

A PRIMER ON USEFUL TOPICS IN \LaTeX

Tess Hick & Thomas Costin

CONTENTS

1. General Typesetting	3
1.1. Quotation Marks	3
1.2. Hyperlinks	3
1.3. Line Spacing	3
1.4. Hardcoding Characters	4
1.5. Lists and Sublists	4
2. Common Math	6
2.1. Tables	6
2.2. Matrices	6
2.3. Equations and Alignment	7
2.4. Piecewise Functions	8
2.5. Sums	8
3. Figures and Images	9
3.1. Inserting Images	9
3.2. Including Figures	10
3.3. Side-by-side figures	12
4. Text Alignment and Document Classes	14
4.1. Aligning text	14

1. GENERAL TYPESETTING

1.1. Quotation Marks. Do your quotation marks look like ”this,” with one side appearing to be backwards? Instead of directly typing them, use the package “dirtytalk.” To install this package, place the following code in the document preamble (next to all the other commands that start with “usepackage”):

```
\usepackage{dirtytalk}
```

Then to put text in quotation marks, simply type:

```
\say{I love Club Penguin.}
```

This gives the output “I love Club Penguin.” Now we have those beautiful correct-looking quotation marks!

1.2. Hyperlinks. Hyperlinks are another small detail that can be difficult in LaTeX since it reads common characters in URLs as commands, like `_`, `&` or `$`. So say we wanted to insert the link to Club Penguin. First, install the hyperref package by writing the following code in the document preamble:

```
\usepackage{hyperref}
```

Now, to insert a link, use the command

```
\url{https://play.cprewritten.net/}
```

This gives us the text: <https://play.cprewritten.net/> Notice how this not only cleans up any issues with reading characters in the url as code, but also adds a hyperlink directly into the document! Accessing Club Penguin directly from your overleaf document has never been easier!

1.3. Line Spacing. To change the spacing in an entire document, insert the following command into the preamble:

```
\renewcommand{\baselinestretch}{1}
```

The 1 in this command indicates single spacing. To double space, replace the 1 with a 2. This also works for decimals, like 1.5 spacing.

To change the amount of vertical space locally, use the command `\vspace{}` You can enter any measurement in the brackets. The most common unit to use is cm, but LaTeX also takes “in” to mean inches. Your number can be positive or negative, and include decimal places.

Finally, other quick spacing commands are:

```
\smallskip \medskip \bigskip
```

These look like the following, from smallest to largest:

I love to play Club Penguin!

My favorite game is Card-Jitsu.

I also like to Sled Race at the Ski Hill.

1.4. Hardcoding Characters. You might be wondering how we’re getting commands to show up as text instead of LaTeX reading them as code. You might also have run into situations where you want to write things like “I have one blue puffle & one red puffle” or “my Club Penguin membership costs \$4.95 per month,” but pesky LaTeX keeps reading & as an alignment character and \$ as math mode.

For single characters, there’s an easy fix. Simply put a backslash before the character, like this:

`\&` or `\$`

For longer phrases of code, use the “`verb`” command, meaning verbatim. You can look at the source code for many examples of this.

1.5. Lists and Sublists. There are many ways to make lists in LaTeX. The first way gives simple bullet points, and is called “`itemize`” where each line is called an “item.” Let’s say we want to list puffle colors. We’d use the following code:

```
\begin{itemize}
  \item Red
  \item Blue
  \item Rainbow
\end{itemize}
```

Which gives the following output:

- Red
- Blue
- Rainbow

For a numbered list, we do almost the exact same thing but with the “`enumerate`” command instead of “`itemize`”:

```
\begin{enumerate}
  \item Red
  \item Blue
  \item Rainbow
\end{enumerate}
```

Which gives the following output:

- (1) Red
- (2) Blue
- (3) Rainbow

It is also possible to nest lists in LaTeX by beginning the enumerate environment repeatedly. The code looks like this:

```
\begin{enumerate}
  \item First level item
  \item First level item
  \begin{enumerate}
    \item Second level item
    \item Second level item
    \begin{enumerate}
      \item Third level item
      \item Third level item
      \begin{enumerate}
        \item Fourth level item
        \item Fourth level item
      \end{enumerate}
    \end{enumerate}
  \end{enumerate}
\end{enumerate}
```

Which gives this nice nested list as its output:

- (1) First level item
- (2) First level item
 - (a) Second level item
 - (b) Second level item
 - (i) Third level item
 - (ii) Third level item
 - (A) Fourth level item
 - (B) Fourth level item

2. COMMON MATH

2.1. Tables. Perhaps we want to make a more comprehensive puffle guide. We might want to outline some of the traits each color possesses in a table. To do this, we use the tabular environment. This is the default tabular environment code and its output:

```
\begin{tabular}{c|c}
  item 1 & item 2 \\
  item 3 & item 4 \\
\end{tabular} \\
```

```

  item 1 | item 2
  item 3 | item 4
```

The `{c|c}` at the top means you have two columns separated by a line, with the text centered within the column. You can add additional columns and lines by adding more `c`'s or vertical lines. To add horizontal lines, use `\hline`. You can also remove the lines to create a column effect. You can see how this looks below in the matrices section. Finally, you can change the text alignment by changing `c` to `l` (left-aligned) or `r` (right-aligned). You can also replace `c` with a specified width for the column by writing `p{'width'}`. Our final puffle table might look like this:

Color	Trait
Blue	Loyal
Red	Adventurous
Rainbow	Farts sparkles

The code for this table is:

```
\begin{center}
\begin{tabular}{|c|c|}
\hline
  Color & Trait \\
\hline
  Blue & Loyal \\
\hline
  Red & Adventurous \\
\hline
  Rainbow & Farts sparkles \\
\hline
\end{tabular}
\end{center}
```

There are many more advanced things you can do with tables in LaTeX, but this will get you started!

2.2. Matrices. The generic code for a matrix is below. Entries in each row are separated by an ampersand, each row is ended by a double backslash, and the whole thing is in math mode ($...$ at the beginning and end).

```

 $\begin{matrix}$ 
  1 & 2 & 3 \\
  a & b & c
 $\end{matrix}$ 

```

This code gives a matrix with no brackets:

```

  1 2 3
  a b c

```

Changing the bracket type is easy, and common types are outlined below. Simply replace “matrix” in the code above with the phrase on the left to achieve the look on the right.

pmatrix $\begin{pmatrix} 1 & 2 & 3 \\ a & b & c \end{pmatrix}$

bmatrix $\begin{bmatrix} 1 & 2 & 3 \\ a & b & c \end{bmatrix}$

vmatrix $\begin{vmatrix} 1 & 2 & 3 \\ a & b & c \end{vmatrix}$

2.3. Equations and Alignment. In this section we’ll talk about how to separate equations from text. If you have only one equation or expression you want to emphasize, math display-style is a good way. To do this, simply put double dollar signs around your equation instead of single.

$$y = (x+6)(x-4)$$

This code gives the following:

$$y = (x + 6)(x - 4)$$

However, if you have multiple steps you want to show, such as expanding this expression, the align environment is the way to go. It allows you to line up a single symbol throughout multiple lines of code. The ampersand (&) is the alignment character here. Place it in the parts of your expression that you want to line up. In the example code below it’s aligning the equal signs. Note that the asterisk (*) after align stops LaTeX from numbering each line of the equation. Also, the align environment is automatically in math mode, so no dollar signs are required.

```

 $\begin{align*}$ 
  y &= (x+6)(x-4) \\
  &= x^2-4x+6x-24 \\
  &= x^2 +2x - 24
 $\end{align*}$ 

```

We then get this beautiful output:

$$\begin{aligned} y &= (x + 6)(x - 4) \\ &= x^2 - 4x + 6x - 24 \\ &= x^2 + 2x - 24 \end{aligned}$$

2.4. Piecewise Functions. Piecewise functions are notoriously difficult to typeset. However, there is an easy way! We'll use the “cases” environment of the `amsmath` package. First, put `\usepackage{amsmath}` in your preamble. Then, we can use the following code. Note that like in matrices, the ampersand (&) separates terms and the double backslashes end lines. Also, the cases environment needs to be in math mode (have dollar signs at the beginning and end).

```
\begin{cases}
  1 & x < 0 \\
  2x^2 & 0 \leq x \leq 100 \\
  3x & 100 < x
\end{cases}
```

We now have this beautiful piecewise function where the bracket size changes to fit your equations perfectly!

$$\begin{cases} 1 & x < 0 \\ 2x^2 & 0 \leq x \leq 100 \\ 3x & 100 < x \end{cases}$$

2.5. Sums. In L^AT_EX, sums can either be part of a line of text, or separated like in display style. To write a sum that is in line with the rest of the text, like this one, $\sum_{n=1}^{\infty} 2^{-n} = 1$, we use the following code:

```
\sum_{n=1}^{\infty} 2^{-n} = 1
```

To create a sum that's separated in display style, like this,

$$\sum_{n=1}^{\infty} 2^{-n} = 1,$$

you can use the exact same code with double dollar signs!

```
$$\sum_{n=1}^{\infty} 2^{-n} = 1$$
```


3. FIGURES AND IMAGES

3.1. Inserting Images. Inserting images into a L^AT_EX document is actually quite simple. To begin, make sure to have the `\usepackage{graphicx}` command in the preamble of your document. This will allow you to use commands like `\includegraphics[]{}`. For example, if we wanted to include a picture of the Club Penguin logo we could do so using the following commands

```
\begin{center}
  \includegraphics[scale=.15]{Club_Penguin_Logo.svg.png}
\end{center}
```

which would give the following output



Notice how the `[scale = .15]` command shrank the image down so that it would fit on the page. This is one of many different options that can be specified for included graphics. Other useful options are summarized in the table below (created using the `tabularx` package).

<code>width =</code>	width of the picture
<code>height =</code>	length of the picture
<code>angle =</code>	degrees of rotation
<code>keepaspectratio</code>	maintains original aspect ratio

It worth noting that you can specify multiple options at once when including an image. For example, if I wanted to both scale and rotate the club penguin logo I could do so using the following commands

```
\begin{center}
  \includegraphics[scale=.15,angle=180]{Club_Penguin_Logo.svg.png}
\end{center}
```

which would give the following output



One final thing worth mentioning is that the picture itself must be uploaded to overleaf before you try to insert it into your document. If you click horizontal arrow on the left of your screen and then the “upload” button you’ll be given the option to upload an image from your computer.

3.2. Including Figures. Figures are another way that visuals can be incorporated into a document. Begin by including the `\usepackage{caption}` command in the preamble of your document. To get started, lets include a very basic figure that contains an image of a penguin. Using the following commands

```
\begin{figure}[h]
  \centering
  \includegraphics[scale=0.15]{Penguins.jpg}
  \caption{Three happy penguins}
\end{figure}
```

we would get the following output



FIGURE 1. Three happy peguins

This is nice, but what if we want a way to reference the figure directly in the text? To do this, we would use the same set of commands as above but include `\label{fig: threepenguins}` as an additional line. Using the new set of commands

```
\begin{figure}[h]
  \centering
  \includegraphics[scale=0.15]{Penguins.jpg}
  \caption{Three happy penguins}
  \label{fig:threepenguins}
\end{figure}
```

which would give us an output that looks exactly the same as the previous figure.



FIGURE 2. Three happier penguins

However, we can now cite this figure directly in the text using the `\ref{fig:threepenguins}` command. For example, we might have a statement like “The penguin is the happiest animal on earth, as shown in figure 2”.

