

L^AT_EX Code Talk Notes

Brady Griffith

October 03, 2018

1 Editors

L^AT_EX is not a specific program, but a markup language for preparing documents. Because of this, there are multiple ways to use it.

1.1 Dedicated L^AT_EX Editors

There are many, but notably TeXworks. Opening the program will open a blank L^AT_EX document. Just begin typing. A useful feature is that it shows the rendered PDF in another window.

1.2 Web Editors

There are also many of these, but www.overleaf.com is the most popular. You begin by creating a new project. The file will be prefilled with text. You can clear it to follow along.

It doesn't store the files on your computer, but it can be used on any device without installing anything. It also displays the rendered PDF to the side.

1.3 Text Editors and Terminal

And text editor can be used to create L^AT_EX files. The files are then rendered using a terminal command (Discussed Below). This is preferred setup for using L^AT_EX. I use the Emacs editor. The rendered PDFs can then be viewed as PDFs are normally.

2 Basic Outline

The fundamental structure for L^AT_EX has 2 parts.

Firstly, a format must be specified with

```
\documentclass{article}
```

The article class is almost always the best choice.

You must always begin and end the document section.

```

\begin{document}
  Some text goes here.
\end{document}

```

3 Compiling/Rendering

A \LaTeX file is used as instructions to create a PDF. The process of making the PDF is called compiling/rendering. The process of doing this depends on the editor used.

For TeXworks, you press the green play button at the top. This will generate a PDF and update the document display.

On www.overleaf.com you hit the Recompile button. This will generate a PDF and update the document display, and provide a button to download the PDF.

When using a text editor and a terminal you compile using a command. Go to the location of the \LaTeX file in the terminal and execute the command

```
pdflatex your-file.tex
```

“your-file” should be replaced with the actual file name. This will generate the PDF which can be viewed as normal.

4 Titles

To format a title, place the following above the document section.

```

\title{\LaTeX \ Example}
\author{Brady Griffith}
\date{October 18, 2017}

```

And at the beginning of the document section add

```

\maketitle
\newpage

```

5 Sections

Note that the star suppresses auto-numbering

```

\section*{Question 1}
\subsection*{Part A}

```

6 Math

The velocity of the particle is given by $v(t) = at$.

To find the position, velocity is integrated with respect to time.

$$x(t) = \int_{t_0}^{t_1} v(t) dt = \frac{a}{2} t^2 + x_0$$

7 Custom Headers and Footers

I prefer to have my name, the assignment, and the page number in the header of my homeworks. You do that by firstly importing the fancyhdr package

```
\usepackage{fancyhdr}
```

And then you define the format by having the following before the document section.

```
\pagestyle{fancy}
\fancyhead{}
\fancyfoot{}
\fancyhead[L]{Brady Griffith}
\fancyhead[C]{\LaTeX \ Example}
\fancyhead[R]{\thepage}
```

8 Tables

```
\subsection*{Part B}
The positions for 1s and 2s are given. \\
\begin{tabular}{l | l}
t (s) & x (m) \\ \hline
1 &  $\frac{1}{2}$  \\
2 & 2 \\
\end{tabular}
```

9 Graphics

In the header

```
\usepackage{graphicx}
```

In the body, where example is the image without an extension

```
\includegraphics[width=2in]{example}
```

10 Matrices

In the header

```
\usepackage{amsmath}
```

In the body

```
\section*{Question 3}
```

The following equations

```
$$ 2x + 3y $$
```

```
$$ 4x + 5y $$
```

are given by the following matrix

```
\[
```

```
\begin{bmatrix}
```

```
2 & 3 \\\
```

```
4 & 5
```

```
\end{bmatrix}
```

```
\]
```

11 Aligned Equations

Aligned equations are helpful way to show a lot of algebra. To use them in the heading

```
\usepackage{amsmath}
```

In the body an ampridant align together

```
\section*{Question 4}
```

```
\begin{align*}
```

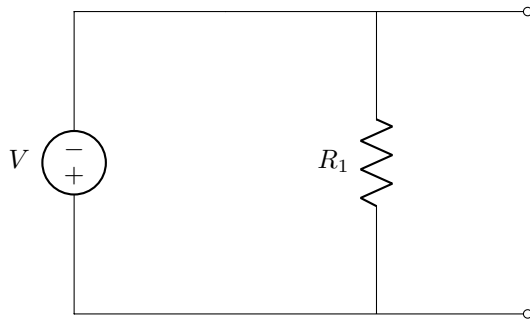
```
a &= 2x + 2y \\\
```

```
&= 2(x + y)
```

```
\end{align*}
```

12 Intro to Tikz and Circuits

Tikz is a procedural graphics tool built into L^AT_EX, and circuitikz is an implementation of it. There are many. Chess Patterns, Feynman Diagrams, and more. Google them and read their docs



This uses

```
\usepackage[american]{circuitikz}
```

In the body

```
\begin{circuitikz} \draw
(0,0) to[short, o-] (-6,0)
      to[american voltage source, l=$V$] (-6,4) — (-4,4)
      to[short, -o] (0,4)
      (-2,0) to[R, l=$R_1$] (-2,4)
;
\end{circuitikz}
```

13 Code Snippets

Code listings support many languages, including Mathematica, C, and Matlab.

This uses

```
\usepackage{listings}
```

In the body

```
\section*{Question 6}
\begin{lstlisting}[language=FORTRAN, caption=FORTRAN example]
PROGRAM EXAMPLE
  INTEGER :: N = 3
  N = N + 1
  PRINT *, N
END PROGRAM EXAMPLE
\end{lstlisting}
```

14 More Resources

Google is usually the best way to find resources, but when choosing links, I find that the best answers are on the following sites

- <https://www.sharelatex.com/learn>
- <https://en.wikibooks.org/wiki/LaTeX>
- <https://tex.stackexchange.com/> (Of course)