
# SM-2302 Software for Mathematicians

## LaTeX3: Not Just Papers–Presentations & More

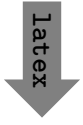
Dr. Nurul Huda Ramli
Mathematical Sciences, Faculty of Science, UBD
https://haziqj.ml

Semester I 2022/23

# LaTeX Recap

- You write your document in `plain text` with **`commands`** that describe its structure and meaning.
- The `latex` program processes your text and commands to produce a beautifully formatted document.

```
The rain in Spain falls \emph{mainly} on the plain.
```




The rain in Spain falls *mainly* on the plain.

# LaTeX Recap: Commands & Arguments

- A command starts with a *backslash* \ .
- Some commands take an *argument* in curly braces { } .
- Some commands also take *optional arguments* in square brackets [ ] .

```
\includegraphics[
  width=0.5\textwidth]{gerbil}

\includegraphics[
  width=0.3\textwidth,
  angle=270]{gerbil}
```

Image license: CC0

# LaTeX Recap: Environments

- The `\begin` and `\end` commands are used to create many different environments — contexts.
- The **`itemize`** and **`enumerate`** environments make lists.

```
\begin{itemize} % for bullet points
\item Biscuits
\item Tea
\end{itemize}

\begin{enumerate} % for numbers
\item Biscuits
\item Tea
\end{enumerate}
```

- Biscuits
- Tea

1. Biscuits
2. Tea

# LaTeX Recap: Mathematics

- The `equation` environment makes a numbered equation.

```
\begin{equation}
  \sum_{k=1}^{n} \frac{1}{2^k}
\end{equation}
```

$$\sum_{k=1}^{n} \frac{1}{2^k} \tag{1}$$

- Use dollar signs $ to mark mathematics in text.

```
% not so good:
Let a and b be distinct positive
integers, and let c = a - b + 1.

% much better:
Let $a$ and $b$ be distinct positive
integers, and let $c = a - b + 1$.
```

Let a and b be distinct positive integers, and let c = a - b + 1.

Let $a$ and $b$ be distinct positive integers, and let $c = a - b + 1$.

- Always use dollar signs in pairs — one to begin the mathematics, and one to end it.

In fact, we could have written `$...$` as `\begin{math}...\end{math}`.

# LaTeX Recap: Document Structure

- Starts with the `\documentclass` — what type of document.
- Metadata (`\title` and `\author`) and packages in the preamble.
- Content between `\begin{document}` and `\end{document}`.
- The `\maketitle` command creates the title; `\section` commands create numbered sections.

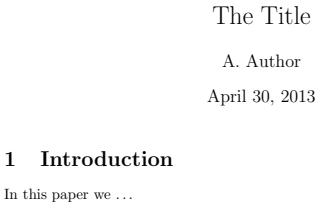
```
\documentclass{article}
% preamble
\title{The Title}
\author{A. Author}

\begin{document}
% body
\maketitle

\section{Introduction}

In this paper we \ldots

\end{document}
```



# LaTeX Recap: Exercise

1. Here is the text for a short article:[1]

   Click to open this exercise in **Overleaf**

2. Add LaTeX commands to the text to make it look like this one:

   Click to open the model document

## Hints

- Use the `enumerate` and `itemize` environments for lists.
- To typeset a % percent sign, *escape* it with a backslash (`\%`).
- To typeset the equation, use `\frac` for the fraction and the `\left(` and `\right)` commands for the parentheses.

[1] Based on `http://www.cgd.ucar.edu/cms/agu/scientific_talk.html`

LaTeX Recap

Presentations with `beamer`

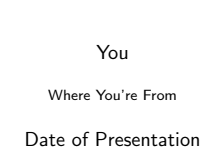
Drawings with TikZ

Notes with `todonotes`

Spreadsheets with `spreadtab`

# Presentations with `beamer`

- Beamer is a package for creating presentations (such as this one!) in LaTeX.
- It provides the **`beamer`** document class.
- Use the **`frame`** environment to create slides.



# Presentations with `beamer`: Following Along

- As we go through the following slides, try out the examples by typing them into the example document on **Overleaf**.

