Writing Mathematics in LaTeX by Example

Jason B. Hill

University of Colorado – Boulder

Wednesday, October 31, 2008

Contents

1 Document Headers

The document header for this file looks like the following:

```
\documentclass[letterpaper,10pt]{article}
\usepackage{fullpage,amsfonts,amsmath,amssymb}

\setlength\parindent{0pt}

\author{\textsc{Jason B. Hill}\\\textsc{\small University of Colorado -- Boulder}}
\title{\textsc{Writing Mathematics in {\LaTeX} by Example}}
\date{\textsc{\small Wednesday, October 31, 2008}}
```

1.1 Document Classes

The first line of a LaTeX file is typically the `\documentclass[options]{class}` argument. This document uses the `article` class. Almost everything commonly written in mathematics will use this class. Documents written in the `article` class may contain sections, subsections, subsubsections, paragraphs and subparagraphs. They may not contain parts or chapters. (See Section 2.)

If you intent to print your LaTeX document duplex style (front and back) and bind it, then using the class `book` will alternate margin sizes left and right to accomodate the extra spacing needed by the binding.

1.2 Document Class Options

It's generally a good idea to specify your paper type and font size in the header. Doing this ensures that your audience will view the results of your LaTeX code as you intend. These options are in square brackets, such as `[letterpaper,10pt]` within the document class declaration.

For documents beind viewed/printed in North America: `[letterpaper]`
For documents beind viewed/printed outside North America: `[a4paper]`

The option `[letterpaper,10pt]` then makes a US letter sized paper with the standard font being 10pt.

1.3 Packages

LaTeX is mostly a giant collection of packages on top of old-school TeX. These packages are really what make LaTeX work. The packages typically used for math are:

`amsfonts`	Makes certain math symbols/fonts possible. E.g., $\mathbb{R}$
`amsmath`	Makes most math notation possible. E.g., $\displaystyle\int_1^x \frac{1}{x^2}\, dx$
`amssymb`	Make some symbols possible. E.g., $\blacksquare$

The `fullpage` package will make sure the margins of your page are 1 inch on all sides. Without it, you'll see that the article class in particular gives much bigger margins. More packages are discussed in this document as they are needed.

1.4 Document Length Declarations

You'll notice that the header for this file contains `setlength\parindent{0pt}`. By default, LaTeX indents new paragraphs, which can actually be quite annoying when writing mathematics. The `parindent` setting in the header in this case resets that length to zero. (You could make it bigger if you wanted to.) Other length declarations that may be useful are as follows.

`\parindent`	normal paragraph indentation
`\baselineskip`	vertical distance between lines in a paragraph
`\parskip`	extra vertical space between paragraphs
`\textwidth`	the width of text on the page
`\textheight`	the height of text on the page

1.5 New Commands and Renew Commands

Defining new commands and renewing standard commands can make your code writing more efficient. For instance, typically to write a $\mathbb{R}$ symbol in LaTeX one would write `\mathbb{R}` in math mode (more on math mode later). We can make this more efficient.

```
\newcommand{\mbr}{\mathbb{R}}
```

With the above line in the document (preferably in the header, but almost anywhere will do, just keep in mind that the command can only be used below where it is defined in the code), we would only need to type `\mbr` in math mode to achieve the same symbol. The idea is as follows: whenever an instance of the content inside the first brackets is found in your LaTeX document, it is replaced with the content in the second set of brackets and then compiled. The following example takes this a step further.

```
\newcommand{\mbr}[1]{\mathbb{#1}}
```

Here, `[1]` is an argument that we send to the command `\mbr` and in turn we get the output `\mathbb{1}`. For example, if we type `\mbr{Q}` in math mode, it will display $\mathbb{Q}$. It is also possible to renew commands so that they do something different than their current definition. For example, writing

```
\renewcommand{\mathbb}[1]{\mathf[thm]rak{#1}}
```

will force `\mathbb{C}` to display as $\mathfrak{C}$ (a fraktur letter) instead of $\mathbb{C}$ (a blackboard letter).

