A. Donev (Courant Institute)

Tools

February 7th, 2018
LaTeX
LaTeX

What is LaTeX?

Some content taken from Wikipedia.

- TeX is a **typesetting** system: “allow anybody to produce high-quality books using minimal effort, and to provide a system that would give exactly the same results on all computers, at any point in time.” Knuth had the idea to use mathematics to typeset mathematics!

- LaTeX is a **markup language** for technical writing, with special emphasis on math-heavy writing, built on top of TeX: “TeX handles the layout side, while LaTeX handles the content side for document processing.”

- What’s a markup language and how does it differ from WYSIWYG ("what you see is what you get") word processors like Microsoft Word? Compare to html, and contrast interpreted versus compiled languages.

- Using LaTeX: **write-format-preview** (compare to code-compile-execute).
Why \LaTeX?

- Advantages of \LaTeX: (interactive)
  - Abstract: Separate presentation from content: focus on the content and not visual appearance.
  - Portable: \LaTeX files are simple text files so perfectly portable and easy to open/edit/share/diff.
  - Flexible: Change appearance/format by changing one word, e.g., the document class.
  - Extensible: macros allow one to add new functionality.

- Any advantages of WYSIWYG? (interactive)
  - \textbf{LyX} is a combination of the two: Focus on content but also see it on your screen! (Lyx Demo, including change tracking).
  - LyX files are still text files, in yet another markup language.
  - \textbf{Overleaf/sharelatex} (soon to merge!) is an alternative (Overleaf demo)
    Think of google docs versus Word.
How to LaTex

- Just like code, LaTex files need to be formatted to be organized, clear, readable by others:
  Yes, there is such a thing as bad LaTex just like there is bad code!

- If not using LyX/Overleaf, find a good LaTex editor (same as coding!):
  Use TexMaker to get started, or follow links from course homepage to programmer’s editors atom and sublime with LaTex plugins, or xemacs for “experts”

- What does a good editor provide? (interactive)?
  - Syntax highlighting
  - Indentation tools (automatic, select and indent, etc.)
  - Delimiter+block matching
  - Sophisticated find/replace with regular expression matching
  - Shortcuts/sub-windows to compile/preview
Installing a **LaTeX engine** ASAP (see links on homepage)
- For Windows/linux use TeXLive (usually pre-installed on linux)
- For OS X use MACTex and consider installing homebrew

Use `pdflatex` to compile/typeset (why?) and not `latex`

Add \texttt{\usepackage\{hyperref\}} to enable hyperlinks for references/citations.

Beware of **font issues** (PDF not actually as portable as dvi). Recommend inserting \texttt{\usepackage\{ae,aecompl\}} in latex preamble so PostScript-\textgreater PDF looks nice also.
- Learn how to use **BibTex+Mendeley** (demo, google scholar).
- For presentations in LaTex, use the **beamer class** (demo).
- How about **PowerPoint** or **keynote**? (interactive)
  Use **LatexIt** to format equations in latex as images.
- In LyX/Overleaf use **templates** to get started. Read **documentation**!
- What is **github** and **git/svn** all about? (demo and discussion)
  git is a distributed **version control system**; github is to git what Overleaf is to latex
Use **scalable vector graphics** for graphs (**EPS, SVG**) and not rasterized/pixelized formats!
If you must (e.g., huge figures), **use PNG** for line graphics and not JPG, as wavelets do not compress lines well.

- **Use indentation and spacing** liberally to improve readability
- **Do not insert manual line breaks** (editor handles splitting lines for you)
- **Use macros** to emphasize logical structure in the source, e.g:
  - Shortcuts: \texttt{\textbackslash def\textbackslash R\{\textbackslash M\{\textbackslash mathcal\{R\}\}\}}
  - Bold for vectors: \texttt{\textbackslash def\textbackslash V\#1\{\texttt{\textbackslash boldsymbol\{\#1\}}\}}
  - Norm of a vector: \texttt{\textbackslash def\textbackslash norm\#1\{\texttt{\textbackslash left\textbackslash Vert \#1\textbackslash right\textbackslash Vert }\}}
Number all formulas or only those you wish to reference later.

Add **labels** at the end or beginning of equations consistently to make it easy to find them.

Place labels for figures at the beginning of the caption.

Use **label prefixes** (eq:, fig:, tab:, sec:, subsec:, etc.) in order to be able to distinguish (this is done in LyX).

Use `\eqref` for referencing equations, `\ref` for figures/sections/etc.

AMS packages: `\usepackage{amssymb,amsmath}`

Use `\text{}` from package amsmath for inserting text into equations (not `\mbox` and definitely not `\text{}`).
For example, $x=y\quad \text{or} \quad x\neq y$

$$x = y \quad \text{or} \quad x \neq y.$$ 

Use `\textbf{}` from \texttt{bm} package for bolding letters to get italic letters instead of upright letters as with `\textbf{}`!

- `\textbf{b}` is typeset correctly with spacing $b$, but even better as $a\textbf{sin} x$
- `\textbf{mathbf{b}}` gives no spacing $b$

Always use `\operatorname{}` and not `\text{}`:

- $a\sin x$ is typeset correctly with spacing $a\sin x$, but even better as $a\sin x$
- $a\text{mathrm{sin}} x$ gives no spacing $asinx$

AMS packages let you define your own operator, e.g.,

`\DeclareMathOperator{\text{rank}}{rank}`
Insert **spaces in formulas** for readability using \, or \textthinspace, \,; or \textmedspace, \,; or \textthickspace, or \textquad and \textqquad for wider spaces.

Use **wide accents** \textwidetilde (\tilde{x} and \tilde{X}) and \textwidehat (\hat{x} and \hat{X}) instead of the narrow \tilde (\tilde{x} and \tilde{X}) and \hat (\hat{x} and \hat{X}) for capital letters.

For **matrices** use pmatrix (parenthesis) or bmatrix (brackets) environments.

Use **\left and \right** for delimiters to get automatic sizing, even if larger than strictly necessary.

- \left(\sum_{i=1}^{n}\right) gives (\sum_{i=1}^{n})
- (\sum_{i=1}^{n}) gives (\sum_{i=1}^{n})

Use **\textbf** to make text “italic” and not \textit.
To ensure things don’t get broken across lines use a tie ~, e.g., Knuth\cite{knuth}.

For a period different than a full stop, add control space e.g., ’p.\ 12’

Use - for hyphen (open-access repository), double dash -- for en-dash – (Moore–Penrose inverse), triple dash --- for sentence delimiter em-dash —.

Put numbers inside math to properly format sings, e.g., not -3 but –3.

For quotes use “text” and not double quotes (but LyX knows “text”).

Watch out: no blank lines after lists, quotations, and mathematical display formulas — this starts a new paragraph!
In math use \textbackslash colon to get punctuation, e.g., A(1\textbackslash colon r) to get A(1: r).

Use \textbackslash dots (or \textbackslash cdots) and not ... for \textbf{ellipses}

Lesson: The “right” way to do things in LaTex may seem obvious sometimes but it’s not, so \textbf{look at documentation} (web), e.g.

\begin{align*}
\int \int \int & \quad \text{versus} \quad \int \int \int
\end{align*}

\textbf{Any others? (discussion?)}