin with a \documentclass command, and then there are a bunch of weird commands giving special instructions to the LaTeX compiler. Everything until one gets to the \begin{document} command falls into this category and is called the "frontmatter" of the LaTeX file. You are encouraged to ignore this frontmatter when you are first learning the software.

The main part of the LaTeX file occurs between a \begin{document} command and an \end{document} command. This is the part of the file that actually corresponds to printed text in the final output.

It is sometimes useful to include comments in a LaTeX file which are not part of the printed document, but rather are notes or instructions to the author. To do this, use the % symbol. Anything on a line after a % symbol is ignored by the LaTeX compiler.

Commands for spacing

The first thing to know is that skipping multiple lines in your .tex file has absolutely no effect on the final output. TeX is designed to ignore almost all "blank space" in the .tex file. So feel free to skip lines to make the file easier for you to read. If you want to produce some skipped lines in the final document use "\smallskip" or "\medskip" or "\bigskip" or something like "\vspace{2in}". All of these commands produce vertical spaces of various sizes.

When LaTeX encounters a blank line in the .tex file, it interprets this as a command to start a new paragraph and automatically indents. Sometimes you *don't* want it to start a new paragraph, and so you use the \noindent comment.

Mostly LaTeX is good about figuring out page layouts on its own, but sometimes you want to *force* it to start a new page. This is done with the \newpage command.

Sometimes you want to add a tiny bit more horizontal space to make something look right. For example,

$$\int \sqrt{1+x^2}dx$$

looks a little cramped to me and I prefer

$$\int \sqrt{1+x^2}\,dx$$

or maybe even

$$\int \sqrt{1+x^2}\, dx.$$

The command "\," says "add a tiny bit more space" whereas the command "\ " (there is a space after the backslash) says add a full space. The above three integrals were produced by the following three lines of LaTeX code:

(1) \$\int \sqrt{1+x^2} dx\$

(2) \$\int \sqrt{1+x^2}\, dx\$

(3) \$\int \sqrt{1+x^2}\ dx\$

In math mode you can also use the commands \quad and \qquad to produce horizontal spacing of increasing sizes. For example,

$$x^2+y^2=1, \qquad\qquad a^2+b^2=1$$

was produced with the code

\[x^2+y^2=1, \qquad\qquad a^2+b^2=1 \]

Basics of math mode

Any math symbols must be enclosed within a pair of dollar signs: there has to be a dollar sign at the beginning to tell TeX to enter math mode, and a dollar sign at the end telling TeX to exit math mode. One of the most common errors is to leave out a matching dollar sign, and then TeX gets all confused.

There is regular math mode, which occurs within paragraphs, and there is also "displayed math mode" which puts math stuff on its own line, with a bit of extra space, and centered in the middle of the page. For example, the code " \[8x+9y+10z=1 \]" produces the output

$$8x+9y+10z=1.$$

The command \[means "enter displayed math mode" and \] means "leave displayed math mode". They have to appear together. Note also that you never want to mix regular math mode (dollar signs) with displayed math mode (\[and \])—use one or the other, never both at the same time.

Miscellaneous math stuff

Root symbols: To produce $\sqrt{x^2+y^2}$ you would use

\sqrt {x^2+y^2}

whereas to produce $\sqrt[3]{x^2+y^2}$ you would use

\sqrt[3] {x^2+y^2}

Matrices: The following LaTeX code typesets a matrix:

\begin{bmatrix}

5 & x & 17 \\

y & 8 & e^2 \\

\end{bmatrix}

This has to be in math mode or displayed math mode. The & signs say 'go the the right one entry', where as the \\ signs say 'go to the next row and start over'. So the above code produces the matrix

$$\begin{bmatrix} 5 & x & 17 \\ y & 8 & e^2 \end{bmatrix}$$

The code for a 3×1 matrix would look like this:

\begin{bmatrix}

x \\

y \\

z \\

\end{bmatrix}

whereas the code for a 1×3 matrix would look like

\begin{bmatrix}

x & y & z \\

\end{bmatrix}

Size of delimiters: If we write

$$f(\frac{3}{5}) = \frac{7}{8}$$

then it looks weird. We need to be able to tell LaTeX to use bigger parentheses, or bigger brackets. The commands that do this come in four sizes:

\big, \Big, \bigg, \Bigg.

The code

```
\[ f \Big ( \frac{3}{5} \Big )=\frac{7}{8} \]
```

produces the output

$$f\Big(\frac{3}{5}\Big)=\frac{7}{8}$$

whereas the code

```
\[ f \Bigg ( \frac{3}{5} \Bigg )=\frac{7}{8} \]
```

produces the output

$$f\left(\frac{3}{5}\right)=\frac{7}{8}.$$

The big/Big/bigg/Bigg 'size commands' can be used with parentheses, brackets, set brackets, the '|' sign, and various other delimiters. Some people like to write '\Biggl' and '\Biggr' to distinguish between left and right delimiters, as this sometimes makes the spacing work a little nicer, but it is very subtle.