2 Document Organization

This is Section 2, although the number "2" doesn't appear anywhere in the code for this sentence. LaTeX automatically numbers many constructs (pages, sections, subsections, chapters, theorems, lemmas, etc.) and has a special environment for labelling and referencing such constructs. This is very convenient, since you can move constructs around (e.g., change the order of theorems) and LaTeX will automatically change any references. First, we will look at some of these constructs. Then, we will see how to assign labels and references.

2.1 Parts, Chapters and Sections

A part may include several chapters, a chapter may include several sections, each section may include several subsections, and so on. We can begin a new section (or whatever) by writing

```
\section{Put Section Title Here}
```

The constructs available in LaTeX for this sort of organization are as follows. (We will discuss them in a little more detail in the coming subsections.)

`\part{title}`	Can only be used in certain document classes. (Not in `article`.)
`\chapter{title}`	Can only be used in certain document classes. (Not in `article`.)
`\section{title}`	Automatically numbered 1, 2, 3, ...
`\subsection{title}`	When in Section 3, automatically numbered 3.1, 3.2, 3.3, ...
`\subsubsection{title}`	When in Subsection 4.3, automatically numbered 4.3.1, 4.3.2, ...

2.2   Table of Contents

Once you have defined sections and subsections, using the following command (preferably at the beginning of your document) will result in a table of contents where LaTeX will insert the document structures, substructures and page numbers automatically.

    \tableofcontents

You can force part/chapter/section/subsection/subsubsection names to appear differently in the Table of Contents than they appear in headings by using the following modification:

    \subsection[Parts, Chapters \& Sections]{Parts, Chapters and Sections}

Notice that in the current document, Subsection 2.1 has the title `Parts, Chapters & Sections` in the Table of Contents, yet has the title `Parts, Chapters and Sections` in the subsection heading. (The backslash is placed in front of the & because & is a special character in LaTeX and placing a backslash in front of it tells LaTeX to actually display a &.)

2.3   Theorems, Corollaries and Lemmas

Like Sections and other constructs, LaTeX can automatically number Theorems and such. We need to do a little work first to tell LaTeX explicitly how to number the theorems and how to display the content of the theorems. LaTeX thinks of five constructs here: theorems, corollaries, lemmas, definitions and remarks. Mathematical texts typically write the first three of these in italics, and the latter two in a regular font. Following that convention, the following should be placed in the document header.

    \newtheorem{thm}{Theorem}[section]
    \newtheorem{cor}[thm]{Corollary}
    \newtheorem{lem}[thm]{Lemma}

This is somewhat complex, but it tells LaTeX to order theorems by Section (so the third theorem in Section 2 will be Theorem 2.3) and to order corollaries and lemmas with the same ordering as theorems (so the first corollary in Section 5 will be Corollary 5.7 if it comes after Theorem 5.6). Removing the `[section]` declaration will order all theorems, corollaries and lemmas linearly throughout the entire document (the third such theorem/corollary/lemma in the document will be numbered 3 regardless of which section in which it appears). The `[thm]` declaration in the Corollary and Lemma definitions just tell LaTeX to tie the numbering of Corollaries and Lemmas to the numbering of Theorems. If you remove this declaration, it would be possible to have Lemma 1 appear after Theorem 32, for instance.

For proofs, use the `\begin{proof}` and `\end{proof}` environments (these require the `amsthm` package). LaTeX will automatically place a square at the end of the proof. If you want to use a different symbol, renew (Section 1.5) the command `\qedsymbol`.

The following is an example that demonstrates this. Notice that we are currently in Section 2 of the current document, and so we expect the first theorem/corollary/lemma to be numbered 2.1. Here is what we type:

```
\begin{thm}
This is a theorem.
\end{thm}

\begin{proof}
This is the proof. Notice the box thing at the end.
\end{proof}

\begin{cor}
Here is a corollary.
\end{cor}

\begin{lem}
And a lemma.
\end{lem}

\begin{thm}[Some Famous Name]
 And another theorem.
\end{thm}
```

That will display the following.

