Unix Tutorial **6: LaTeX**

Today, we are going to introduce the LaTeXsystem for document processing. This is pre-installed on the Glasgow Linux systems. However if you are running your own Linux distro, you will almost certainly need to install extra packages. On Ubuntu, try

```
$ sudo apt-get install texlive-full
```

or on Fedora systems,

```
$ sudo yum install texlive-scheme-full
```

You write a script that specifies the markup for the final document. Then you compile this script into a pdf to view the rendered document. We recommend that you use LaTeXfor reports in PSD3 and for your Honours dissertation in your final year project. So it's worth investing some time to learn the tool now . . .

Your first document

Below is the simple scaffolding—the minimal number of macros that you need for a simple document.

```
\documentclass{article}
\begin{document}
Hello world.
\end{document}
```

Create this file in vim, save it as firstdoc.tex. Now you can turn it into a pdf and view it:

```
$ pdflatex firstdoc.tex
$ evince firstdoc.pdf &
```

Note that the evince tool requires a Window manager—you need to be running with a GUI rather than a terminal. Other pdf viewing apps are available, such as Acrobat Reader, pdf.js (Mozilla) and Preview.app (Mac).

Now add some more complex markup to firstdoc.tex, for instance . . .

```
In \LaTeX, you may use \textit{italic} and \textbf{bold} face fonts.
It is also possible to have \texttt{typewriter} and
\textsf{sans−serif} fonts.
You may change the font size from {\tiny very small}
to {\Huge very large}.
```

You will also want to include `\section` directives to structure your document. You can use `\label` and `\ref` commands to add references to different sections as follows:

```
\section{Introduction}
\label{sec:intro}

Some intro text.

\section{Another Section}
\label{sec:nother}

Please refer back to Section \ref{sec:intro}.
```

Mathematical Typesetting

In case you hadn't guessed, all the Algorithmic Foundations tutorial sheets were produced using LaTeX. Indeed, formatting math. equations is one of LaTeX's strengths. For instance, try adding the following snippets to the body of your firstdoc.tex:

```
Below is an instance of a De Morgan rewrite rule:
\begin{equation}
\lnot(A \land B) \equiv \lnot A \lor \lnot B
\end{equation}
```

which should produce this equation:

$$\neg(A \land B) \equiv \neg A \lor \neg B \tag{1}$$

Check out the webpage at `https://en.wikibooks.org/wiki/LaTeX/Mathematics#List_of_Mathematical_Symbols` for more details of mathematical symbols in LaTeX. Note that you can use the $ notation for inline mathematical expressions, like $\exists a \in S.(a > 0)$ and so on.

More Exercises

0.1 Make a Makefile

Remember your Makefiles from yesterday? Write a Makefile to automate the build of the `firstdoc.pdf` file from LaTeX. It will be useful to add a `clean` target that removes all the intermediate files (.aux, .log, .out) generated by the pdflatex execution.

0.2 Recall your Maths

Remember all the set equivalences you learnt in Algorithmic Foundations? Typeset them in a LaTeXdocument. I'll give you one to get started: it's a distributivity rule . . .

$$A \cup (B \cap C) \equiv (A \cup B) \cap (A \cup C) \tag{2}$$

```
\begin{equation}
A \cup (B \cap C) \equiv (A \cup B) \cap (A \cup C)
\end{equation}
```

0.3 Timetabling

Check out the info at `www1.maths.leeds.ac.uk/latex/TableHelp1.pdf` to find out how to draw tables in LaTeX. Armed with this information, construct a lecture timetable for yourself using LaTeX. Each column can represent a day, each cell an hour for a lecture.

Further Reading

There are plenty of LaTeXresources online.

- tutorials e.g. at `http://www.latex-tutorial.com/tutorials/`
- collaborative editor at `http://sharelatex.com`
- cheat sheet of common commands at `http://users.dickinson.edu/~richesod/latex/latexcheatsheet.pdf`