The square brackets after the `begin figure` command allow us to specify where on the page the image should appear. A summary of those commands and what they do is provided in the table below.

h	place the float <i>here</i> , approximately at the same spot it occurs in the source
t	position at the <i>top</i> of the page
b	position at the <i>bottom</i> of the page
p	Put on a special <i>page</i> for floats only
!	Override internal parameters determine “good” float positions
H	Places the float exactly in the location of the L ^A T _E Xcode

3.3. Side-by-side figures. While having just one figure is nice, there may be instances when we need to include multiple images within a single figure. To do this, we would use the following commands (filled in as appropriate)

```

\begin{figure}
\centering
\begin{subfigure}[b]{0.3\textwidth}
\centering
\includegraphics[width=\textwidth]{}
\caption{}
\label{}
\end{subfigure}
\hfill
\begin{subfigure}[b]{0.3\textwidth}
\centering
\includegraphics[width=\textwidth]{}
\caption{}
\label{}
\end{subfigure}
\hfill
\begin{subfigure}[b]{0.3\textwidth}
\centering
\includegraphics[width=\textwidth]{}
\caption{}
\label{}
\end{subfigure}
\caption{}
\label{}
\end{figure}

```

and end up with the following figure



(A) “Speedy penguins”



(B) “The stare down”



(C) “Love at first sight”

FIGURE 3. Famous works of penguin art

There are many other things that can be done with figures and images but the ideas presented above will give the new user a strong foothold.

4. TEXT ALIGNMENT AND DOCUMENT CLASSES

4.1. Aligning text. Aligning text can be one of the most infuriating exercises when using L^AT_EX. That being said, once you’ve read this guide I’m sure your blood pressure will return to normal levels. Beginning with the basics, we use the `\noindent` and `\indent` commands to remove or add indents, respectively. To center text we use the `\begin{center}` command, which will create something like the example below.

Example 1: “Club Penguin is honestly my favorite game of all time. I absolutely *love* how **centered** the game makes be feel. Mad props to Disney.”
- Thomas Costin

The center environment sure does make me feel centered, but odds are the average math student is more interested in right (or left) aligned environments. To align text to the right we use the `\begin{flushright}` command, which will create a paragraph that looks like the one below.

Example 2: Club Penguin was a massively multiplayer online game (MMO), involving a virtual world that contained a range of online games and activities. It was created by New Horizon Interactive (now known as Disney Canada Inc.). Players used cartoon penguin-avatars and played in an arctic-themed open-world. After beta-testing, Club Penguin was made available to the general public on October 24, 2005, and expanded into a large online community, such that by late 2007, it was claimed Club Penguin had over 30 million user accounts. In July 2013, Club Penguin had over 200 million registered user accounts.

Similarly, we could use the `\begin{flushleft}` command on the same paragraph to align the text to the left. Instead of using the commands discussed above, we could instead use the `\centering`, `\raggedright`, and `\raggedleft` commands, which will apply text alignment to all text beneath the command. As we usually do not want to apply text formatting to the entire document, we can use the `\begin{group}` and `\end{group}` commands. For example, the following are three examples of how we could use the `\centering`, `\raggedright`, and `\raggedleft` commands.

Example 3 Tis a crying shame that our glorious Disney Club Penguin has been so quickly snatched from us. May our thoughts and prayers lend the many penguins in that wonderful game the strength they need to survive their next journey.

Example 4 We love us some good ol left aligned text. Tis the most glorious type of text one could imagine. Back when I was a young lad we didn’t have any of this new fangled “centering” the kids these days are using! My stars how the world changes quickly nowadays.

Example 5 Four score and seven years ago our Penguins brought forth on this continent, a new club, conceived in Liberty, and dedicated to the proposition that all gamers are created equal.

Although these commands may seem simple at first glance they can be used in very powerful ways. Just remember, with great power comes great responsibility. Good luck out there my fellow penguins!