Using LaTeX2e

ACM Lecture Series
July 17, 2003

Derek Williams
derk@acm.org
http://www.louisville.edu/~dawill03/

A companion website for this lecture is at
http://www.louisville.edu/~dawill03/LaTeX/
What is LaTeX2e?

- LaTeX2e (pronounced lay-tech two e) originated from TeX (pronounced tech), written by Donald Knuth in 1977.
- TeX is a program for typesetting text and mathematical formulae. It is a formatting engine.
- LaTeX2e is a macro package built on top of TeX.
What is LaTeX2e?

• LaTeX2e is not a WYSIWYG (what you see is what you get) editor.
• Writing in any variant of TeX is much like writing an HTML document, using various commands to format the way text appears on the screen.
When To Use LaTeX2e

- LaTeX2e is not a panacea.
- LaTeX2e does the following well:
  - Technical papers
  - Mathematics
  - Bibliographies & citations
  - Simple graphics
- Don’t be a fanatic and use LaTeX2e for everything – it won’t make life easier, which is exactly why you’d want to use LaTeX2e in the first place!
LaTeX2e Command Syntax

- LaTeX2e commands:
  - Are CaSe SeNsItIvE
  - Start with a backslash \ 
  - May have mandatory arguments enclosed in braces \{\}
  - May have optional arguments, separated by commas, enclosed in brackets \[[\] (these come before the mandatory arguments)
Starting a Document

• All LaTeX2e documents must start with the \documentclass{…} command.

• This command tells LaTeX exactly what type of document you’re creating. Valid arguments for this are:
  – article
  – report
  – book
  – slides
Starting a Document

• Various optional arguments exist for \documentclass, including (but not limited to):
  – 10pt, 11pt, or 12pt for font size
  – a4paper, legalpaper, etc. for paper size
  – landscape or portrait for layout on paper
• Defaults are 10pt, letterpaper, and landscape.
\begin{itemize}
\item \begin{document} and \end{document} are used for blocks of commands and text.
\item Documents must be encapsulated within \begin{document} and \end{document}.
\item There exist other uses for \begin{document} and \end{document}.
\item These commands begin and end environments – we shall consider other environments later.
\end{itemize}
A First Document

- Here is a very simple LaTeX2e file to begin with.
- Now we must compile this raw text into a document suitable for viewing and printing.

```latex
% simple.tex
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```
Compiling

- Multiple file formats are suitable for output, including: DVI, PS, PDF, and HTML.
- First, save the file with a `.tex` extension.
- Next, run `latex filename.tex`
- If errors are produced, the program will tell you exactly what’s wrong and where.
- The program will output the file `filename.dvi`
Compiling

• From the `.dvi` file we can output to the other formats, for example:
  – To go to PS (PostScript)
    • `dvips -Pcmz filename.dvi -o filename.ps`
  – To go to PDF (Portable Document Format)
    • `dvipdf filename.dvi`

• In many cases, you can go straight from `.tex` to `.pdf` like this:
  – `pdflatex filename.tex`
Special Characters

- Characters such as \ and \{} that are used in commands have to be entered in differently.

<table>
<thead>
<tr>
<th>Character</th>
<th>In LaTeX2e</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>$</td>
<td>$</td>
</tr>
<tr>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>^</td>
<td>^{}</td>
</tr>
<tr>
<td>&amp;</td>
<td>&amp;</td>
</tr>
<tr>
<td>_</td>
<td>_</td>
</tr>
<tr>
<td>{</td>
<td>{</td>
</tr>
<tr>
<td>}</td>
<td>}</td>
</tr>
<tr>
<td>~</td>
<td>~{}</td>
</tr>
<tr>
<td>\</td>
<td>$\backslash$</td>
</tr>
</tbody>
</table>
Whitespace & Comments

- Whitespace is considered much like in HTML – multiple spaces are only interpreted as one.
- Comments are done using the `%` character – everything after that character on a line is ignored, as well as any whitespace at the beginning of the next line.
- A tilde `~` character or braces `{ }` after a command will force a space to be recognized.
Paragraphs & Line Breaks

• Paragraphs are distinguished by leaving a blank line in the file. LaTeX2e will automatically indent for you.
• Word wrap exists, but if you want to force a single line break, put \ in the file.

% paragraphs.tex
\documentclass{article}
\begin{document}
It was the best of times\\
It was the worst of times
The End.
\end{document}
Page Breaks

- Page breaks are accomplished using \texttt{\newpage}

