A LaTeX Workshop

Sydney University Mathematics Society (ΣUMS)

March 22, 2016

Contents

Abstract

This is the document's abstract. It's an environment that automatically prints smaller font with wider margins.
In this document, we will explain some basic LaTeX syntaxes. If you follow along while reading the `.tex` file, you'll be able to find even more hidden comments!

Part I
The basic stuff

1 Structure of a .tex document

A `.tex` document can have two components:

1. Preliminary definitions are found before the `\begin{document}` command. This is often called the **preamble**.
2. The actual contents of the document is found between the lines:

   ```
   \begin{document}
   ...
   \end{document}
   ```

The simplest document you can produce is by typing:

```
\documentclass{article}
\begin{document}
Here is a line of text.
\end{document}
```

We will only go through the document classes "article" and "beamer" in this workshop. For the purposes of undergraduate mathematics assignments, "article" is probably the simplest document class to use. That being said, you should keep in mind LaTeX is flexible and has many more classes than just these two. You can even design your own classes!

2 Environments

You can think of environments as special regions in your `.tex` file that format your text differently. The abstract you saw in this document has a special styling because we used the **abstract** environment. Elsewhere we have inserted LaTeX syntaxes into this document without messing up the commands because we used the **verbatim** environment.

Another example of an environment is the **enumerate** environment, which produces numbered lists. You can also use the **itemize** environment to produce bullet-point lists.

1. This is the first item of an enumerate environment.

2. This is the second item of the same enumerate environment.

    (a) This is the first item of a new enumerate environment, which is nested inside the first enumerate environment.

        - This is the first item of an itemize environment, which is nested inside the first enumerate environment.

There are many more environments worth checking out – Google and the LaTeX wiki are your friends!

3 A comment about the preamble

In general, the preamble is where we can put in extra settings, packages and commands to use in conjunction with LaTeX.

A package is like an add-on to your LaTeX document; it allows extra symbols, environments or styles.

For example, to use a different style of the **enumerate** environment, you can uncomment the line `%\usepackage{enumerate}` by deleting the percentage sign in the preamble (or just type `\usepackage{enumerate}` somewhere in your preamble).

Now copy and paste the following block into your `.tex` file to see a different style of enumerate:

```
\begin{enumerate}[a.]
\item This is a different style of enumerate!
\item Instead of numbering, we are using letters!
\item That's what the \verb|[a.]| bit is doing!
\end{enumerate}
```

Or you can uncomment the line `%\newcommand{\RealNumbers}{\mathbb{R}}` in the preamble and type `\RealNumbers` into your `.tex` file to see a cool maths symbol!

This command is particularly useful if you have an expression that you want to use constantly in your documents but do not want to type out the full syntax every time. Here we are replacing the code snippet `\mathbb{R}` with the short-cut phrase `\RealNumbers`

You might think typing `\RealNumbers` isn't saving you much time in this case. Why not try changing it to `\Reals` or even `\R`? Just make sure you're not going to be using `\R` to mean anything else in your document...

Part II
The pretty stuff

4 Maths mode

Typing mathematical expressions in LaTeX requires using maths mode.

There are four basic maths modes to suit your typesetting needs. Most commonly, you will see mathematical symbols and short equations written between dollar signs: `$ equation goes here $` .

Inside each of these maths modes, you can use a range of syntaxes to typeset a range of symbols. Keep in mind that these syntaxes are almost universal in the science community, making it incredibly easy to communicate ideas to your friends and colleagues. Examples of appropriate syntaxes are can found here: https://wch.github.io/latexsheet/.

Most symbols have a very intuitive name for their syntax. For example, each letter of the Greek alphabet is written as a backslash followed by its conventional English spelling. Check out our favourite Greek letter:

- Lower-case sigma σ is just `$\sigma$`.
- The variant terminating sigma ς is given by `$\varsigma$`.
- And upper-case sigma Σ is `$\Sigma$`.

Let's look at an equation example. Typing:

```
\begin{equation}
\alpha \beta \gamma leq \frac{1}{abc} .
\end{equation}
```

will give you:

$$\alpha\beta\gamma \leq \frac{1}{abc}. \tag{1}$$

Before we start to typeset more complex equations, let's back-track a bit. The four basic maths modes are:

1. `$...$` for inline symbols like: $\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6}$.
2. `$$...$$` or `\[...\]` for equations that are important:

$$\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6}.$$

3. If your equation is super important, you can number it by using the **equation** environment:

```
\begin{equation}
...
\end{equation}
```

$$\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6}. \tag{2}$$

4. The **align** environment is used for multi-line **equation**s. If you include an asterisk, using the **align*** environment, none of the lines will be numbered.

```
\begin{align}
...
\end{align}
```

$$\begin{aligned} \zeta(2) &= \sum_{k=1}^{\infty} \frac{1}{k^2} \\ &= \frac{\pi^2}{6}. \end{aligned} \tag{3}$$

If you want to dynamically refer to an equation by its equation number, you can assign a name to the equation itself using `\label{name}`, and later reference it by inserting `\eqref{name}`. This is safer than statically referencing equations (by just typing "(2)", say), since you might change the order of equations in your document later.

