A Whistle-Stop Tour of \LaTeX\ (Part 2)
Computing Science and Mathematics Skill Sharing

Alexander E. I. Brownlee
Nadarajen Veerapen
| 1. More table goodness |
| 2. Algorithms          |
| 3. Source code         |
| 4. Beamer Slides       |
| 5. Vector Graphics     |
| 6. Misc                |
More table goodness
The column specification can be altered using the \texttt{array} package. This is done in the argument of the \texttt{tabular} environment using >\{\texttt{\textbackslash command}\} for commands executed right before each column element and <\{\texttt{\textbackslash command}\} for commands to be executed right after each column element.

As an example: to get a column in math mode enter: \begin{tabular}{>{$}c<{$}}.

Another example is changing the font: \begin{tabular}{>{\tiny}c} to print the column in a tiny font.

1 \begin{tabular}{cc>{\tiny}c}c
2 Hello & Hello & Hello & Hello \ \ \\
3 I & I & I & I \ \ \\
4 am & am & am & am \ \ \\
5 a & a & a & a \ \ \\
6 table & table & table & table \ \ \\
7 \end{tabular}

See \url{https://en.wikibooks.org/wiki/LaTeX/Tables#Column_specification_using_.3E.7B.5Ccmd.7D_and_.3C.7B.5Ccmd.7D}
Use `siunitx` to round and align decimals in tables. (this package will also do loads of other stuff with units, not covered here)

```latex
\usepackage{siunitx}
\sisetup{round-mode=places} \% can also use "figures" for sig. figs.
{Density} & {Number of aircraft} & {QPPTW} & {Buf-QPPTW} & {Fuzzy-QPPTW} \\
0.8 & 44476 & 43.1 & 3.754500 & 736.5383 \\
0.9 & 518 & 3.363167 & 4.037117 & 634.6640 \\
1.0 & 578 & 3.485500 & 4.901850 & 743.5543 \\
\end{tabular}
```

<table>
<thead>
<tr>
<th>Density</th>
<th>Number of aircraft</th>
<th>QPPTW</th>
<th>Buf-QPPTW</th>
<th>Fuzzy-QPPTW</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.8</td>
<td>44476</td>
<td>43.1</td>
<td>3.75</td>
<td>737</td>
</tr>
<tr>
<td>0.9</td>
<td>518</td>
<td>3.36</td>
<td>4.04</td>
<td>635</td>
</tr>
<tr>
<td>1.0</td>
<td>578</td>
<td>3.49</td>
<td>4.90</td>
<td>744</td>
</tr>
</tbody>
</table>
Algorithms
Various packages:

\texttt{algorithmic, algorithm2e, algorithmicx}

...incompatible with each other!
\usepackage{algorithm,algpseudocode} \newcommand{Euclid's algorithm}{Euclid's algorithm} \begin{algorithm}[H]
\caption{Euclid's algorithm} \label{alg:euclid}
\begin{algorithmic}
\Procedure{Euclid}{$a,b$} \Comment{The g.c.d. of a and b}
\State $r \gets a \bmod b$
\While{$r \not= 0$} \Comment{We have the answer if r is 0}
\State $a \gets b$
\State $b \gets r$
\State $r \gets a \bmod b$
\EndWhile \label{euclidendwhile}
\State \textbf{return} $b$ \Comment{The gcd is b}
\EndProcedure
\end{algorithmic}
\end{algorithm}
Algorithm 1 Euclid’s algorithm

1: procedure Euclid\( (a, b) \)
2: \hspace{1em} \text{r} \leftarrow a \mod b
3: \hspace{1em} \textbf{while} \ r \neq 0 \ \textbf{do}
4: \hspace{2em} a \leftarrow b
5: \hspace{2em} b \leftarrow r
6: \hspace{2em} r \leftarrow a \mod b
7: \hspace{1em} \textbf{end while}
8: \hspace{1em} \textbf{return} \ b
9: \hspace{1em} \textbf{end procedure}

\( \triangleright \) The g.c.d. of a and b
\( \triangleright \) We have the answer if r is 0
\( \triangleright \) The gcd is b
Source code
The listings package: www.ctan.org/pkg/listings

\usepackage{listings}

\begin{lstlisting}[caption={Some source, showing an XML/KML \lstinline|way| element.},label={lst:osm-xml-example},float,floatplacement=H,language=xml]
<way id="4232478" visible="true" ... >
  <nd ref="25256057"/>
  <tag k="aeroway" v="taxiway"/>
  <tag k="width" v="23"/>
</way>
\end{lstlisting}

Listing 1: Some source, showing an XML/KML way element.