Click to open the example document in **Overleaf**

# Presentations with `beamer`: Frames

- Use `\frametitle` to give the frame a title.
- Then add content to the frame.
- The source for this frame looks like:

# Presentations with `beamer`: Sections

- You can use `\sections` to group your `frames`, and `beamer` will use them to create an automatic outline.
- To generate an outline, use the `\tableofcontents` command. Here's one for this presentation. The `currentsection` option highlights the current section.

```
\tableofcontents[currentsection]
```

# Presentations with `beamer`: Multiple Columns

- Use the **`columns`** and **`column`** environments to break the slide into columns.
- The argument for each **`column`** determines its width.
- See also the **`multicol`** package, which automatically breaks your content into columns.

```
\begin{columns}
  \begin{column}{0.4\textwidth}
    \begin{itemize}
    \item Use the columns ...
    \item The argument ...
    \item See also the ...
    \end{itemize}
  \end{column}
  \begin{column}{0.6\textwidth}
    % second column
  \end{column}
\end{columns}
```

# Presentations with `beamer`: Highlights

- Use `\emph` or `\alert` to highlight:

```
I should \emph{emphasise} that
this is an \alert{important} point.
```

I should *emphasise* that this is an important point.

- Or specify bold face or italics:

```
Text in \textbf{bold face}.
Text in \textit{italics}.
```

Text in **bold face**. Text in *italics*.

- Or specify a color (American spelling):

```
It \textcolor{red}{stops}
and \textcolor{green}{starts}.
```

It stops and starts.

- See `http://www.math.umbc.edu/~rouben/beamer/quickstart-Z-H-25.html` for more colors & custom colors.

# Presentations with `beamer`: Figures

- Use `\includegraphics` from the **`graphicx`** package.
- The **`figure`** environment centers by default, in **`beamer`**.

```
\begin{figure}
\includegraphics[
  width=0.5\textwidth]{gerbil}
\end{figure}
```

Image license: CC0

# Presentations with `beamer`: Tables

- Tables in LaTeX take some getting used to.
- Use the **`tabular`** environment from the **`tabularx`** package.
- The argument specifies column alignment — **l**eft, **r**ight, **r**ight.

```
\begin{tabular}{lrr}
Item   & Qty & Unit \$ \\
Widget & 1   & 199.99  \\
Gadget & 2   & 399.99  \\
Cable  & 3   & 19.99   \\
\end{tabular}
```

| Item | Qty | Unit $ |
|---|---|---|
| Widget | 1 | 199.99 |
| Gadget | 2 | 399.99 |
| Cable | 3 | 19.99 |

- It also specifies vertical lines; use **`\hline`** for horizontal lines.

```
\begin{tabular}{|l|r|r|} \hline
Item   & Qty & Unit \$ \\\hline
Widget & 1   & 199.99  \\
Gadget & 2   & 399.99  \\
Cable  & 3   & 19.99   \\\hline
\end{tabular}
```

| Item | Qty | Unit $ |
|---|---|---|
| Widget | 1 | 199.99 |
| Gadget | 2 | 399.99 |
| Cable | 3 | 19.99 |

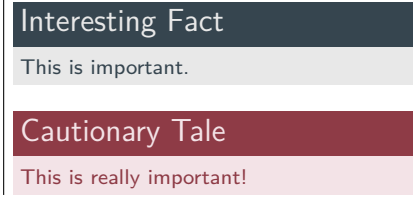
- Use an ampersand [&] to separate columns and a double backslash [\][\] to start a new row.

# Presentations with `beamer`: Blocks

- A **`block`** environment makes a titled box.

```
\begin{block}{Interesting Fact}
This is important.
\end{block}

\begin{alertblock}{Cautionary Tale}
This is really important!
\end{alertblock}
```




- How exactly they look depends on the theme...

# Presentations with `beamer`: Themes

- Customise the look of your presentation using themes.
- See `http://deic.uab.es/~iblanes/beamer_gallery/index_by_theme.html` for a large collection of themes.

# Presentations with `beamer`: Animation

- A frame can generate multiple slides.
- Use the **`\pause`** command to show only part of a slide.

```
\begin{itemize}
\item Can you feel the
\pause \item anticipation?
\end{itemize}
```

- Can you feel the

# Presentations with `beamer`: Animation

- A frame can generate multiple slides.
- Use the `\pause` command to show only part of a slide.

```
\begin{itemize}
\item Can you feel the
\pause \item anticipation?
\end{itemize}
```

- Can you feel the
- anticipation?

- There many more clever ways of making animations in **beamer**; see also the `\only`, `\alt`, and `\uncover` commands.

