Lab 12: LaTeX Practice

Contents

1 LaTeX

1.1 Headers and Templates ........................................ 1
1.2 LaTeXGuide ..................................................... 2
1.3 Your First Document ........................................... 2

2 More LaTeX ................................................................ 3

3 Creating Document Offline ...................................... 6

Objectives

By the end of this lab you will:

1. be able to compose simple LaTeX documents

1 LaTeX

LaTeX (pronounced “lay-teck” or “lah-teck”) is a typesetting system for creating high-quality documents. It is especially useful for creating technical documents, but Spike uses it for everything from writing a letter to his 90-year-old great Aunt to writing to the Providence municipality to debate a parking ticket.

Why should you learn LaTeX? It is the standard for typesetting documents in scientific fields, so there is a good chance you will have to use it again someday. This document (and all other CS 17/18 documents) is typeset in LaTeX!

Unlike Microsoft Word (but similar to HTML), LaTeX is not “what you see is what you get” (WYSIWYG). You type your document in a text editor using special syntax, and then LaTeX reads your document and produces a PDF.

1.1 Headers and Templates

Getting started with Latex can be a bit difficult, but in CS 17, we have provided a LaTeX template to help you out. The template is available at /course/cs0170/src/lab12/template.tex. When you’re writing up assignments in CS 17, you can always start from this template.

Task: Copy the template to your home directory using the cp command, whose syntax we remind you is: cp ⟨filenotomove⟩ ⟨targetlocation⟩.
If you enter `cp /course/cs0170/src/lab06/template.tex <current-assignment>`, for example, you’ll have a copy of the template in your current directory.

Open up the template you just copied and take a look at it. At the very top of this template, and all \LaTeX documents, there are headers, which tell \LaTeX how to process the document. As a bare minimum, you need a header of this form:

```
\documentclass{article}
```

Alternatives to `article` include `report`, `letter`, and `book`. In addition to this parameter, there are optional parameters, which are delimited by square brackets. For example, take a look at the `documentclass` of the template:

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

Here, `12pt` indicates the font size, and `letterpaper` is an alternative to, say, `a4paper` or `a5paper`.

Sometime after the document class is declared, the document begins, like this:

```
\begin{document}
```

This is the main section of the document, and where you will write the actual contents of the document.

**Note:** Before beginning your document, you can include commands to do other cool things, like create a title, include the date, etc. We have provided this syntax for you, so all you need to do is enter the assignment name and number in the appropriate part of the template, and you’ll be good to go.

We’ve also designed some useful things for you in the template: `\begin{theorem} ... \end{theorem}` will make a nicely displayed theorem for you, and after it, you can write `\begin{proof} ... \end{proof}` to give yourself someplace to write a proof, nicely terminated with a small box (called a “Halmos symbol,” believe it or not). While we won’t be using \LaTeX in this course, this template may be useful for future CS courses.

### 1.2 \LaTeX Guide

Even with our handy \LaTeX template, there are still plenty of obscure aspects of \LaTeX syntax that could be difficult to understand. To help with this, we strongly recommend visiting ShareLatex’s helpful documentation site, which can be found below:

[https://www.sharelatex.com/learn/](https://www.sharelatex.com/learn/)

Remember that online documentation is your friend, and if you are still stuck, don’t hesitate to ask a TA!
1.3 Your First Document

The easiest way to create a LaTeX document is by using an online tool called Overleaf. Overleaf is an online LaTeX editor with realtime preview and a nice user interface. To begin, go to [www.overleaf.com](http://www.overleaf.com) and make an account.

**Task:** Create a new blank project. Then, open up the default main.tex file and copy in the contents of the template.tex file.

Conveniently, Overleaf provides the option of automatically compiling your latex file every time you edit it. Alternatively you may choose to click the recompile button to compile the latex when you see fit.

**Task:** Click the down arrow next to the recompile button and select an option under Auto Compile.

**Task:** Use LaTeX to make a simple pdf document in your main.tex file. It should have (at least) two things in it:

1. The following sentence:

   Benjamin Franklin once said, “The game of chess is not merely an idle amusement. Several very valuable qualities of the mind, useful in the course of human life, are to be acquired or strengthened by it... For Life is a kind of Chess, in which we have often points to gain, and competitors or adversaries to contend with.”

   **Hint:** Here’s how to put something in quotes in LaTeX:
   ```latex
   ``This is the syntax that produces quotation marks''.
   ```
   The left delimiter is the tilde key twice, without shifting, and the right delimiter is the quotation mark key twice, also without shifting.

2. A list (just like this list!) of four of your favorite board games—two per partner.

   **Hint:** It may be a good idea to check out the ShareLatex documentation that was mentioned earlier, they have lots of information about LaTeX syntax, including lists!

You’ve reached a checkpoint! Please sign up to get a lab TA to review your work.

2 More LaTeX

Now, let’s continue to our new friend LaTeX. You can change the format of text using LaTeX commands with just a few easy commands. For example `\texttt{}` makes words look like this, and the `\emph{}` makes words emphasized. If you want to denote a mathematical symbol, like a plus sign, number, or expression, put a $ on either side.

A brief digression:

This second form is *strongly preferred*. The word “emph” is short for “emphasis”, and someone who designed the “article” document style decided that emphasized content in a scholarly article should be italic. If you want to write an article in which emphasized content is bold, all you need to do is given “emph” a new meaning, and everything that was formerly made italic for emphasis will
now be made bold for emphasis. Contrast this with what you’d have to do if you’d written your document in Microsoft Word, of if, in \LaTeX, you’d written \textit{something} every time you wanted something in italics. You might think that you could simply replace “textit” with “textsbf” everywhere . . . But what if you also made all units, like meters-per-second or kilogram-newtons, be formatted in italic. Then a global search and replace would make them all bold, which might be against the Journal’s standard.