<way id="4232478" visible="true" ... >
  <nd ref="25256057"/>
  <tag k="aeroway" v="taxiway"/>
  <tag k="width" v="23"/>
</way>
You can add a `\lstinline|code|` snippet

You can add a `code` snippet

- note the unusual delimiters! They can be almost anything. Syntax is

```latex
\lstinline[<key=value list>]<character><source code><same character>
```

so `\lstinline!var i:integer;!` is possible.

Import source `\lstinputlisting{source_filename.py}`...

Also possible to use colourful syntax highlighting. See

http://texblog.org/2011/06/11/latex-syntax-highlighting-examples/
Configure in the preamble:

```latex
\lstset{
    language=XML,
    basicstyle=\small\ttfamily, % font
    keywordstyle=\color{blue},
    stringstyle=\color{red},
    commentstyle=\color{green},
    morecomment=[l][\color{magenta}]{\#}
    numbers=left, % line numbers
    frame=tb, % default float placement
    columns=fullflexible, % char width / col alignment
    captionpos=b,
    showstringspaces=false,
    morekeywords={node,way,tag,lat,lon} % add to language
}
```

```latex
\lstdefinestyle{latex}{
  language=\LaTeX TeX,
  basicstyle=\small \ttfamily,
  keywordstyle=\color{blueaccent},
  columns=fullflexible,
  showstringspaces=false,
  breaklines=true,
  numbers=left,
  morekeywords= subsection,toprule,cmidrule,midrule,bottomrule,subfloat,
  graphicspath,color,eqref,mathbb,text,subtitle,institute,inst,usetheme,
  useoutertheme,tableofcontents,pause},
```

Beamer Slides
\documentclass{beamer}
\begin{document}
\begin{frame}
  \frametitle{Frame Title}
  \framesubtitle{Frame Subtitle}
  \%content here
\end{frame}
\end{document}
\documentclass{beamer}
\title[Better Wash Viewing]{Enhancing the User's Observation of the Wash Process}
\subtitle{Far better than watching paint dry}
\author{Alfa, Bravo\inst{1} \and B. Bravo\inst{2}}
\institute{Fernglas Uni, F. Uni of Lavatrice
\inst{1} \Zeiss Institute of Optics\Fernglas University \and \inst{2} \Zanussi Institute of Physics\Free University of Lavatrice}
\date{CFO 2017\Conference on Fluids and Optics, 2017}
\usetheme{Warsaw}
\begin{document}
\frame{\titlepage}

Enhancing the User’s Observation of the Wash Process
Far better than watching paint dry

A. Alfa\inst{1} B. Bravo\inst{2}

1Zeiss Institute of Optics
Fernglas University

2Zanussi Institute of Physics
Free University of Lavatrice

Conference on Fluids and Optics, 2017
\usepackage{Warsaw}
\useoutertheme{infolines}
\begin{document}
...
\section{First Section}
\begin{frame}{Table of Contents}
\tableofcontents[currentsection]
\end{frame}
\section{Second Section}
...
\section{Third Section}
\end{document}
This presentation uses the custom Metropolis theme.

https://github.com/matze/mtheme

1 \documentclass[aspectratio=1610]{beamer}
2 \usetheme{metropolis}
\begin{block}{This is a Block}
  This is important information
\end{block}

\begin{alertblock}{This is an Alert block}
  This is an important alert
\end{alertblock}

\begin{exampleblock}{This is an Example block}
  This is an example
\end{exampleblock}
\begin{columns}
\begin{column}{0.6\textwidth}
  \textbf{%something here}
\end{column}
\begin{column}{0.4\textwidth}
  \textbf{%something there}
\end{column}
\end{columns}
Something here
\pause
Some more stuff appears
\pause
Surprise surprise
Something here
\pause
Some more stuff appears
\pause
Surprise surprise

