

MU \LaTeX Workshop

2nd May 2022



Why LaTeX and what is it?

Getting started

Basic Writing

Formatting Documents

Mathematics

Alignment and Arrays

Figures

Concluding Remarks

Why \LaTeX ? Why bother with it?

- ▶ \LaTeX is the gold standard for formatting maths. Many apps support LaTeX maths formatting (eg. Notion, Markdown, Discord, websites).
- + It is durable (doesn't corrupt like Word)
- + Separation of content (what you're writing) from style (how it looks)
 - + Automatically places figures/tables, etc
 - + You can swap out the styles easily
 - + The styles look great by default
- + Can easily separate out your document into manageable files (eg. for sections)
- = Really good for academia!
 - ↪ 90%+ of papers in engineering/maths/physics are written using LaTeX!

What is L^AT_EX?

- ▶ L^AT_EX vs Word
 - ▶ *LaTeX is a markup language* (ie. kinda like a coding language), that can be compiled to PDF by using a LaTeX engine.
“What you **mean** is what you get” (WYMIWYG) editor
 - ▶ *Word is a visual, GUI editor.*
“What you **see** is what you get” (WYSIWYG) editor
- ▶ Where to write LaTeX
 - ▶ Online (via. Overleaf)
 - ▶ Locally (using MikTeX, VScode, or other)
- ▶ Lay-tech?
 - ▶ Valid pronunciations include ‘Lay-tech’, ‘Lah-tech’, or ‘Lay-tex’. This [Stackoverflow article](#) explains it

When shouldn't I use \LaTeX ?

- ▶ For maths working heavy documents (if you want to be time efficient)
 - ▶ Maths working can take a while to set out in LaTeX, still faster to handwrite (unless you become a LaTeX god™)
- ▶ If you want absolute control over styling/positioning
 - ▶ Not as much control over styling out the box. Text interface makes it difficult to see final positioning.

Before we start!

If you forget everything in this workshop, remember this!

- ▶ LaTeX Wikibook (<https://en.wikibooks.org/wiki/LaTeX>)
 - ▶ Amazing resource for learning LaTeX!
- ▶ Detexify (<https://detexify.kirelabs.org/>)
 - ▶ Draw a symbol to find its LaTeX command!
- ▶ Various cheat sheets online!
(<http://tug.ctan.org/info/undergradmath/undergradmath.pdf>)

Why LaTeX and what is it?

Getting started

Basic Writing

Formatting Documents

Mathematics

Alignment and Arrays

Figures

Concluding Remarks

Getting started: Document Layout

Typical Document Set Up

```
\documentclass[a4paper,14pt]{article}
% Import any packages here!
\title{ Title }
\author{ Author }
\date{ Date }

\begin{document}

\maketitle

\section{Introduction}
Text Here!
\end{document}
```


Getting started: Reserved Characters

- ▶ Backslash `\` is the master key - distinguishes command from content. `\\` creates a new line.
- ▶ Curly brackets `{}` are used to contain and define commands
- ▶ Percent sign `%` is used for comments

Special character	Using special character as text
<code>%</code>	<code>\%</code>
<code>\$</code>	<code>\\$</code>
<code>{</code> or <code>}</code>	<code>\{</code> or <code>\}</code>
<code>&</code>	<code>\&</code>
<code>#</code>	<code>\#</code>
<code>_</code>	<code>_</code>
<code>^</code>	<code>\^{}{}</code>
<code>~</code>	<code>\~{}{}</code>

Getting started: Macros and Environments

- ▶ Commands are either macros or environments
- ▶ Macros
 - ▶ Macros begin with `\`. They may sometimes require *arguments* given using curly brackets `{}` (or optional arguments in square brackets `[]`)
- ▶ Environments
 - ▶ Surrounded by `\begin{environment}` and `\end{environment}`

Example 2

```
\[  
\left|  
\begin{array}{cc}  
2-\lambda & 0\\  
4 & 3-\lambda\\  
\end{array}  
\right|  
\]
```

$$\begin{vmatrix} 2-\lambda & 0 \\ 4 & 3-\lambda \end{vmatrix}$$

Getting started: Packages

Extend base LaTeX by using packages! Adds macros and environments to your document. Some of the most useful to use:

Package	Use
amsmath, amssymb, amsfonts	various maths tools
geometry	to finely tune page size + margins
pgfplot	to create figures and plots
hyperref	to embed links
graphicx	to add graphics and figures
siunitsx	for units in maths

Why LaTeX and what is it?

Getting started

Basic Writing

Formatting Documents

Mathematics

Alignment and Arrays

Figures

Concluding Remarks

Basic Writing: Text Options

We can change font options either using the

1. **command**

- ▶ Surround small blocks of text
- ▶ `\command{text}`

2. **switch command**

- ▶ Convert large blocks of text
- ▶ Use `\switchcommand`
- ▶ Will keep text option unless changed or;
- ▶ can surround text with `{\switchcommand text}`

Basic Writing: Text Options

Can change font size:

<code>\tiny</code>	Example
<code>\scriptsize</code>	Example
<code>\small</code>	Example
<code>\large</code>	Example
<code>\Large</code>	Example
<code>\LARGE</code>	Example
<code>\huge</code>	Example
<code>\Huge</code>	Example

Each can act as a switch command or normal command.

Basic Writing: Text Options

Can change font style:

<code>\textbf</code>	Example	Bold
<code>\textit</code>	<i>Example</i>	Italic
<code>\textsc</code>	EXAMPLE	Small Caps

Or font family:

<code>\textrm</code>	Example	Roman
<code>\texttt</code>	Example	Typewriter
<code>\textsf</code>	Example	Sans Serif

Whose switch commands can be evoked using `\rmfamily` etc.

Basic Writing: Text Options

Can change font colour:

Changing Text Colour

Here is how you can change the colour of single `\textcolor{red}{words}` using the command, or changing `{ \color{red}` blocks of texts by surrounding it in curly brackets `}` using the switch command.

Here is how you can change the colour of single **words** using the command, or changing **blocks of texts by surrounding it in curly brackets** using the switch command.

The package `xcolor` is particularly useful for getting very specific colours.

Basic Writing: Lists

- ▶ We can make ordered and unordered lists using the following environments:
 1. `\begin{enumerate}` for unordered lists
 2. `\begin{itemize}` for ordered lists
 3. `\begin{description}` for long paragraphs
- ▶ Each point is denoted using the `\item` command
- ▶ You can nest up to four lists
- :) You can change the bullet by specifying it as an argument in `\item[new bullet]`

Basic Writing: Lists

Example List

```
\begin{itemize}
  \item Heres an example of a list!
    \begin{enumerate}
      \item And how you can nest lists!
      \item Of any kind!
    \end{enumerate}
  \item[!] And how you can change bullet points
\end{itemize}
```

Why LaTeX and what is it?

Getting started

Basic Writing

Formatting Documents

Mathematics

Alignment and Arrays

Figures

Concluding Remarks

Formatting Documents: Title Pages

- ▶ We can make a title with `\maketitle` with the various options we give it
 - ▶ `\title{}`
 - ▶ `\author{}`
 - ▶ `\date{}`
- ▶ A page of contents with `\tableofcontents`
- ▶ An abstract using `\begin{abstract}` environment
- ▶ Further customisation using `\begin{titlepage}` environment

Formatting Documents: Sections

Hierarchy of \LaTeX is always important

1. `\section{}`
2. `\subsection{}`
3. `\subsubsection{}`

We can suppress numbering putting an asterix after the command, such as `\section*{}`

Formatting Documents: Spacing

- ▶ Kerning exists
- ▶ `\,, \:, \;; \!` and `\` (with a space) give increasing spaces between words
- ▶ `\quad` gives space equal to current font size (from the quadrat!)
- ▶ `\vspace{<length>}` and `\hspace{<length>}`
- ▶ `\\` ends the line and starts another
- ▶ `\newline` will insert a new line altogether
- ▶ `\newpage`

Why LaTeX and what is it?

Getting started

Basic Writing

Formatting Documents

Mathematics

Alignment and Arrays

Figures

Concluding Remarks

Mathematics: Environments

- Instead of writing in *text* mode, we write in *math* mode

	Inline	Block equation
TeX (outdated)	$\$ \$$	$\$ \$ \$ \$$
LaTeX	$\backslash (\backslash)$	$\backslash [\backslash]$
Environment	$\backslash \text{begin}\{\text{math}\}$	$\backslash \text{begin}\{\text{displaymath}\}$

- We can do multi-line equations using the `align` environment.
- $\backslash [\backslash]$ will not number equations, $\backslash \text{begin}\{\text{equation}\}$ will (we can suppress numbering using $\backslash \text{begin}\{\text{equation*}\}$, but might as well use `displaymath` at that point)

Mathematics: Environments

Inline and Oneline

An example of inline math to write

`\phi=2\sin(\theta)`. Or an equation like:

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

An example of inline math to write $\phi = 2 \sin(\theta)$. Or an equation like:

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

Mathematics: Environments

Multiline

And a multi-line math environment such as

```
\begin{align*}
0 &= x^2 - 1 \\
0 &= (x-1)(x+1)
\end{align*}
```

And a multi-line math environment such as

$$\begin{aligned} 0 &= x^2 - 1 \\ 0 &= (x - 1)(x + 1) \end{aligned}$$

Mathematics: Symbols

- ▶ Symbols are accessed via macros
- ▶ Greek symbols:
 - ▶ `\Omega` for uppercase Ω
 - ▶ `\omega` for lowercase ω
- ▶ [Cheat sheet!](#) You'll learn them eventually
- ▶ Some of my most common

\int_{-1}^1	<code>\int_{-1}^1</code>	$\sum_{n=1}^{\infty}$	<code>\sum_{n=1}^{\infty}</code>
$\frac{a}{b}$	<code>\frac{a}{b}</code>	\times	<code>\times</code>
∂	<code>\partial</code>	\cdots	<code>\cdots</code>
\pm	<code>\pm</code>	\leq	<code>\leq</code>

Mathematics: Superscripts and Subscripts

- ▶ we use `_` and `^`
- ▶ delimit its use with curly brackets `{ }` if more than one character is used

Multiline

We can add subscripts like x_i or x_{i+1} \\
Or superscripts like a^2 and a^{n+1} \\
Both can be used in conjunction like a_n^i \\
But this can be improved using curly brackets as well; $\{a_n\}^i$

We can add subscripts like x_i or x_{i+1}
Or superscripts like a^2 and a^{n+1}
Both can be used in conjunction like a_n^i
But this can be improved using curly brackets as well; a_n^i

Mathematics: Operators

- ▶ `\sin`, `\cos`, `\tan` use `\sin`, `\cos`, `\tan` in math mode
- ▶ `\frac{}{}` creates a fraction
- ▶ `\dfrac{}{}` will force a *display style* fraction
- ▶ `\sqrt[]{}{}` will create a square root by default, or *nth* root if specified in `[]`
- ▶ `\mathop{dx}` dx for upright differential operators if you care
- ▶ `\text{}` to write text in math mode

Various Operators

`$ \hat{a}, \dot{a}, \ddot{a}, \bar{a} $`

$\hat{a}, \dot{a}, \ddot{a}, \bar{a}$

Mathematics: Delimiters

Along with the standard delimiters found on the keyboard,

\langle	<code>\langle</code>
\rangle	<code>\rangle</code>
$\{$	<code>\{</code>
$\}$	<code>\}</code>
$\ $	<code>\mid</code>

Mathematics: Delimiters

- ▶ We can use *delimiters* to surround various expressions
- ▶ `\left<delimiter>` and `\right<delimiter>` to wrap using a delimiter
- ▶ Must have both commands to wrap
- ▶ use `\left.` if you do not want a left delimiter (or vice versa)

Delimiters

```
\[ f(x,y)=\left| \frac{x+y}{2} \right| \]
```

$$f(x,y) = \left\| \frac{x+y}{2} \right\|$$

Mathematics: Delimiters

- ▶ You can manually set the size of delimiters to distinguish nested equations

Delimiter Size

```
\[ \big( \Big( \bigg( \Bigg( \]
```



Four nested left parentheses of increasing size, demonstrating the effect of the `\big`, `\Big`, `\bigg`, and `\Bigg` commands.

Mathematics: Delimiters

Delimiter Size Example

```
\[ \Big( 2\times \big( f(x,y)+g(x,y) \big) \Big)^2 \]
```

$$\left(2 \times (f(x,y) + g(x,y))\right)^2$$

Why LaTeX and what is it?

Getting started

Basic Writing

Formatting Documents

Mathematics

Alignment and Arrays

Figures

Concluding Remarks

Alignment and Arrays: Align

- ▶ The master key for alignment is the `&`
- ▶ In the `\begin{align}` environment, we can use it to align equations
- ▶ We separate lines using the `\\` macro

Alignment

```
\begin{align*}
0 &= x^2+x\\
0 &= x(x+1)\\
&\implies x = 0, x=-1
\end{align*}
```

$$\begin{aligned} 0 &= x^2 + x \\ 0 &= x(x + 1) \\ \implies x &= 0, x = -1 \end{aligned}$$

Alignment and Arrays: Arrays

- ▶ `\begin{array}{}` command
- ▶ takes alignment argument (c,l,r)
- ▶ each cell is separated by the `&`
- ▶ each line is separated by the `\\`
- ▶ we can use delimiters to form varying matrices!
- ▶ Array writes in math mode

Alignment and Arrays: Arrays

Example 2

```
\[  
\left(  
\begin{array}{cc}  
2-\lambda & 0\\4 & 3-\lambda\\ \end{array}  
\right)  
\]
```

$$\begin{pmatrix} 2-\lambda & 0 \\ 4 & 3-\lambda \end{pmatrix}$$

Alignment and Arrays: Matrix environments

We can also use `pmatrix` and `bmatrix` environments:

Example 2

```
\[
\begin{pmatrix}
2-\lambda & 0\\
4 & 3-\lambda
\end{pmatrix}
\\
\begin{bmatrix}
2-\lambda & 0\\
4 & 3-\lambda
\end{bmatrix}
\\
```

$$\begin{pmatrix} 2-\lambda & 0 \\ 4 & 3-\lambda \end{pmatrix}$$

$$\begin{bmatrix} 2-\lambda & 0 \\ 4 & 3-\lambda \end{bmatrix}$$

Alignment and Arrays: Tables

- ▶ the text mode version is `\begin{tabular}{}{}`
- ▶ also takes an alignment argument
 - ▶ `l` → Left align
 - ▶ `c` → Centre align
 - ▶ `r` → Right align
- ▶ `\hline` to make horizontal breaks
- ▶ Extended table use
 - ▶ `\usepackage{booktabs}` → fancy rulings
 - ▶ `\usepackage{multicol}` and `\usepackage{multirow}` → Merged cells

Alignment and Arrays: Tables

Table example

```
\begin{tabular}{l|c|r} % /'s -> vertical lines
  \textbf{l align column} & \textbf{c align
column} & \textbf{r align column}\\
  \hline\hline
  1 & 2 & 3\\
  4 & 5 & 6\\
  \hline
\end{tabular}
```

l align column	c align column	r align column
1	2	3
4	5	6

Why LaTeX and what is it?

Getting started

Basic Writing

Formatting Documents

Mathematics

Alignment and Arrays

Figures

Concluding Remarks

Figures: Floats

- ▶ Floats cannot be broken across pages
- ▶ \LaTeX will place floats *for* you where it makes sense
- ▶ Do not try to move floats to exactly where you want them - \LaTeX knows best!
- ▶ `\begin{tabular}` environments can be put into a `\begin{table}` to float

Figures: Graphics

- ▶ we use `\usepackage{graphicx}` to import PNG, JPEG and PDFs
- ▶ `\includegraphics[parameters]{file path}`
 - ▶ `scale=`
 - ▶ `width=`
 - ▶ `height=`
 - ▶ `angle=`
- ▶ `\paperwidth`, `\textwidth`, `cm` and `in` are super useful
 - ▶ `width=0.8\textwidth`
 - ▶ `width=0.5\paperwidth`
 - ▶ `width=5cm`
- ▶ I would recommend putting your graphics into folders

Figures

- ▶ We can specify figures using the `\begin{figure}[]` environment
- ▶ Everything inside our environment will be floated
- ▶ Figures can take placement parameters
 - h Try to place exactly here
 - t Try to place on top of page
 - b Try to place at bottom of page
 - p Place on separate page only for floats
- ▶ You can put as many as you want, and \LaTeX will go through them one by one

Figures

- ▶ Commonly used with `\centering`
- ▶ `\caption{}`
- ▶ `\label{}` can be used to reference your figure

Default Figure

```
\begin{figure}  
  \centering  
  \includegraphics{  
  \caption{Caption}  
  \label{fig:my_label}  
\end{figure}
```

Figures: Referencing Figures

- ▶ We can use `\ref{label}` to call any figure or table
- ▶ \LaTeX will count figures and tables separately
- ▶ The same can be done with equations if you label them (they will also be counted separately)

Figures: Arranging Figures

- ▶ If you'd like to arrange figures in a specific way, you can use `\begin{minipage}{width}`
- ▶ minipages will align themselves next to each other if there is no whitespace
- ▶ perfect solution for arrays of figures



Figure: Frogs (2).

Figures: Arranging Figures

The source code for the previous figure:

Frogs Figure

```
\begin{figure}[h]
  \begin{minipage}{4cm}
    \includegraphics[width=4cm]{img/frog.jpg}
  \end{minipage}
  \begin{minipage}{4cm}
    \includegraphics[width=4cm]{img/frog.jpg}
  \end{minipage}
  \caption{Frogs (2).}
\end{figure}
```


Why LaTeX and what is it?

Getting started

Basic Writing

Formatting Documents

Mathematics

Alignment and Arrays

Figures

Concluding Remarks

Concluding Remarks: What Now?

- ▶ Practice!
- ▶ Explore new packages that may be of use to you
 - ▶ pgfplots
 - ▶ tikz
 - ▶ booktabs - lots more
 - ▶ mhchem - for chemical equations'
 - ▶ biblatex - referencing
- ▶ <https://tex.stackexchange.com/>

Concluding Remarks: What Now?

- ▶ Tinker with other built in commands and formatting tools
- ▶ Make your own commands and macros
 - ▶ `\newcommand{}[]{}`
 - ▶ `\newcommand{\R}{\mathbb{R}}`
 - ▶ `\newcommand{\deriv}[1]{\frac{d}{d#1}}`

Thank You!