For something short like a homework, using “texttt” works fine, but the beauty of \LaTeX is that it’s a \textit{programming language} and you can defined new functions in it. So if you wrote

\begin{verbatim}
\newcommand{\code}[1] { \texttt{#1}}
\end{verbatim}

it would mean “from now on, when you see \code{foo}, that actually means {\texttt{foo}}.”

Now you can use “code” to format code nicely, and if you ever decide that code should be bold-italic instead of “typewriter” font, you can change it, for every single piece of code, by just redefining the \code command!

End of digression.

If you get stumped, don’t forget to look at the documentation!

**Task:** Produce the following outputs using \LaTeX. **Note:** you can increment the bullet points by using

\begin{verbatim}
\begin{enumerate}
\end{enumerate}
\end{verbatim}

Furthermore, if you want to change how the points are enumerated (for instance, you want it in an alphabetical order), you can do something like the following:

\begin{verbatim}
\begin{enumerate}[	extit{a.}]
\end{enumerate}
\end{verbatim}

**Hint:** For the following outputs, if there are commands that need to take in more than one character, you can use {} to enclose the characters that use them.

**Hint:** The following 8 characters have special meaning in \LaTeX, and the first seven of them can usually be typeset by prepending a backslash.

& % $ # _ { } 

1. Checkmate
2. Checkmate
3. fold-right
4. $2^{5x+20}$
5. $x_{20}$
6. $x = \{3, -5\}$
7. $n \leq 5x$
8. \( y \neq 12 \)
9. \( |x^3| \)
10. \( \pm \sqrt{b^2 - 4ac} \)
11. \( \left( \frac{x + 1}{2} \right)^3 \)
12. \( f \in O(n \mapsto n) \)
13. \( g \in \Theta(n \mapsto 2^n) \)
14. \( h \in \Omega(n \mapsto n!) \)
15. 
   \[ f(n) = \begin{cases} 
   \sin(n) & \text{if } n \leq 0 \\
   f(n - 1) + \log_2(n) & \text{otherwise}
   \end{cases} \]
16. (a) a. i. \textit{Can} 
   ii. \textit{You} 
   b. \textit{Do} 
   c. \textit{This}

\textbf{Task:} Produce the following aligned equations using \LaTeX: 

\[ y = (35 + x)^3 + (x + 1)^2 \]  
(1)
\[ y = x^3 + 105x^2 + 3675x + 42875 + x^2 + 2x + 1 \]  
(2)
\[ y = x^3 + (105 + 1)x^2 + (3675 + 2)x + (42875 + 1) \]  
(3)
\[ y = x^3 + 106x^2 + 3677x + 42876 \]  
(4)

Here's how to make a table:

<table>
<thead>
<tr>
<th>cell1</th>
<th>cell2</th>
<th>cell3</th>
</tr>
</thead>
<tbody>
<tr>
<td>cell4</td>
<td>cell5</td>
<td>cell6</td>
</tr>
<tr>
<td>cell7</td>
<td>cell8</td>
<td>cell9</td>
</tr>
</tbody>
</table>

For this example, you would use this command:

\begin{center}
\begin{tabular}{ |c|c|c| } \hline
 cell1 & cell2 & cell3 \\ 
 cell4 & cell5 & cell6 \\ 
 cell7 & cell8 & cell9 \\ \hline
\end{tabular}
\end{center}
Notice that the columns and vertical lines in the table are defined on this line:
\begin{tabular}{ |c|c|c| }
And that the rows are defined by the number of lines ending with `\\' in the \texttt{tabular} environment.
Also note that the horizontal lines are created with the \texttt{\hline} command.

**Task:** Try to make this table!

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>6</td>
<td>9</td>
</tr>
</tbody>
</table>

**Task:** Produce the following proof using \LaTeX:  

*Proposition:* The sum of two odd integers is an even integer.

*Proof.* Let our two odd integers be expressed as $2k + 1$ and $2n + 1$, with $k, n \in \mathbb{Z}$. Adding these together, we have the following:

1. $2k + 1 + 2n + 1 = 2k + 2n + 2$ by combining like terms  
2. $= 2(k + n + 1)$ by factoring out a 2

Let $m = k + n + 1$, $m \in \mathbb{Z}$. We can rewrite the sum as $2m$, and because twice any integer is an even integer, we have proven our claim. \(\blacksquare\)

You’ve reached a checkpoint! Please sign up to get a lab TA to review your work.

## 3 Creating Document Offline

Lastly, let’s try making \LaTeX files with our old friend Atom! Using this tool for future assignments is completely optional; if you like making \LaTeX files using Overleaf, then go for it! This is just an alternate way which some may find easier when editing files offline.

**Task:** Create a file called `lab12.tex` in Atom by typing the following command into your terminal within your `lab12` directory:

```
atom lab12.tex
```

This `lab12.tex` file will be what \LaTeX uses to create your PDF.

**Task:** Copy and paste the contents of the `.tex` file you just created on Overleaf into your `lab12.tex` file. Then generate the `.pdf` file from the `lab06.tex` file by running the following command:

```
pdflatex lab12.tex
```

This command will build a PDF document which you can open using a PDF reader, such as Acrobat Reader or Evince, like this:
Now you know how to build a pdf file from a LaTeX file using command line! Conveniently, when you rebuild the PDF, bvince will automatically update it for you!

Once a lab TA signs off on your work, you’ve finished the lab! Congratulations! Before you leave, make sure both partners have access to the code you’ve just written.

Please let us know if you find any mistakes, inconsistencies, or confusing language in this or any other CS 17 document by filling out the anonymous feedback form: http://cs.brown.edu/courses/csci0170/feedback.