Theorem 2.1. *This is a theorem.*

Proof. This is the proof. Notice the box thing at the end. □

Corollary 2.2. *Here is a corollary.*

Lemma 2.3. *And a lemma.*

Theorem 2.4 (Some Famous Name)**.** *And another theorem.*

2.4   Labels and References

For items in LaTeX which contain some sort of number, we can assign a label to that number and reference it throughout the document. To place a label, use the `\label{labelname}` command. To reference the labelled item, use the `\ref{labelname}` command. For example, if we wish to reference a theorem through the text of a document, the following will accomplish that.

```
\begin{thm}\label{importanttheorem}
This is a very important theorem.
\end{thm}

We can reference Theorem \ref{importanttheorem} like so.
```

That will display the following:

Theorem 2.5. *This is a very important theorem.*

We can reference Theorem 2.5 like so.

LaTeX will automatically number many equations and other items as well. Since there are so many things that may be numbered, it is generally a standard practice to label things so that the label gives some information about the item it labels. For instance, a label of `\label{thm:HeineBorel}` would be used on a theorem (thm) named Heine Borel and a label of `\label{eqn:myequation}` could be used on an equation (eqn).

3 Writing Text

Generally, writing text in LaTeX results in that text showing up in your document. There are a few exceptions in the form of special characters. There are also some formatting issues.

3.1 Special Text Characters in LaTeX

The following characters will not appear as expected in LaTeX as they do special things. If you wish for these characters to appear as themselves in your document, you must place a backslash before them, with the exception of the \ symbol itself which is written as `$\backslash$`.

$$\#, \ \$, \ \%, \ \sim, \ \backslash, \ \_, \ \{, \ \}$$

Other characters (mainly those not appearing on your keyboard) will have special commands that display them in LaTeX. Much of the time learning LaTeX is spent learning these commands. A good reference (or an editor with a nice collection of selectable characters) is always nice to have.

3.2 Font Faces

The following will generally only work in text mode (i.e., generally won't work in math mode).

Command	Font	Example	Result
`\text{}`	Plain Text (for inside math mode)	`\text{Example 1}`	Example 1
`\textsf{}`	Sans Serif Font	`\textsf{Example 2}`	Example 2
`\texttt{}`	Typewriter Font	`\texttt{Example 3}`	`Example 3`
`\textbf{}`	Bold Face Font	`\textbf{Example 4}`	**Example 4**
`\textit{}`	Italic Font	`\textit{Example 5}`	*Example 5*
`\textsc{}`	Small Caps Font	`\textsc{Example 6}`	Example 6

3.3 Font Sizes

You want to declare font sizes *within an environment* or LaTeX will try to declare them globally since there is no terminating comand for a font size. Putting curly brackets around the text you want to size, or using font size declarations within other commands are ways of accomplishing this.

Command	Example	Result
`\tiny`	`{\tiny Example 1}`	Example 1
`\scriptsize`	`{\scriptsize Example 2}`	Example 2
`\footnotesize`	`{\footnotesize Example 3}`	Example 3
`\small`	`{\small Example 4}`	Example 4
`\normalsize`	`{\normalsize Example 5}`	Example 5
`\large`	`{\large Example 6}`	Example 6
`\Large`	`{\Large Example 7}`	Example 7
`\huge`	`{\huge Example 8}`	Example 8
`\Huge`	`{\Huge Example 9}`	Example 9

3.4 Horizontal Spacing

Spacing can be tricky in LaTeX. Horizontal space commands can be used in both text and math environments. Inserting a space in text mode generally results in a space in the document. But, a collection of spaces together will show up as a single space. Inserting a space in the code in math mode will generally have no effect. For more options, use the following.