# Presentations with `beamer`: Exercise

Recreate Peter Norvig's excellent "Gettysburg Powerpoint Presentation" in **`beamer`**.[2]

1. Open this exercise in **Overleaf**:

   Click to open this exercise in **Overleaf**

2. Download this image to your computer and upload it to **Overleaf** via the files menu.

   Click to download image

3. Add LaTeX commands to the text to make it look like this one:

   Click to open the model document

[2] `http://norvig.com/Gettysburg`

LaTeX Recap

Presentations with `beamer`

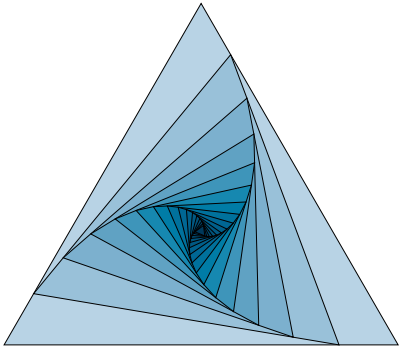
Drawings with Ti*k*Z

Notes with `todonotes`

Spreadsheets with `spreadtab`

# Drawings with Ti*k*Z

- Ti*k*Z is a package for drawing figures in LaTeX.
- It defines a powerful drawing language inside LaTeX. Short programs can draw surprisingly complicated things.

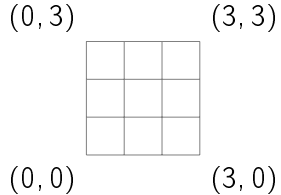


- We'll start with simple things. To draw a line in Ti*k*Z:

```
\begin{tikzpicture}
\draw (0,0) -- (1,1); % a line
```

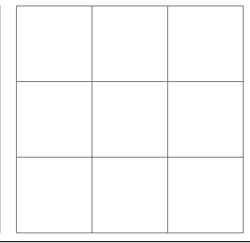
# Drawings with Ti*k*Z: Coordinates

- The default coordinates are centimeters, with the usual sense:




- It helps to draw a grid when you are working with Ti*k*Z:

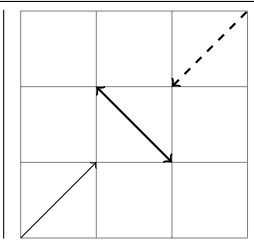
```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,3);
\end{tikzpicture}
```



# Drawings with Ti*k*Z: Lines

- Arrow heads and line styles are specified as options to the `\draw` command.
- End each draw command with a ; semicolon.

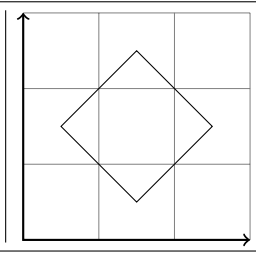
```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,3);
\draw[->] (0,0) -- (1,1);
\draw[<->, thick] (2,1) -- (1,2);
\draw[<-, thick, dashed] (2,2)--(3,3);
\end{tikzpicture}
```



# Drawings with Ti*k*Z: Paths

- You can specify multiple points to form a path.
- Arrows will appear only at the ends of the path.

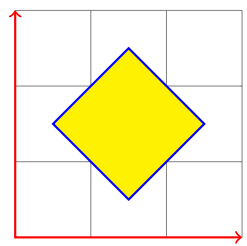
```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,3);
% axes:
\draw[<->, thick] (0,3)--(0,0)--(3,0);
% diamond:
\draw (1.5,0.5) -- (2.5,1.5) --
      (1.5,2.5) -- (0.5,1.5) --
      cycle; % close the path
\end{tikzpicture}
```



# Drawings with TikZ: Colours

- Colours are also specified as options to `\draw`.

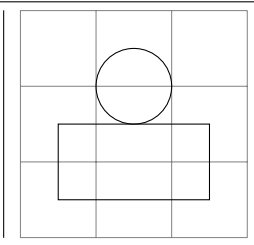
```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,3);
% axes
\draw[<->, thick, red]
  (0,3)--(0,0)--(3,0);
% diamond
\draw[thick, blue, fill=yellow]
  (1.5,0.5) -- (2.5,1.5) --
  (1.5,2.5) -- (0.5,1.5) --
  cycle;
\end{tikzpicture}
```



# Drawings with Ti*k*Z: Shapes

- Ti*k*Z has built-in commands for simple shapes.

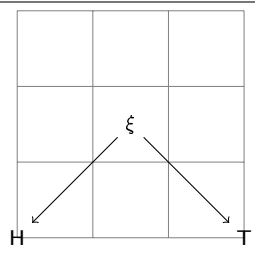
```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,3);
\draw (1.5,2.0) circle (0.5);
\draw (0.5,0.5) rectangle (2.5,1.5);
\end{tikzpicture}
```



# Drawings with Ti*k*Z: Nodes & Labels

- Use nodes to place text (and math) in Ti*k*Z drawings.
- You can also use nodes as coordinates — useful for diagrams.

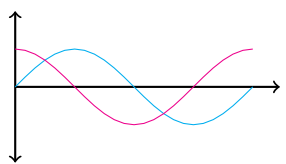
```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,3);
\node (h) at (0,0) {H};
\node (x) at (1.5,1.5) {$\xi$};
\node (t) at (3,0) {T};
\draw[->] (x) -- (h);
\draw[->] (x) -- (t);
\end{tikzpicture}
```



# Drawings with Ti*k*Z: Functions

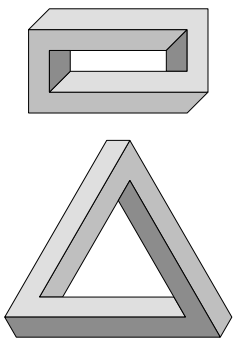
- You can even plot some simple functions.

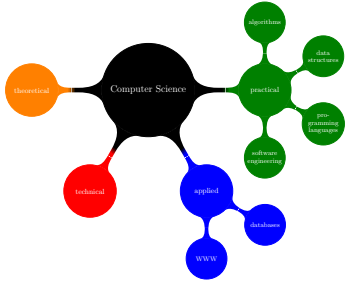
```
\begin{tikzpicture}[scale=0.5]
% y axis
\draw[<->, thick] (0,2) -- (0,-2);
% x axis
\draw[ ->, thick] (0,0) -- (7, 0);
% curves
\draw[cyan,domain=0:2*pi]
  plot (\x, {sin(\x r)});
\draw[magenta,domain=0:2*pi]
  plot (\x, {cos(\x r)});
\end{tikzpicture}
```



# Drawings with Ti*k*Z: Examples

- Check out TEXample.net for many Ti*k*Z examples:





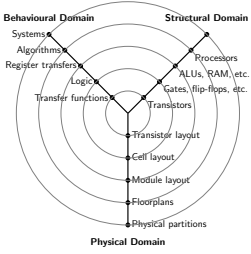
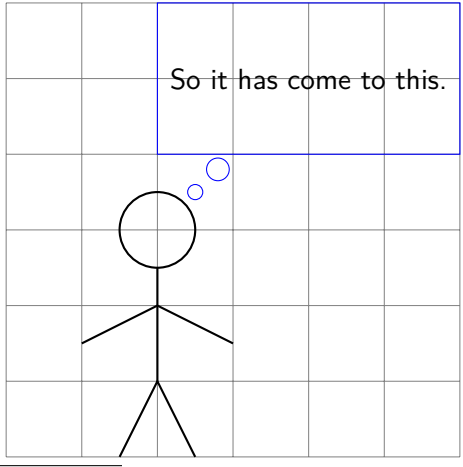




Figure 1: Gajski-Kuhn Y-chart

# Drawings with Ti*k*Z: Exercise

Draw this in Ti*k*Z:[3]




[3] Based on `http://xkcd.com/1022`

LaTeX Recap

Presentations with `beamer`

Drawings with TikZ

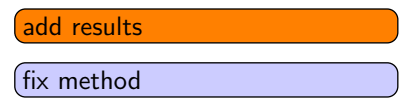
Notes with `todonotes`

Spreadsheets with `spreadtab`

# Notes with `todonotes`

- The `\todo` command from the **`todonotes`** package is great for leaving notes to yourself and your collaborators.

```
\todo{add results}
\todo[color=blue!20]{fix method}
```

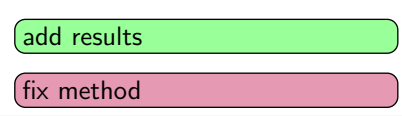



- Pro Tip: define your own commands with `\newcommand`

```
\newcommand{\alice}[1]{\todo[color=green!40]{#1}}
\newcommand{\bob}[1]{\todo[color=purple!40]{#1}}
```

This can save a lot of typing:

```
\alice{add results}
\bob{fix method}
```



# Notes with `todonotes`

- Only inline notes are supported with beamer, but margin notes are supported for normal documents.
- There is also a handy `\listoftodos` command.

Towards the Confusing Unification of Rasterization and Local-Area Networks in State Machines

Alice Bob, Carol David, Edward Fredrick

**Todo list**



**Abstract**


Rasterization and Smalltalk, while important in theory, have not until recently been considered important. Given the current status of wearable methodologies, analysts clearly desire the refinement of IPv4. Purr, our new heuristic for the producer-consumer problem [1], is the solution to all of these problems.


Are they polynomial time?

Realize multicast access points?

## 1 Introduction

Recent advances in certifiable symmetries and Bayesian technology synchro-

LaTeX Recap

Presentations with `beamer`

Drawings with TikZ

Notes with `todonotes`

Spreadsheets with **`spreadtab`**

# Spreadsheets with `spreadtab`

- Now that you've seen how LaTeX can replace Word and PowerPoint, what about Excel?
- Homework: try the **`spreadtab`** package!

Thanks, and happy TEXing!