Note that to make a set bracket you have to type "\{" because LaTeX thinks that { by itself is an internal grouping, not text. For example, the text

```
\[ \Bigl \{ A\in M_{2\times 2}(\R) \, \Big | \, A^2=A \Bigr \} \]
```

produces the output

$$\Big\{A \in M_{2\times 2}(\mathbb{R}) \,\Big|\, A^2=A\Big\}$$

Notice the use of "\," here. This command says "add a tiny bit more space". Without this, the math would look a bit too cramped around the "such that" bar in the set descriptor.

Math operators

Let's say you want to type $\sin(x)$. If you use "$sin(x)$" what comes out is

$$sin(x).$$

It's not terrible, but the font of the sin makes it look like math variables. It is better to have it come out as

$$\sin(x)$$

and to do this you type "$\sin(x)$". The command \sin is called a **math operator**, and there are lots of familiar ones you can use:

$$\sin,\ \cos,\ \tan,\ \log,\ \det,\ \sec,\ \csc,\ \lim,\ldots$$

Multi-line equations

For long sequences of equations that don't fit on one line, we want to spread them out over multiple lines of text and have some alignment. Here is an example:

$$\begin{aligned}(x+1)^2+(x+3)^2 &= x^2+2x+1+x^2+6x+9\\ &= 2x^2+8x+10\\ &= 2(x^2+4x+5).\end{aligned}$$

This was produced via the so-called "align" environment, using the following code:

```
\begin{align*}
(x+1)^2+(x+3)^2 &= x^2+2x+1 + x^2 +6x+9 \\
&= 2x^2+8x+10 \\
&=2(x^2+4x+5).
\end {align*}
```

The & symbols tell LaTeX where the alignment should occur, and the double-backslash symbols \\ tell it to advance to the next line.

Using text in displayed math:

Sometimes you want to typeset text inside of math mode, for example here:

$$x=\frac{-b\pm\sqrt{\Delta}}{2a}, \quad \text{where } \Delta=b^2-4ac.$$

If you use the code

```
\[ x=\frac{-b\pm \sqrt{\Delta}}{2a}, where \Delta=b^2-4ac \]
```

then what LaTeX produces is

$$x=\frac{-b\pm\sqrt{\Delta}}{2a}, where \Delta=b^2-4ac.$$

Notice that the font and spacing around the "where" is all wrong, because LaTeX thinks these are math symbols: it thinks you are trying to say the equivalent of $w\cdot h\cdot e\cdot r\cdot e$, with the letters being math variables. The way around this is to use the \text command, via

```
\[ x=\frac{-b\pm \sqrt{\Delta}}{2a}, \ \text{where}\ \Delta=b^2-4ac \]
```

Note that use of the manual space command "\ " (with a space following the backslash) to get the spacing correct.

It is important to know that inside the braces of \text{blah blah blah} you can do anything you want, including using math mode! This is sometimes the best way to do something. For example, to typeset

$$\{x \,|\, x \text{ is a multiple of 3 and a divisor of 300}\}$$

I used

```
\[   \{x\, |  \, \text{$x$ is a multiple of $3$ and a divisor of $300$}  \} \]
```

Special symbols

The symbols $, &, #, %, {, }, \ , ^, _ are all used by LaTeX for special purposes. So you have to work a little extra if you want to include them as text in your document. I used the commands

```
\$
\&
\#
\%
\{
\}
\textbackslash
\^{}
\underline{\  }
```

to produce the symbols in the first sentence.

Quotation marks

In typesetting there are left apostrophes like ‘ and right apostrophes like ’. These are two separate keys on computer keyboards, and LaTeX is just fine with them. For quotation marks like "this", notice that there is a difference between the left quotation mark and the right one. In LaTeX the way do to quotation marks is via double apostrophes, either left or right as appropriate. So to typeset "this" you use

‘‘this’ ’

Understand? LaTeX interprets the keyboard quotation mark as a right apostrophe, so if you use

"this"

then LaTeX outputs ”this”.

Colored text

LaTeX can produce colored text like this and also like this. You can even go crazy if you want. To produce colored text, you will need the line

\usepackage[usenames,dvipsnames,svgnames]{xcolor}

in the frontmatter of your .tex file (after the \documentclass command but before the \begin{document} command).

The text

{\color{red} this is in red} while {\color{blue} this is in blue}

will produce

this is in red while this is in blue.

Note that all of the colored text should be grouped within set braces { }. Otherwise, everything ends up in color.

A list of color names can be found here:

www.latextemplates.com/svgnames-colors

Miscellaneous math advice

(1) Intead of writing $x * y$ write $x \cdot y$. Here \cdot is the command for producing a multiplication dot, so I wrote “ $ x \cdot y$”.

(2) LaTeX uses a slightly larger font in displayed math mode, and sometimes fractions come out looking too big. Notice the difference between

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$$

and

$$e^x = 1 + x + \tfrac{x^2}{2!} + \tfrac{x^3}{3!} + \cdots$$

Either is fine, but it depends on what you like. You can use the \tfrac command to produce fractions that look like they are in regular “text” math mode rather than displayed math mode. For example, the commands

\frac{53}{42} and \tfrac{53}{42}

produce

$\frac{53}{42}$ and $\tfrac{53}{42}$.