Command	Example	Result
\quad	`Used to insert \quad four \quad spaces`	Used to insert four spaces
\qquad	`Eight\qquad spaces`	Eight spaces
~	These~spaces~won't~break~into a~new~line	These spaces won't break into a new line
\,	Half a space. E.g., `Hold breath for 10\,s.`	Hold breath for 10 s.
\hspace{}	`Insert\hspace{0.5in}half an inch`	Insert half an inch

3.5 Vertical Spacing

Vertical spacing in LaTeX can take some practice. Here are some guidelines.

- Writing a `\\` results in a line break. For example: "`This is a \\ line break`" produces

    This is a
    line break

- Writing `\newline` forces a new line. For example: "`This is a \newline new line`" produces

    This is a
    new line

    The difference between `\newline` and `\\` is that `\newline` will attempt to justify text when possible, since it creates a new paragraph while `\\` does not.

- Start a new paragraph by skipping a line in your LaTeX document. Skipping more than one line will have no added effect.

- You may force vertical space with the `\smallbreak, \medbreak, \bigbreak` commands. Many LaTeX authors will both skip a line and insert a `\bigbreak` at the beginning of a new paragraph.

- You may force a certain amount of vertical space with the `\vspace{}` command. In text mode, this command must have a line break before it (cannot be written inline). To skip a large amount of space, write something like `\vspace{5in}` or `\vspace{10pc}`

3.6 List Environments

The first example of a list environment is a bulletted list. (Many times there is a command in your editor to start such a list for you.) Such a list starts with the command `\begin{itemize}` and ends with `\end{itemize}`, with each bullet created by the command `\item`.

```
\begin{itemize}
    \item Here is a bullet.
    \item And another bullet.
    \begin{itemize}
        \item Some subbullet.
        \item And another.
    \end{itemize}
    \item And a final bullet.
\end{itemize}
```

- Here is a bullet.

- And another bullet.

    - Some subbullet.
    - And another.

- And a final bullet.

Another popular kind of list is an enumeration.

```
\begin{enumerate}
    \item Here is the first thing.
    \item Here is the second thing.
    \begin{enumerate}
        \item First subthing.
        \item Second subthing.
    \end{enumerate}
    \item Third thing.
\end{enumerate}
```

    (a) First subthing.

    (b) Second subthing.

3. Third thing.

1. Here is the first thing.

2. Here is the second thing.

1. Here is the first thing.

2. Here is the second thing.

    (a) First subthing.

    (b) Second subthing.

3. Third thing.

If you wish to change the numbering, or have it display in a specific way, you can use the following sort of option when declaring an item:

```
\item[\textbf{13:}]
```

Notice the difference.

```
\begin{enumerate}
    \item[\textbf{1.}] Here is the first thing.
    \item[\textbf{2.}] Here is the second thing.
    \begin{enumerate}
        \item[\textbf{2a.}] First subthing.
        \item[\textbf{2b.}] Second subthing.
    \end{enumerate}
    \item[\textbf{3.}] Third thing.
\end{enumerate}
```

1. Here is the first thing.

2. Here is the second thing.

    2a. First subthing.

    2b. Second subthing.

3. Third thing.

3.7 Tabular Environments

Tables can also take some practice in LaTeX. If you want something other than the basic table, look online as there are entire pages that are devoted to table structures in LaTeX. But keep this in mind: Tabular environments are for text and arrays are for math.

```
\begin{center}
    \begin{tabular}{|r|c|l|}
    \hline
    \hline
    \textbf{Col 1} & \textbf{Col 2} & \textbf{Col 3}\\
    \hline
    r1c1 & r1c2 & r1c3 \\
    r2c1 & r2c2 & r2c3 \\
    \hline
    \end{tabular}
\end{center}
```

Col 1	Col 2	Col 3
r1c1	r1c2	r1c3
r2c1	r2c2	r2c3

One of the main things to notice in the tabular environment is the {|r|c|l|} option. This tells LaTeX to use three table columns, with text aligned right, center and left respectively. The | just says to draw a vertical line. The command \\ (which is usually a new line command) tells LaTeX that the current row is finished, where rows are separated by the & symbol.