Something here
Some more stuff appears
Something here
\pause
Some more stuff appears
\pause
Surprise surprise
\begin{itemize}
\item Always here
\item Appears second
\item Appears on the third slide and then disappears
\item Stays for two slides
\item This one becomes \alert<6>{important} at the end
\end{itemize}
\begin{itemize}
\item Always here
\item Appears second
\item Appears on the third slide and then disappears
\item Stays for two slides
\item This one becomes \alert<6>{important} at the end
\end{itemize}
\begin{itemize}
\item Always here
\item Appears second
\item Appears on the third slide and then disappears
\item Stays for two slides
\item This one becomes \alert<6>{important} at the end
\end{itemize}
\begin{itemize}
\item Always here
\item Appears second
\item Appears on the third slide and then disappears
\item Stays for two slides
\item This one becomes \texttt{\textbackslash alert}<6>{important} at the end
\end{itemize}

- Always here
- Appears second
- Stays for two slides
- This one becomes important at the end
\begin{itemize}
\item Always here
\item Appears second
\item Appears on the third slide and then disappears
\item Stays for two slides
\item This one becomes \texttt{\alert<6>}{important} at the end
\end{itemize}
\begin{itemize}
\item Always here
\item Appears second
\item Appears on the third slide and then disappears
\item Stays for two slides
\item This one becomes \alert{important} at the end
\end{itemize}
\begin{itemize}[<+->]
\item Simple increment
\item Simple increment
\item Simple increment
\item Simple increment
\end{itemize}
\begin{itemize}[<+->]
\item Simple increment
\item Simple increment
\item Simple increment
\item Simple increment
\item Simple increment
\end{itemize}
\begin{itemize}[<+->]
\item Simple increment
\item Simple increment
\item Simple increment
\item Simple increment
\item Simple increment
\end{itemize}
### Useful commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>\textbf{}</code></td>
<td>controls when to bold text</td>
</tr>
<tr>
<td><code>\textit{}</code></td>
<td>controls when to italicize text</td>
</tr>
<tr>
<td><code>\color{}</code></td>
<td>controls when to change colour of text</td>
</tr>
<tr>
<td><code>\alert{}</code></td>
<td>controls when to highlight text (theme-dependent colour)</td>
</tr>
<tr>
<td><code>\only{}</code></td>
<td>controls when to reveal text, occupies NO space otherwise</td>
</tr>
<tr>
<td><code>\uncover{}</code></td>
<td>controls when to reveal text, DOES occupy space otherwise</td>
</tr>
<tr>
<td><code>\alt{}</code></td>
<td>reveals first argument when specification is true, otherwise reveals second argument</td>
</tr>
</tbody>
</table>
Also works with environments

1 \begin{theorem}<1->[Pythagoras]
2 \[ a^2 + b^2 = c^2 \]
3 \end{theorem}

4 \begin{corollary}<3->
5 \[ x + y = y + x \]
6 \end{corollary}

7 \begin{proof}<2->
8 \[ \omega + \phi = \epsilon \]
9 \end{proof}

10 \begin{onlyenv}<3->
11 \text{some stuff}
12 \end{onlyenv}
Also works with environments

\begin{theorem}<1->[Pythagoras]
\[ a^2 + b^2 = c^2 \]
\end{theorem}

\begin{corollary}<3->
\[ x + y = y + x \]
\end{corollary}

\begin{proof}<2->
\[ \omega + \phi = \epsilon \]
\end{proof}

\begin{onlyenv}<3->
some stuff
\end{onlyenv}
Beamer Overlays (5)

Also works with environments

\begin{theorem}<1>[Pythagoras]
\[ a^2 + b^2 = c^2 \]
\end{theorem}

\begin{corollary}<3>
\[ x + y = y + x \]
\end{corollary}

\begin{proof}<2>
\[ \omega + \phi = \epsilon \]
\end{proof}

\begin{onlyenv}<3>
some stuff
\end{onlyenv}

Theorem (Pythagoras)
\[ a^2 + b^2 = c^2 \]

Corollary
\[ x + y = y + x \]

Proof.
\[ \omega + \phi = \epsilon \]

some stuff
Beamer Overlays (6)

Flatten overlays, usually when printing.

1 \documentclass[handout,notes=show]{beamer}
2 ...
4
5 %keep these two pictures on separate slides
6 \only<1| handout:1>{\includegraphics{pic1.eps}}
7 \only<2| handout:2>{\includegraphics{pic2.eps}}
8 ...
10
11 %hide a frame in handout mode
12 \begin{frame}<handout:0>
13 ...
15
16 %some notes
17 \begin{frame}
18 \end{frame}
19 \note{I need to remember to say this.}
If you wish to use a `verbatim` environment in a frame, you have to add the option `[fragile]` to the `{frame}` environment. The `\end{frame}` must be alone on a single line.

```latex
\begin{frame}[fragile]{Something important}
...
\end{frame}
```
Backup slides: `appendixnumberbeamer` package, calling `\appendix` will turn off slide numbering and progress bars for slides in the appendix.
Vector Graphics
Two main options for “writing” vector graphics:

- **pstricks**
  - Needs to be compiled to PostScript
- **PGF/TikZ**
  - PGF is a lower-level language, while TikZ is a set of higher-level macros that use PGF
  - Same (original) developer as Beamer (tight integration between the two)
Seismic focal mechanism and Pression-Tension axis.

http://www.texample.net/tikz/examples/seismic-focal-mechanism-in-3d-view/
\begin{tikzpicture}
    \draw [help lines, dashed] (0,0) grid (4,2);
    \draw [<->] (0,2) -- (0,0) -- (4,0);
    \draw [thick] (0,1.5) -- (3,0);
    \draw [ultra thick] (0,0) -- (2,2);
\end{tikzpicture}
```
\usepackage{tikz}
...
\begin{tikzpicture}
\draw [help lines, dashed] (0,0) grid (4,2);
\draw [<->] (0,2) -- (0,0) -- (4,0);
\draw [thick] (0,1.5) -- (3,0);
\draw [ultra thick] (0,0) -- (2,2);
\end{tikzpicture}

\tikz \fill [even odd rule] (0,0) circle (1) (1,0) circle (1);
```
\[ y_{ijk} = \mu + r_i + c_j + t_k + \epsilon_{ijk} \]
\[ y_{ijk} = \mu + r_i + c_j + t_k + \epsilon_{ijk} \]
Overall mean

$y_{ijk} = \mu + r_i + c_j + t_k + \epsilon_{ijk}$

Effect of row $i$

Effect of column $j$

Effect of treatment $k$
TikZ/Beamer integration (1)

\[ y_{ijk} = \mu + r_i + c_j + t_k + \epsilon_{ijk} \]  (1)

- Overall mean
- Effect of row \( i \)
- Effect of column \( j \)
\begin{align*}
y_{ijk} &= \mu + r_i + c_j + t_k + \epsilon_{ijk} \\
\end{align*}

- Overall mean
- Effect of row \(i\)
- Effect of column \(j\)
- Effect of treatment \(k\)
\[ y_{ijk} = \mu + r_i + c_j + t_k + \epsilon_{ijk} \] (1)

- Overall mean
- Effect of row \( i \)
- Effect of column \( j \)
- Effect of treatment \( k \)

http://tex.stackexchange.com/questions/55216/tikz-animated-equation-in-beamer
\begin{document}
\begin{frame}
\begin{itemize}
\item Overall mean \tikz[baseline] \node[coordinate] (s1) {};
\item \[
\begin{equation}
    y_{ijk} = \mu + r_i + c_j + t_k + \epsilon_{ijk}
\end{equation}
\]
\end{itemize}
\end{frame}
\end{document}
\item<3-> Effect of row $i$ \tikz[na] \node[coordinate] (s2) {};
\item<4-> Effect of column $j$ \tikz[na] \node[coordinate] (s3) {};
\item<5-> Effect of treatment $k$ \tikz[na] \node[coordinate] (s4) {};
\end{itemize}
\begin{tikzpicture}[overlay]
\path<2->[->] (s1) edge [bend left] (d1);
\path<3->[->] (s2) edge [bend right] (d2);
\path<4->[->] (s3) edge [out=0, in=-90] (d3);
\path<5->[->] (s4) edge [out=0, in=-90] (d4);
\end{tikzpicture}
1 \usepackage{tikz}
2 \usepackage{smartdiagram}
3 ...
4 \smartdiagram[flow diagram]{edit, pdflatex, bibtex, pdflatex, pdflatex}
Misc
\texttt{natbib} replaces the standard \texttt{\cite{}} command. Call \texttt{\usepackage[natbib][sort&compress]} to reorder and tidy multiple citations. Call \texttt{\usepackage[natbib][numbers]} or \texttt{\usepackage[natbib][authoryear]} to choose format.

<table>
<thead>
<tr>
<th>Command</th>
<th>Author/Year mode</th>
<th>Numbers mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{\citet{jon90}}</td>
<td>Jones et al. (1990)</td>
<td>Jones et al. [21]</td>
</tr>
<tr>
<td>\texttt{\citet[chap. 2]{jon90}}</td>
<td>Jones et al. (1990, chap. 2)</td>
<td>Jones et al. [21, chap. 2]</td>
</tr>
<tr>
<td>\texttt{\citep{jon90}}</td>
<td>(Jones et al., 1990)</td>
<td>[21]</td>
</tr>
<tr>
<td>\texttt{\citep[chap. 2]{jon90}}</td>
<td>(Jones et al., 1990, chap. 2)</td>
<td>[21, chap. 2]</td>
</tr>
<tr>
<td>\texttt{\citep[see][]{jon90}}</td>
<td>(see Jones et al., 1990)</td>
<td>[see 21]</td>
</tr>
<tr>
<td>\texttt{\citep[see][chap. 2]{jon90}}</td>
<td>(see Jones et al., 1990, chap. 2)</td>
<td>[see 21, chap. 2]</td>
</tr>
</tbody>
</table>
Useful packages

(sometimes helpful packages are automatically included with styles, e.g. sig-alternate, so check their documentation too)

1. **hyperref** - adds clickable links to urls, citations and internal references. 
   \usepackage[hidelinks]{hyperref} hides the boxes drawn around links.

2. **cite** - makes numeric citations pretty! Sorting and compression (e.g. [1-4, 7, 8]), as well as some other formatting. Alternative to natbib - useful if the latter isn’t compatible with your document class.

3. **soul** - provides \hl{stuff} so you can highlight text (e.g. TODOs) like this: stuff. Also improvements to hyphenation for other formatting like character spacing, underline, strikethrough and SMALL CAPS. Needs \usepackage{color} to highlight in colour.
Making your own commands

1 \textbf{\texttt{\newcommand\todo[2][Yum]{To do: \colorbox{yellow}{#2} - \textbf{#1}}}}
2
3 \texttt{\todo{have cake, eat it}}
4
5 \texttt{\todo[Mmm]{have cake, eat it}}

To do: \texttt{have cake, eat it} - \texttt{Yum}

To do: \texttt{have cake, eat it} - \texttt{Mmm}

- \texttt{\todo} is the new command’s name
- [2] is the number of parameters
- [Yum] is a default for the first parameter, making it optional
- The rest is the body of the command, with \texttt{#1} etc being the parameters
- Use \texttt{\renewcommand} in the same way to overwrite an existing one
Sometime we want to squeeze a tiny drop of space out of a paper. Usually we can rewrite to save a few lines, but in case we can’t, the following can be used:

1 \noindent
2 \vspace{-1cm}
3 \tiny

NB - this is a last resort - most “foo” usually breaks the formatting guidelines!
Some more tips:

- **Read the output from \LaTeX!**
- Often things can be resolved by deleting temp files and recompiling a couple of times
- Look out for document classes that redefine commands, or load packages that might conflict with the ones you want
- With \textasciitilde and many other commands the space after is part of the command, replacing a {} so writing this way results in no space after the command, e.g. Hello \textasciitilde New Word renders as: Hello ~New Word. "\textasciitilde " with two spaces doesn’t work because \LaTeX ignores redundant whitespace and the two spaces squash into one. Either write \textasciitilde{} or \textasciitilde~ to force a gap