```latex
\documentclass{article}
\begin{document}
What a waste
\newpage
What a waste of paper!
\end{document}
```
LaTeX2e’s Behavior

- LaTeX2e will try to make your document look as nice as possible, but if it can’t figure out how to do so it will extend it out to the sides. LaTeX2e issue an “overfull hbox” warning in this case.

- Using `\sloppy` or `\fussy` will relax and restrict the rules, respectively.
Special Characters and Language

• Characters used in other alphabets, such as those with umlauts or tildes, exist in LaTeX2e, but once again there is a special way to write them. For a full table, see the references section.

• Language packages exist so that non-English commands can be used, too.
Partitioning Your Document

• Partitioning commands:
  – \chapter{...} (only in book and report)
  – \section{...}
  – \subsection{...}
  – \paragraph{...}
  – \subparagraph{...}
  – \part{...} (doesn’t increase numbering)
Table of Contents

• Once your document is partitioned, using the `\tableofcontents` command will insert a table of contents based on your partitioning.
  – Note: You must compile it twice in order for it to generate the table of contents correctly. Why? Because LaTeX2e operates in a sequential manner, and can’t insert information about the sections that it hasn’t even encountered yet!
Partitioning & Table of Contents

• Note that a section, or any partition for that matter, ends whenever a new one is defined.
• A * can be placed before the { in any partition command to make it not show up in the table of contents.

% sections.tex
\documentclass{article}
\begin{document}
\tableofcontents
\newpage
\section{This is a section}
Hey, check this out!
\subsection{This is a subsection}
Awesome!
\section{Next section}
Cool.
\end{document}
Titles

- Much like the table of contents, a title macro exists to automatically generate a nice looking title given some information.

- \texttt{\maketitle} can be called after \texttt{\title{...}} and \texttt{\author{...}} are defined.