4 Math Mode

There are many different ways of writing things in math mode. The best way to learn is by example. First, we'll introduce "inline math mode" and "block math mode."

4.1 Generic Notes on Math Modes

- To make subscripts, use _. For instance, `$x_\alpha$` produces x_α.

- To make superscripts, use ^. For instance, `$x^\beta$` produces x^β.

- You can use the two together: `$x_\alpha^\beta$` produces x_α^β.

- If you want to include more than one "thing" as a sub/superscript use curly brackets:

  `$\mathfrak{A}_{\alpha+\beta}^{\gamma+\delta}$`   produces $\mathfrak{A}_{\alpha+\beta}^{\gamma+\delta}$

4.2 Inline Math Mode

Inline Math Mode enables you to write math inline so things like $f(x) = x^2 + 2x - 8$ appear within the regular flow of text.

To write inline math, put a `$` on either side of your math. For example, writing `$h(x)=e^x$` produces the inline math text $h(x) = e^x$.

- Many mathematical notations will display in a compact version when written in inline mode. For instance, writing `$\int_0^1\sin x\,dx$` results in $\int_0^1 \sin x \, dx$. You may attempt to get around this in certain cases by using `$\displaystyle \int_0^1\sin x\,dx$`. In this case, writing that displays the following: $\displaystyle \int_0^1 \sin x \, dx$.

- You may use text within math mode with the `\text{}` command.

4.3 Block Math Modes

There are several options here. If you just want to include math in a block (centered on the page, not compact like inline math mode) then the most common option is to write something like the following.

```
\[
\int_0^1\sin x\,dx
\]
```

This produces:

$$\int_0^1 \sin x \, dx$$

4.3.1 Equations

Notice the difference between

```
\begin{equation}
\cos(x)=\cos(-x),
\end{equation}
```

which produces

$$\cos(x) = \cos(-x), \tag{1}$$

and

```
\begin{equation*}
\cos(x)=\cos(-x),
\end{equation*}
```

which produces

$$\cos(x) = \cos(-x).$$

The difference is that LaTeX automatically numbers equations in the `equation` environment, while it doesn't number those in the `equation*` environment.

4.3.2 Aligns

Some LaTeX authors will recommend the `eqnarray` environment here. Use of that environment is not recommended (it was removed from the LaTeX standard many years ago) as it displays different in various compilers and there is no protection for spacing (equation numbers may be overrun). If you submit your code to a journal, for instance, using the `eqnarray` environment, there is no sure bet that their compiled result will appear as the result on your personal computer.

An example of the `align` environment is as follows:

```
\begin{align*}
\sum_{k=1}^5(k^2+2) & = (1^2+2)+(2^2+2)+(3^2+2)+(4^2+2)+(5^2+2)\\
      & = 3+6+11+18+27\\
      & = 65.
\end{align*}
```

This produces the output:

$$\sum_{k=1}^{5}(k^2 + 2) = (1^2 + 2) + (2^2 + 2) + (3^2 + 2) + (4^2 + 2) + (5^2 + 2)$$
$$= 3 + 6 + 11 + 18 + 27$$
$$= 65.$$

We can make multiple equation aligns per row as follows:

```
\begin{align*}
    a_{11}  & =  b_{11}    &    a_{12}  & =  b_{12}\\
    a_{21}  & =  b_{21}    &    a_{22}  & =  b_{22}+c_{22}.
\end{align*}
```

$$a_{11} = b_{11} \qquad\qquad a_{12} = b_{12}$$
$$a_{21} = b_{21} \qquad\qquad a_{22} = b_{22} + c_{22}.$$

4.3.3 Misc. Block Math Environments

Here, the left will correspond to what we write in LaTeX and the right will be the corresponding displayed result.

```
\begin{multline*}
    a+b+c+d+e+f\\
    +i+j+k+l+m+n
\end{multline*}
```

$$a + b + c + d + e + f$$
$$+ i + j + k + l + m + n$$

```
\begin{gather}
a_1=b_1+c_1\\
a_2=b_2+c_2-d_2+e_2
\end{gather}
```

$$a_1 = b_1 + c_1 \tag{2}$$
$$a_2 = b_2 + c_2 - d_2 + e_2 \tag{3}$$

4.4 More, Largely Random Examples

Again, the input is on the left. The output is on the right, except where this would create spacing problems, in which cases the input is displayed above the output.

Fractions

```
\frac{2}{3}
```
$$\frac{2}{3}$$

We can force LaTeX to not compress fractions. (If you make a very complicated fraction, the text can start to become compressed.) Use `\tfrac{}{}` in text mode and `\dfrac{}{}` in math modes.

```
\begin{equation*}
\Re{z} =\frac{n\pi \dfrac{\theta +\psi}{2}}{
        \left(\dfrac{\theta +\psi}{2}\right)^2 + \left( \dfrac{1}{2}
        \log \left\lvert\dfrac{B}{A}\right\rvert\right)^2}.
\end{equation*}
```

This outputs:

$$\Re z = \frac{n\pi \dfrac{\theta + \psi}{2}}{\left(\dfrac{\theta + \psi}{2}\right)^2 + \left(\dfrac{1}{2}\log\left|\dfrac{B}{A}\right|\right)^2}.$$

Binomial Expressions
LaTeX uses the `\binom` command. See the following example.

```
2^k-\binom{k}{1}2^{k-1}+\binom{k}{2}2^{k-2}
```

$$2^k - \binom{k}{1}2^{k-1} + \binom{k}{2}2^{k-2}$$

Continued Fractions

```
\cfrac{1}{\sqrt{2}+
 \cfrac{1}{\sqrt{2}+
  \cfrac{1}{\sqrt{2}+\dotsb
}}}
```

$$\cfrac{1}{\sqrt{2}+\cfrac{1}{\sqrt{2}+\cfrac{1}{\sqrt{2}+\cdots}}}$$

Limits and Such

```
\lim_{x \to +\infty}, \inf_{x > s} \text{ and } \sup_K
```

$$\lim_{x \to +\infty}, \inf_{x > s} \text{ and } \sup_K$$

Sums

```
\sum_{k=1}^n k^2 = \frac{1}{2} n (n+1).
```

$$\sum_{k=1}^n k^2 = \frac{1}{2}n(n+1).$$

Integrals

```
\int_0^4 x^2+2x-\frac{1}{2}x\,dx
```

$$\int_0^4 x^2 + 2x - \frac{1}{2}x\,dx$$

Here is a more complicated example. The limits are moved under the integral instead of beside it.

```
    \int\limits_{x^2 + y^2 \leq R^2} f(x,y)\,dx\,dy
      = \int_{\theta=0}^{2\pi} \int_{r=0}^R
        f(r\cos\theta,r\sin\theta) r\,dr\,d\theta
```

$$\int\limits_{x^2+y^2\leq R^2} f(x,y)\,dx\,dy = \int_{\theta=0}^{2\pi} \int_{r=0}^R f(r\cos\theta, r\sin\theta)r\,dr\,d\theta$$

Matrices

```
\left(\begin{array}{ccc}
   a & b & c \\
   d & e & f \\
   g & h & i
\end{array}\right)
```

$$\left(\begin{array}{ccc} a & b & c \\ d & e & f \\ g & h & i \end{array}\right)$$

Certain Characters
Don't ever use the keyboard keys $<$ or $>$ when you mean to use one of the following

```
\langle A \rangle
```

$$\langle A \rangle$$

```
\left\langle \frac{A}{B} \right\rangle
```

$$\left\langle \frac{A}{B} \right\rangle$$