LaTeX and Asymptote Manual for Beginners

A simple and straightforward tool to learn mathematical typesetting

Aman Jain

April 2022

“Awesome quote”
-someone

Contents

0 Preliminaries

0.1 What is LaTeX?

Simply put, LaTeX is a typesetting language designed for mathematical/scientific texts, but its versatility allows it to be used in a variety of scenarios. At its core, LaTeX is used to create documents (articles, reports, books, journals, etc.) for academic purposes. As a language, LaTeX interprets plaintext and uses TeX to generate an output (usually a `.pdf` file).

0.2 Why learn and use LaTeX?

In scientific and academic settings, LaTeXis used extensively to make professional documents. But even for less formal and more personal purposes, LaTeX is extremely useful. It is much more powerful and produces much more visually appealing documents than typical editors like Google Docs or Microsoft Word. Also, it gives you, the user, a lot of control over how you would like your writing to look. Finally, there are tons and tons of built in and/or freely available *packages* which make a lot of document features like bibliographies, tables, graphics, formatting, and beyond very easy and customizable.

0.3 What is Asymptote?

Asymptote is a language (generally) used in LaTeX to generate vector graphics. Asymptote is written in its own language, but its syntax is similar to programming languages like `C++`. It is a powerful way to create a variety of images, from charts to geometry diagrams. Specifically, Asymptote is an environment in LaTeXused to add vector graphics to documents.

0.4 Why Asymptote?

It's true that LaTeX has other alternatives to Asymptote, most notably TikZ. As far as I know, there is no reason to lean towards either of the two. However, in my experiences I have always used Asymptote, and I have come across something it wasn't capable of doing. Furthermore, I think modules like `geometry.asy` and `olympiad.asy` make Asymptote the clear winner when it comes to geometry diagrams.

0.5 How do I use LaTeX?

I think online LaTeX compilers/editors (most notably Overleaf) are really useful and user-friendly. I used Overleaf for a while (and still do for certain things), and it has great functionality. In particular, it makes it really easy to collaborate on relatively short-term and small-scale projects.

That being said, from my experiences I can say that installing LaTeX on your computer is ultimately the way to go. (For example, do people usually code in repl.it or do people use editors like VSCode?) From what I know, all LaTeX softwares already include pretty much any package you could need in your projects, and they also make it easier to add your own packages and customization in the form of `.cls` and `.sty` files (more on these later). Also, compilation is much faster on your own device than it is through some other site. As an added bonus, LaTeX integrates well with `git`.

I have a Mac, so I use the MacTeX software, which includes TeXShop, a LaTeX editor. I'm not familiar with its equivalent on Windows or other operating systems, but the internet has plenty of resources.

Now that we have addressed formalities, it's time to dive into the actual learning!

1 Creating a LaTeX Document

For simple purposes, LaTeX can often just be used to create small mathematical images to be added to other documents. After all, the greatest aspect of LaTeX is its ability to render math. However, the true purpose of LaTeX is to typeset entire documents.

1.1 A Basic Document in LaTeX

The basic, bare structure of every document is

```
\documentclass{<class>}
\begin{document}
        <text and commands>
\end{document}
```

For now, it is not essential to understand each of these commands to the fullest; they will be addressed in future chapters. The important thing to understand is, every LaTeX document contains an *preamble* and a *body*. The preamble is everything that comes before the `\begin{document}`, and the body is the rest of the document, contained between the `\begin{document}` and `\end{document}`.

1.2 The Preamble

Every LaTeX document must have a preamble. At the very least, the preamble always has to specify the document *class*, using the command `\documentclass{<class>}`. In LaTeX, a class defines the structure and formatting of a document. Some of the more common classes are `article`, `report`, and `book`, but there are dozens more with their own uses. I won't be going over any individual classes in this manual, but each of these will have a documentation at CTAN (Comprehensive TeX Archive Network). The material found here can sometimes be on the complicated side though, but the internet will always be helpful as well; simple searches should be enough to find answers to most questions.

Most projects will need more than just a class in the preamble. The preamble is where you can take the class (say, article) and essentially customize it to fit your needs. In the preamble you can choose headers and footers, define your title, add custom formatting, and more. (The specific ways to do this varies from class to class; again, online resources will have ample help.) One of the most important components of th preamble is the addition of *packages*, which are

added using the `\usepackage{}` command. Packages are somewhat similar to classes, but they are more specific to certain parts of your document, whereas classes tend to provide an outline for how the entire document will feel and function. For instance, asymptote is a package used to create geometry diagrams (more on that later, of course). Some of the most prominent packages are `amsmath` and `amssymb`, `graphicx`, `xcolor`, `geometry`, `fancyhdr`, and more. Note that this list is nowhere near complete; there are hundreds of LaTeX packages, which are all exceedingly useful in certain scenarios.

Putting it all together, let's say you wanted to write a (rather short) article in which your

1.3 The Body