Getting started with LaTeX

Brian Buccola
Alanah McKillen

December 19, 2011

1 Introduction

1.1 What are TeX and LaTeX?

TeX is a powerful typesetting system designed by mathematician and computer scientist Donald Knuth back in 1978. It understands a few hundred low-level commands called primitives, which are rarely used directly anymore. Nowadays, most users use LaTeX, which is a markup language for TeX developed in the 1980s by Leslie Lamport. It incorporates many more useful and intuitive commands that allow users to concentrate on content rather than on formatting: you write, and LaTeX does the rest.

LaTeX comes in a variety of "distributions", the most common ones currently being TeX Live 2011 for Unix machines, and MiKTeX for Windows. (The most popular distribution for Mac OS X, MacTeX, is a bundle of TeX Live 2011 and some extra stuff.)

1.2 About this document

This document includes basic instructions on how to install the latest version of LaTeX onto Mac, Windows, and Linux machines. For more detailed instructions specific to your operating system, just type `latex installation`, followed by the name of your OS, into any good search engine.

In addition, this document describes the basic steps you'll run through to write, compile, and view your first, very simple document. You may also wish to look at the source file for this document, `latex-getting-started.tex`, to get an idea of what some of the commands and environments look like. (You can even edit and/or compile the source file yourself for practice.)

Finally, there are some resources at the end of the document. However, as always, a good search engine is your best resource.

If you have any questions or comments, just contact us at:

`brian.buccola@mail.mcgill.ca`
`alanah.mckillen@mail.mcgill.ca`

Happy holidays, and happy LaTeXing!

2 Mac OS X

Mac users will want to download MacTeX, available at `http://www.tug.org/mactex/2011/`. MacTeX is a package that consists of the complete TeX Live 2011 distribution of LaTeX, as well as many useful "TeXtras", like BibDesk (for managing references) and TeXShop (for easy editing, compiling, and viewing of documents).

To install, click the link above, and download the MacTeX package:

```
MacTeX.mpkg.zip
```

Once the `.zip` file is downloaded, unzip it, double click on `MacTeX.mpkg`, and follow the installation instructions that come up on the screen. After installing, you will have a `tex` folder in your applications folder which includes BibDesk, TeXShop, etc.

The package is over 1 GB in size, but it's worth the download: it contains just about every package, macro, font, etc. that you'll probably ever need, meaning you'll rarely (if ever) have to manually download new packages.

3 Windows

Windows users seem to have two options: MiKTeX, the standard Windows implementation of TeX for some time now, and proTeXt, a relatively new distribution based on MiKTeX. The LaTeX project website, `www.latex-project.org`, recommends proTeXt, but you may want to research a bit before choosing.

To install proTeXt, go to `www.tug.org/protext/`, and follow the directions.

To install MiKTeX, go to `www.miktex.org/2.9/setup`, and follow the directions.

Since Alanah uses Mac OS X and Brian uses Linux, neither of us has any experience installing or using LaTeX on a Windows machine. However, if you have questions, please contact us anyway, and we'll do our best to help.

4 Linux

The easiest way for Linux users to get LaTeX is to simply check your usual software source for TeX Live. You can either install `texlive`, which is a basic subset of the full package, or `texlive-full`, which is the complete package. `texlive-full` is recommended, if you have the time and storage space, since it will minimize the amount of packages, fonts, etc. you may have to install later.

The other option is to get TeX Live directly from `www.tug.org/texlive/`. Just follow the directions there, and/or ask Brian for help.

Note: Some Linux distros may be behind in their TeX packages. For example, Ubuntu currently ships with TeX Live 2009. So check before downloading from your software source, or get the latest version directly from the site just to be safe.

5 Writing, compiling, & viewing

Now that you've (hopefully) got LaTeX installed, let's produce a document! The basic steps are the following.

1. Using a good text editor, open a new text file, give it a `.tex` extension, and write up your document in plain text, using LaTeX markup (commands) for typesetting.
2. Save the file, and compile it using LaTeX, which produces a PDF output (or possibly DVI, depending on which command/setting is used).
3. View your output to see how it looks.

Note: By "good text editor", we just mean a text editor with LaTeX syntax highlighting. The editors that come with TeXShop (bundled with MacTeX) and with TeXnicCenter (bundled with proTeX) have syntax highlighting. Linux users have several options available, but we recommend Vim.

Note: TeXShop (and probably TeXnicCenter) has a button called "typeset" which, when pressed, saves your file, compiles it, and opens the output in a document viewer, all at once.

We'll now go through these steps in more detail, one by one.

5.1 Step 1: Write

Using your editor, open up a blank text file, and give it a `.tex` extension by saving the file as `hello-world.tex`. Now your editor knows to look out for LaTeX markup.

The first line you should write is the following. **Note:** LaTeX commands always start with a backslash.

```
\documentclass{article}
```

This command tells LaTeX what class of document you wish to write. The most common by far is the `article` class; it's suitable for nearly everything, including research papers, handouts, etc.

Next, begin the actual document by writing the following command beneath the one you just wrote.

```
\begin{document}
```

This command tells LaTeX where the body of the document actually begins. Everything after this is what gets produced in the final output.

Now let's produce a sentence. Beneath `\begin{document}`, write the following sentence.

```
Hello world!
```

For now, don't add anything else, because our purpose here is just to produce a successful output.

Now we end the document by adding the following, final line.

```
\end{document}
```

So your entire text file should look something like this:

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

Pretty simple! Now save the file, and we are ready to compile.

5.2 Step 2: Compile

Compiling a TEX file just means feeding LATEX a text file with a `.tex` extension, containing only plain text and LATEX markup, and getting a PDF (or DVI) output.

To compile, Mac and Windows users should look for a "typeset" or "compile" or "build" button in their text editor window. (You may first need to configure which output file type you want.) Linux users can run the following command in terminal:

```
pdflatex hello-world.tex
```

If all goes well, you should end up with a file called `hello-world.pdf`, or `hello-world.dvi`. If, however, you get an error, check your text file for typos, and also make sure it's been saved before compiling.

Note: You'll also end up with some files having extensions like `.aux`, `.out`, and `.log`. These are where LATEX keeps track of things like labels and compiling errors. You can ignore these files for now.

5.3 Step 3: View the output

This the easy part. Using your favorite document viewer (or whatever came with your LATEX distribution), open your PDF and make sure you see "Hello world!".

At this point, you can go back to your source file, `hello-world.tex`, and play around, edit, re-compile, and re-view. Don't forget to save each time before compiling!

6 Resources

The following is a list of useful resources for getting started with LATEX. You can find many more by searching online for `latex resources`, and you can almost

always find help with specific topics by searching directly for whatever it is you need, e.g. `latex tables`, `latex line spacing`, etc.

- LaTeX Wikibook
  `http://en.wikibooks.org/wiki/LaTeX`
  (Perhaps the best place to start; it's also a great reference, and the one that usually comes up first in Google results.)
- *The Not So Short Introduction to LaTeX 2ε*
  `http://tobi.oetiker.ch/lshort/lshort.pdf`
  (Much more detailed; super useful, especially when you're offline.)
- LaTeX for linguists
  `http://www.essex.ac.uk/linguistics/external/clmt/latex4ling/`
  (Tips on numbered examples, syntax trees, OT tableaux, etc.
  Caveat: some of this info is outdated.)