For example: The Riemann-Zeta function evaluated at $s = 2$ is seen in Equation (2).

There are *many* mathematical symbols you can generate in LaTeX. All you need to do is get used to them! Don't forget that you can always redefine short-cuts for symbols or even strings of symbols in your preamble.

$$\oint \vec{E} \cdot d\vec{A} = \frac{q}{\epsilon_0} \tag{4}$$

$$\oint \vec{B} \cdot d\vec{s} = \mu_0 i + \frac{1}{c^2} \frac{\partial}{\partial t} \int \vec{E} \cdot d\vec{A} \tag{5}$$

$$\liminf_{n \to \infty} \underbrace{\frac{1}{n}}_{\in \mathbb{Q}} = 0 \tag{6}$$

4.1 Some tips

- Braces `{...}` are very important in maths mode. Whenever in doubt, stick a pair of braces around a connected part of an equation.

  Note for example that braces are required for exponentials and subscripts that are more than one character long. Notice that `$2^10$` yields 2^10, while we probably wanted `$2^{10}$` (which gives 2^{10}).

- Having `\usepackage{amssymb}` in the preamble is usually a good idea. Many symbols we use in mathematics are contained in this standard package. For example:

$$\mathbb{R} \tag{7}$$
$$\mathcal{N}(\mu, \sigma^2) \tag{8}$$
$$\mathfrak{A} \tag{9}$$

5 Matrices, tables and images

5.1 Matrices

Here are some examples showing how to typeset matrices. Remember matrices are in maths mode!

The ampersand symbols (`&`) separate out the elements in a row and the double backslashes (`\\`) end a row. This is just like how the **align** environment behaves.

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \tag{10}$$

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix} \tag{11}$$

You can, of course, just type the matrices out if you are familiar with the syntaxes. Otherwise, you could use tools to help you to construct them (e.g. https://www.codecogs.com/latex/eqneditor.php).

WARNING!!

As a general rule, if you generate a matrix in Mathematica (for example, using `IdentityMatrix[4]`, see below), then it is a good idea to **NOT** type this matrix out in LaTeX. There already exists an option to copy your results (not just matrices!) in LaTeX format! A similar output can be created in R using `library(xtable)`.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{12}$$

5.2 Tables

Creating a table is extremely similar to creating a matrix. As you might expect, there are many tools out there that can help you export results into a table syntax without having to manually prepare them.

1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Table 1: This is a table.

5.3 Images

Inserting pictures is easy. You just need to make sure the image file is in the same directory as the `.tex` file. Specifying the full name in the `\includegraphics{...}` command (from the `graphicx` package) will suffice. Though the **figure** environment will allow a nicer placement of picture.

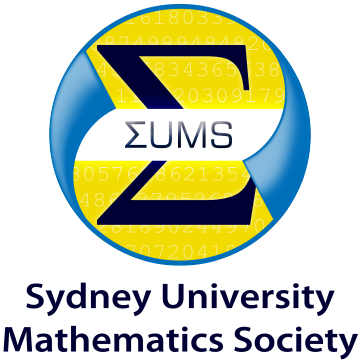

Figure 1: The picture was shrunk to 0.3 times the default line-width.

6 Bibliography

A bibliography may not be necessary for an undergraduate maths assignment, but nevertheless it is easy to create using LaTeX. There are two ways of doing this.

6.1 Manually typing the bibliography

One is to manually type in the following:

```
\begin{thebibliography}{9}

\bibitem{lamport94}
  Leslie Lamport,
  \emph{\LaTeX: a document preparation system},
  Addison Wesley, Massachusetts,
  2nd edition,
  1994.

\end{thebibliography}
```

There are many ways you can adjust the styling of your bibliography. See https://en.wikibooks.org/wiki/LaTeX/Bibliography_Management for more details.

6.2 BibTeX

Perhaps a great advantage of using LaTeX is the convenience of using associated programs like BibTeX. BibTeX is essentially a way for LaTeX users to manage their (often very large) number of references.

1. If you are using EndNote or Mendeley, then you can just add your references into your library and export them into a `.bib` file. Otherwise write the item in correct BibTeX syntax and save the file with the suffix `.bib`. For this workshop, we provided a file named "`ref.bib`", which contains only one item with the key "goossens93".

2. Once this is done, put the following lines at the end of your document:

   ```
   \bibliographystyle{plain}
   \bibliography{ref.bib}
   ```

3. Now you can simply cite any item inside this file. So we can cite the aforementioned item using `\cite{goossens93}` [1].

References

[1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1993.