```latex
% title.tex
\documentclass{article}
\title{How i Solved my Imaginary Problems}
\author{Gerolamo Cardano}
\begin{document}
\maketitle
Blah blah blah.
\end{document}
```
Referencing Partitions

- Use `\label{...}` to set a label for a partition, then use `\ref{...}` to recall the number of the partition and use `\pageref{...}` to recall the page it’s on.

---

% partref.tex
\documentclass{article}
\begin{document}
\section{This is a section}
Hey, check this out!
\subsection{Subsection here!}
\label{subsec}
Woo!
\newpage
\section{Next section}
Go to `\ref{subsec}`.\\
It's on page `\pageref{subsec}`.
\end{document}
Footnotes

• Footnotes are easily done by using the \footnote{...} command.

• Footnotes, just like sections, are automatically numbered – if you reorganize them, the numbers will still match.

% footnotes.tex
\documentclass{article}
\begin{document}
Foo\footnote{bar} and \footnote{Not you.}. friends\footnote{Not you.}.
\end{document}
Font Changing

• Here are some alternate fonts & modes:
  - \texttt{\ldots} – roman
  - \texttt{\ldots} – typewriter
  - \textit{\ldots} – italic
  - \textsc{\ldots} – SMALL CAPITAL LETTERS
  - \textsf{\ldots} – sans-serif font
  - \textbf{\ldots} – bold font
Font Changing

- The size of the font can also be redefined (these point values are for default $10\text{pt}$ as declared in $\texttt{\documentclass{...}}$):
  - $\texttt{\tiny} - 5\text{pt}$
  - $\texttt{\normalsize} - 10\text{pt}$
  - $\texttt{\Large} - 14\text{pt}$
  - $\texttt{\Huge} - 25\text{pt}$
Font Changing

• When in one mode, it is easy to underline or emphasize things.
  – Note: emphasis means different things in different fonts. In roman it means italics, but in italics it means roman.
Enumeration and Itemization

• Enumeration and itemization are accomplished using \texttt{item} in the environments \texttt{enumerate} and \texttt{itemize}, respectively.

• Note the nesting.

\% enumitem.tex
\documentclass{article}
\begin{document}
Here are some lists:
\begin{enumerate}
\item Lists are cool!
\begin{itemize}
\item Just a note
\item Another note
\end{itemize}
\item Next item here
\end{enumerate}
\end{document}
Text Alignment

- Text can be aligned to the left, right, or center. Use the following environments accordingly:
  - flushleft
  - flushright
  - center

% alignment.tex
\documentclass{article}
\begin{document}
\begin{flushleft}
West side!
\end{flushleft}
\begin{center}
In the middle.
\end{center}
\begin{flushright}
East side!
\end{flushright}
\end{document}
Quotes and Verbatim

• Block quotes can be formatted using the environment **quote**
• The **verbatim** environment does no formatting of the text – it will not parse anything inside.

% quoteverbatim.tex
\documentclass{article}
\begin{document}

Alexander Pope once said
\begin{quote}
Know then thyself, presume %
not God to scan;
The proper study of mankind %
is man.
\end{quote}

And here is some raw LaTeX2e:
\begin{verbatim}
Use \begin{quote} everyday!
\end{verbatim}
\end{document}
Mathematics

• Basic math is written using LaTeX2e in mostly the same way it would be done in a simple text file
• LaTeX2e really excels with formatting complex mathematics using commands.
• There are two main modes for writing math:
  – Inline, for a short equation or single variable
  – Math mode, for a longer, multi-line, equation
Inline Mathematics

• Inline mathematics can be accomplished by enclosing the math in dollar signs, $\$.  

• Note that exponents are enclosed in braces, and that instead of using * for multiplication, we have to use \cdot.

% inlinemath.tex
\documentclass{article}
\begin{document}
Some say that $2 + 2 = 4$ \%
but I say that $2^{2} = 4$ \%
and also that $2\cdot 2 = 4$. 
\end{document}
Math Mode

- Math mode puts math on a new line and centers it. It can be used in one of three ways, by placing math between the following:
  - \begin{displaymath} and \end{displaymath}
  - \[ and \]
  - $$ and $$
- Note that we can still go back into text mode by using \texttt{...} while in math mode.

\begin{verbatim}
\documentclass{article}
\begin{document}
Fermat's Last Equation:
\begin{equation}
\begin{split}
  x^n + y^n & \neq z^n \quad \text{for } n \geq 3
\end{split}
\end{equation}
\end{document}
\end{verbatim}
Numbering Equations

- Equations can be numbered, and referenced later.
- This is done with the \texttt{equation} environment and the \texttt{\label{...}} command.

\begin{verbatim}
\% numbeqns.tex
\documentclass{article}
\begin{document}
Pythagorean Theorem:
\begin{equation}
\label{py}
  c^2 = a^2 + b^2
\end{equation}
Please refer to (\ref{py}).
\end{document}
\end{verbatim}
AMS Symbols Package

- The American Mathematical Society has a symbols package \texttt{amssymb} for common math symbols that aren’t in LaTeX2e by default.
- \texttt{\textbackslash mathbb{...}} uses the “blackboard bold” font, which is typical for common sets of numbers:
  - \( Z \) is the set of integers
  - \( Q \) is the set of rationals
  - \( C \) is the complex set

```latex
\documentclass{article}
\usepackage{amssymb}
\begin{document}
Fermat's Last Theorem:
$$ x^n + y^n \neq z^n : \forall n \geq 3 \text{ and } n,x,y,z \in \mathbb{Z}\setminus\{0\} $$
\end{document}
```
Defining Commands

• When something will be used over and over again and would take a lot of space and time to type, we can create a new command as a macro.

• New commands can be written that do something as simple as print a character out, or they can be complex and take values as arguments which are then formatted.
New Command Definition

- New command definitions take two arguments, the first is the new command, the second is the old command.
- The command to define new commands is `\newcommand{...}{...}`
- Also note the use of `{}` after `\eqn` – if this is not done, LaTeX2e will remove the whitespace.

