76-270 Project 3

Rishub Jain, Anna Gupta, Vasu Agrawal, Lisa Liu

January 29, 2018

Contents

1 What Is LaTeX

LaTeX is a way to prepare documents without having to worry about many of the annoyances that normally come with preparing documents. It is most often used for scientific publications and articles, and lends itself well to just about anything involving equations or computer generated output. LaTeX is not a word processor. Instead, LaTeX is used to specify the content of your document, without worrying about the formatting. The compiler (yes, like programming) is then able to turn your LaTeX specification into an actual document! See the example LaTeX specification (top) and the rendered output (bottom) below.

```
\begin{document}
    This is a simple \LaTeX document. \\
    You can do \textbf{all} kinds of \textit{things}.
\end{document}
```

This is a simple LaTeXdocument.
You can do **all** kinds of *things*.

2 Why Should You Use LaTeX?

LaTeX is an incredibly powerful tool for a wide variety of use cases. The language makes it easy to represent complex mathematical formulations, add bibliographies and references, and include symbols from different languages. LaTeX files are simply text files - this means that they can be integrated well into a version control system (e.g. git), and can be generated from computer programs (e.g. MATLAB). The language is even extensible so that any features that are missing can be added by the user. In addition to generating documents (e.g. technical report, homework, or publications), LaTeX has plugins which allow it to render slide decks, resumes, and more.

LaTeX offers a number of advantages over a standard word processing program (e.g. Microsoft Word). The ecosystem around LaTeX makes it easy to do online collaboration, add custom extensions, and work across platforms and on a number of devices. However, LaTeX isn't always the answer. There is a learning curve to perform advanced formatting features in LaTeX, many of which are often better accomplished in a word processing program. It is sometimes difficult to visualize the output of making certain changes to the LaTeX file, as compared to a WYSIWYG (What You See Is What You Get) program like most word processors. Still, these problems can be solved, and we hope that this document helps demonstrate that the advantages of using LaTeX far outweigh the drawbacks.

3 How Do I Start Using LaTeX?

As mentioned above, LaTeX files are simply text files. They can be edited from your favorite text editor, such as Vim, Sublime Text, or even Notepad. However, a number of special purpose tools exist in the LaTeX ecosystem which make working with LaTeX documents easier. Tools such as Texstudio can be installed locally and allow side by side visualization of the LaTeX file and generated output. Online tools such as Overleaf allow similar functionality without having to install anything. Online tools such as ShareLaTeX also make it possible to collaborate on the same LaTeX document with multiple people at the same time, similar to Google Docs. To those new to LaTeX, we recommend the use of Overleaf for its simple graphical interface and quick start up time. For advanced users, we recommend an offline text editor such as Vim, in combination with a version control system (Git) for backups.

4 Latex Template

There is often a lot of boilerplate associated with creating LaTeX documents. You'll often need to include all kinds of different packages, Following the LaTex philosophy in allowing you to focus on content, rather than formatting, we've provided a basic template which should help. This template provides sections for you to enter your name, class, assignment, etc, and then renders them on the first page, as well as in the header for all of the rest of the pages. This template is designed to help you with homework assignments, as the desired format for other forms of documents may be different. You should find the template in a "template.tex" file attached to this document. Open it in your favorite LaTeX editor (e.g. Overleaf), and start writing your content right before the last line. In fact, this tutorial was compiled with that very same template!

5 Mathematical expressions

One of the strongest reasons to use LaTeX is the flexible typesetting of mathematical expressions.

To write an inline math expression, you can use the $ delimiter before and after your expression. Example:

The expression $x^2 + \alpha = 8$ is done inline.

You can also add expressions on their own line using the \[\] delimiters on each side of your expression. Example:

```
The expression \[x^2+\alpha=8\] is done on its own line.
```

The expression

$$x^2 + \alpha = 8$$

is done on its own line.

You can have multi-line expressions that can be aligned vertically using the `\begin{align}` and `\end{align}` delimiters on each side. For example, you could have multiple rows of equations aligned so their equal signs are on the same vertical axis:

```
\begin{align*}
    \beta + x^2 &= y\\
    y_1 &= y_2 + y_5 \\
    A \times B &= C\\
\end{align*}
```

$$\begin{aligned} \beta + x^2 &= y \\ y_1 = y_2 &+ y_5 \\ A \times B &= C \end{aligned}$$

As shown above, you can easily type many different symbols

To use greek letters, you can use their names: `$\alpha$`, `$\beta$`, etc.

The integration symbol, summation symbol, and product symbol can be shown with `\int`, `\sum`, and `\prod` respectively:

```
    \[ \int_0^\infty f(x) dx = \sum_{i=1}^{100} y_i = \prod_{j=0}^2 z_j \]
```

$$\int_0^\infty f(x) dx = \sum_{i=1}^{100} y_i = \prod_{j=0}^2 z_j$$

Nearly any mathematical symbol can be represented in LaTeX. To find one, you can try Googling it, or drawing it on http://detexify.kirelabs.org/classify.html

Matrices can also be shown in LaTeX using the `\begin{bmatrix}` and `\end{bmatrix}`. Example:

```
\[A=
    \begin{bmatrix}
    0 & 1 & 4 & 5 & 0 \\
    8 & 9 & 1 & 2 & 3
    \end{bmatrix}
\]
```

$$A = \begin{bmatrix} 0 & 1 & 4 & 5 & 0 \\ 8 & 9 & 1 & 2 & 3 \end{bmatrix}$$

6 Lists

In LaTeX, you can have both bulleted list and enumerated list. For a bulleted list, you can use the `itemize` environment. For an enumerated list, you can use the `enumerate` environment.

Here is an example of different type of lists:

```
\begin{itemize}
\item This is a bulleted list
\item We can also have a enumerated list
    \begin{enumerate}
    \item This is a enumerated list
    \item They both use item for each new item in the list
    \item A enumerated list will automatically index items
    \end{enumerate}
\item We can have another enumerated list with different indices
    \begin{enumerate}[(i)]
    \item You can only do this if you have  \lstinline!\usepackage{enumerate}! in
    the document header, i.e. before the line \texttt{\textbackslash begin\{document\}}
        \begin{enumerate}[(a)]
        \item You can even have nested lists
        \item And more indexes
        \end{enumerate}
    \end{enumerate}
\end{itemize}
```

- This is a bulleted list
- We can also have a enumerated list
    1. This is a enumerated list
    2. They both use item for each new item in the list
    3. A enumerated list will automatically index items
- We can have another enumerated list with different indices
    - (i) You can only do this if you have `\usepackage{enumerate}` in the document header, i.e. before the line `\begin{document}`
        - (a) You can even have nested lists
        - (b) And more indexes

7 Images

LaTeXalso allows you to insert images into your documents. You need to make sure that the code `\usepackage{graphicx}` (Mind the Spelling!) is somewhere in the document header, i.e. above the line that says `\begin{document}`. Images can then be inserted using the `\includegraphics` command. The format is `\includegraphics[parameters]{filename}`. The only command `\includegraphics` requires is `filename`, which is the name of the image file which

- Excludes the extension, for example `’image’` instead of `’image.png’`
- Includes the path relative to the main .tex file if you are using the local compiler, for example, if `image.png` is stored in a directory called `dir`, you would enter `’dir/image’` instead of just `’image’`

The `parameters` denotes information telling LaTeXhow large you want the image to be. It is optional, and when missing, it will be scaled by the same amount. The simplest way to control the size of the image is to enter `[width=k\textwidth]`, where `k` is a scalling factor between 0 and 1.

8 New Command

If you want to be lazy, LaTeXalso allows you to define new command and use the new command. If you define your own new command, you can save a lot of time on repeated typing of a thing for math modes. You can also define new command on anything you want to be lazy about. For example, if you have to type a long word or sentence for many times, you can simply define a new command for it, and instead of typing the long word or sentence, you can just type the short command and be lazy. Defining new command can also help you avoid copy paste error and make life easy and neat.

To add your own command, use `\newcommand{name}[num]{definition}`. Basically, the command only requires two argumets: the `name` of the command and the `definition` of the command. The `num` in square bracket is optional and specifies the number of arguments the new command takes. If `num` is missing, then it defaults to 0 and no arguments are allowed.

Here is an example:

1. You can use new command to type up some random thing you are too lazy to type, no matter it's text or $math$123. And this is an example that doesn't take in any argument.

2. You can have two arguments: abc and def. The #1 tag represents the first argument, and #2 the second argument and so on.

3. You can also use the new command as many times as you want, with different arguments. You can have two arguments: 123 and 456.