(3) There are two ellipsis symbols, \cdots and \ldots. The difference is shown here:

$$1 + 2 + 3 + \cdots , \qquad \text{versus} \qquad a_0, a_1, a_2, \ldots$$

Lists

Lists come in different forms. A bulleted list looks like

- Milk
- Potatoes
- Strawberries and Oranges

and in LaTeX you do it like this:

```
\begin{itemize}
\item Milk
\item Potatoes
\item Strawberries and Oranges
\end{itemize}
```

Sometimes you want to number the items in a list, rather than just using bullets. And sometimes you want them labelled (a), (b), (c)... rather than (1), (2), (3)... For these kind of things we use the `enumerate` environment. For what I am about to describe you should have the command

```
\usepackage{enumerate}
```

in the frontmatter of your LaTeX file (somewhere in the first few lines of the file). I have included this in most sample LaTeX files I've given you this quarter.

To produce

(1) Milk

(2) Cherries

(3) Soap

you would use

```
\begin{enumerate}[(1)]
\item Milk
\item Cherries
```

```
\item Soap
\end{enumerate}
```

You could also do

```
\begin{enumerate}[(a)]
\item Milk
\item Cherries
\item Soap
\end{enumerate}
```

and you would get

(a) Milk

(b) Cherries

(c) Soap

or

```
\begin{enumerate}[i.]
\item Milk
\item Cherries
\item Soap
\end{enumerate}
```

to get

i. Milk

ii. Cherries

iii. Soap

You get the idea. The argument in the square brackets tells how the labels on the list items should be formatted.

The Not-So-Basics

Frontmatter

For LaTeX, a "documentclass" refers to a style file that sets up basic formatting and commands for your whole document. The most often-used documentclasses are

`article` `letter` `book` `report` `exam`.

In this course I have often used the `exam` documentclass for homework assignments because it automatically takes care of question numbering, multiple parts, and so forth. For exams it can even keep track of how many problems each point is worth. For your midterm you used the `article` documentclass, because it provides nice formatting commands for title, author, sections, subsections, and bibliography. The `\title` and `author` commands you used in your midterm come from the `article` package.

The first line in any LaTeX file has the form "\documentclass{????}" where the question marks are replaced by an appropriate documentclass. Sometimes you will see

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

and here the (optional) argument within the square brackets sets up the default fontsize for the document. The document you are reading right now was written in 12 point type, which as you can see it a little larger than the standard 10 point type.

After the `\documentclass` command, what comes next in a LaTeX document are `\usepackage` commands. People have written a ton of packages for LaTeX that do various things, and these commands load whatever packages you happen to want. For example, here is a list of some common packages together with what they do:

`xcolor`	the ability to do colored text
`amsmath`	extra math stuff from the American Mathematical Society
`amssymb`	more extra math stuff from the American Mathematical Society
`amsthm`	more extra math stuff from the American Mathematical Society
`graphicx`	basic graphics package (for the `picture` environment)
`enumerate`	provides nice list-building features

Finally, after the `\usepackage` commands some various `\newcommand` statements, which define commands that are very specific to your own tastes and working habits. For example, the command `\rightarrow` produces the symbol $\rightarrow$. I use this A LOT, and I don't like having to type "\rightarrow" every single time. So in my documents I include the command

```
\newcommand{\ra}{\rightarrow}
```

This has the effect that every time I type `\ra` in my document, LaTeX will translate that into `\rightarrow`.

Here's another example. Sometimes I want to write statements like

$$\text{area}(X) + \text{area}(Y) = \text{area}(Z).$$

You know by now that if you use

$$\$ \text{ area(X) + area(Y) = area(Z)}\$$$

then you get

$$area(X) + area(Y) = area(Z).$$

It's okay, but the spacing is a little off because LaTeX doesn't do text well in math mode. It is better here to define a new command like this:

```
\DeclareMathOperator{\area}{area}
```

Recall that Math Operators are things like `\lim` and `\sin`. Here we are defining our own math operator called `\area`. After putting this command in the frontmatter of our document, every time we write `$\area$` we will get "area"; for example, `$\area(X)$` gives area(X).

Numbered equations and how to cite them.

You know by now how to write an equation in displayed math format. Sometimes you want the equation to come with a label that you can refer to later, like this

$$E = mc^2 \tag{1}$$

Then later you can say, "Multiplying both sides of equation (**??**) by 17 gives..."

Equation numbers are created using the `equation` environment, which is just like the displayed math environment but adds an equation number. If you want to cite the equation later you have to use a `\label` command, and when you want to cite the equation number you use the `\ref` command. It looks something like this:

```
\begin{equation}
\label{Einstein}
E=mc^2
\end{equation}
...
Multiplying both sides of equation (\ref{Einstein}) by 17 gives...
```

Notice that the argument inside the `\label{???}` command is something you make up, and can be any label you want to use to later refer to this equation. Equation numbers can also be used with the `align` environment. We have previously learned to use the `align*` environment, which is the same thing but without the equation numbers.

You can also use labels inside Theorems, Remarks, Definitions, Sections, and so forth. Just put the `\label` command just inside your Theorem environment, or just after your `\section` command.