```latex
\documentclass{article}
\usepackage{amssymb}
\newcommand{\R}{$\mathbb{R}$}
\newcommand{\eqn}{$y = mx + b$}
\begin{document}
All values in \eqn{} are $\in$ \R.
\end{document}
```
New Command Definition

- Commands that take arguments must be defined as 
  \texttt{\textbackslash newcommand\{\ldots\} [\#] \{\ldots\}}
- For \texttt{n} arguments, they are numbered \texttt{1..n}.
- In the main document, when using the command, just pass the requisite number of arguments, in braces, to the new command.
- Also note the use of the \texttt{\log\{\ldots\}}, \texttt{\ln\{\ldots\}}, and \texttt{\frac\{\ldots\}\{\ldots\}} commands.

\begin{verbatim}
\documentclass{article}
\newcommand{\cob}[2]{% 
  \log_{#1}{#2} = \frac{\ln{#2}}{\ln{#1}}}
\begin{document}
Did you know that:
\[ \cob{2}{10} \]
and also that:
\[ \cob{5}{1000} \]
\end{document}
\end{verbatim}
Expanding Parentheses

- Parentheses will automatically be expanded to match the height of anything inside if you use `\left(` and `\right)`

```latex
\documentclass{article}
\begin{document}
Quadratic Formula:
$$ x = \left( \frac{-b \pm \sqrt{b^2-4ac}}{2a} \right) $$
\end{document}
```
Limits and Integrals

- Limits are done using \( \lim \{ \ldots \} \), where the argument is what goes under “lim”
- Integrals are done using \( \int \) for indefinite integrals, and \( \int_{\ldots}^{\ldots} \) for definite integrals.

---

```
\documentclass{article}
\begin{document}
Interesting:
\[
\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1
\]
Also interesting:
\[
\int 1 \, dx = x + C
\]
\[
\int_{0}^{3} 1 \, dx = 3
\]
\end{document}
```
Summations

- The summation command takes exactly the same arguments as the integral command: \( \sum_{\ldots}^{\ldots} \)
- Note the use of \( \infty \) for the infinity symbol.

```latex
\documentclass{article}
\begin{document}
It's a secret:
$$\sum_{n=0}^{\infty} n \approx \infty$$
\end{document}
```
Theorems

- Theorems are first defined like new commands. The full syntax is: 
  \texttt{\newtheorem{…} [#] {...} [section]}
- The first argument is a label for the environment,
- The second argument is a counter, and is optional
- The third argument is the text that appears in bold before the theorem
- The [section] argument will allow the theorem to be automatically numbered with the current section number.

\begin{document}
\begin{thm}[Chebyshev]
\pi(x) \asymp \frac{x}{\ln{x}}
\end{thm}
\begin{thm}
1 + 1 = 2
\end{thm}
\end{document}
Hyperlinks

- Hyperlinks can be added by importing the `hyperref` package. Various options can be specified.
- The syntax for a hyperlink is `\href{...}{...}`, where the first argument is the URL and the second is the text to be linked.

```latex
\documentclass{article}
\usepackage[pdftex, colorlinks]{hyperref}
\begin{document}
Click the \href{http://www.treeloot.com/}{tree} for money!
\end{document}
```
Bibliographies

• Bibliographies are done using the `thebibliography` environment, which takes an argument of the widest number to expect.
• Items are cited using `\cite{...}`, and references are written using `\bibitem{...}`, where both arguments are labels.
• If using the hyperlinking package, citations will automatically link to the references with a green color link.

% bibliography.tex
\documentclass{article}
\begin{document}
John likes cheese `\cite{ch}.
\newpage
\begin{thebibliography}{9}
  \bibitem{ch} Cheese!
  \bibitem{no} Nope.
\end{thebibliography}
\end{document}
References

• CTAN, the Comprehensive TeX Archive Network is on the web at http://www.ctan.org/

• The closest thing to a “bible” with respect to LaTeX2e is the following: