Some Tips and Tricks for Using LaTeX in Math Theses

by Rob Benedetto

How to Use the files `samplethesis.tex`, `thesis.tex`, and `latextips.tex`

WARNING!!!! This document (`latextips.tex`) is **NOT** a good model to build a math thesis from. The margins are wrong, the spacing is wrong, and the style of writing is far too colloquial. (For example, you should avoid contractions and the pronouns "I" and "you," avoid using quotation marks unless you're actually *quoting* something, minimize the use of parenthetical comments and the word "so," and generally use a more formal writing style than you'll find in this `latextips.tex` document.)

Instead, you should use the file `samplethesis.tex` as a model for your thesis. That file has all the margins and spacing set properly, and it is written in a style appropriate for a math thesis. To actually write your own thesis, start with the file `thesis.tex`, which has all the correct structure and formatting commands you will need but no written content.

Still, I hope you will find this `latextips.tex` document useful as well. It is intended to give suggestions for how to use LaTeX effectively, to give a wider variety of examples of LaTeX tricks than appeared in `samplethesis.tex`, and to explain some of the inner workings of `samplethesis.tex` and `thesis.tex`. That is, you should use `thesis.tex` for its structure, `samplethesis.tex` and `samplethesis.pdf` as an example, and `latextips.tex` and `latextips.pdf` for advice.

I'd recommend you look not only at the three raw `.tex` files, but also at the LaTeX-ed and printed documents they generate. There are a lot of little tricks and techniques embedded in them, and you should familiarize yourself with both the raw LaTeX code and the resulting printed display. Quite a few of those tricks, like how to make the double-bar norm symbols $\|\cdot\|$ that appear later, are not explicitly mentioned in the text of this document, so be ready to scroll back through the `.tex` file to learn how some symbol was generated.

Of course, there's much more to LaTeX than what you'll find here; so get yourself a good LaTeX book, such as G. Grätzer, *Math into LaTeX,* 3rd ed., Birkhäuser, Boston, 2000. There are online sources, too. For example, the LaTeX Wikibook may be found at

`http://en.wikibooks.org/wiki/LaTeX`

More links to a number of online LaTeX manuals and tutorials at

`http://www.tug.org/interest.html#latexmanuals`

Chapter 1

Some LaTeX Examples

First, a quick comment that can only go here, even though it is topically out of place: if I put a Subsection here (before the first Section of a Chapter), I get a weird number, like this:

1.0.1 Subsection Numbers Involving Zero

See, it looks strange to refer implicitly to Section Zero. The same thing happens[1] with Theorems (or Definitions, etc.) appearing before the first Section of a Chapter; see, e.g., Theorem 1.0.1 in `samplethesis.tex`. Avoid numbering involving zeros in a thesis.

1.1 Math Commands

Multiline equations can be generated with the `align*` environment, using `&` for alignment points and column breaks, and `\\` for newlines:

$$\begin{aligned}\psi_i(\log r) &= (p^i+1)\log\|f\|_{\nu(x,r)} - 2p^i(\log r + \log\|f'\|_{\nu(x,r)}) \\ &= (1-p^i)\log\|f\|_{\nu(x,r)} - 2p^i\log\delta(f,\nu(x,r)) \\ &\geq \log\|f\|_{\nu(x,r)} - p^i\Delta(f,\nu(x,r)).\end{aligned}$$

If you want labels on that kind of equation, try the `align` environment.

$$\begin{aligned}\log\|f\|_{\nu(x,\tilde{R})} &\geq \log\|f\|_{\nu(x,R'')} + \frac{2p}{p-1}[\delta(r,\nu(x,R'')) - \delta(r,\nu(x,\tilde{R}))] \\ &\geq \frac{1}{p-1}\left[p\log|\alpha| + 2p\log\mu - \log\|f\|_{\nu(x,R'')}\right]. \qquad (1.1)\end{aligned}$$

Note that since I only wanted the second line, and not both lines, of inequality (1.1) to have a label, I used the `\notag` command in the `.tex` file on the first line.

You can refer to labelled equations, like equation (1.1) or sequence (1.2), from anywhere in the paper, even before they appear. But make sure to use `\eqref` rather than `\ref` when referencing equations; `\ref` gives us equation 1.1 rather than the nicer equation (1.1).

[1] That's because I numbered Theorems by Section in `samplethesis.tex`. You'd also get zeros if for some reason you put a Theorem before Chapter 1, even if you number your Theorems by Chapter.

If you want just a regular one-line displayed equation labelled, use the `equation` environment. For example, here's a sequence of ones and zeros.

$$\underbrace{0,\ldots,0}_{m_0},\underbrace{1,\ldots,1}_{M_1},\underbrace{0,\ldots,0}_{m_1},\underbrace{1,\ldots,1}_{M_2},\underbrace{0,\ldots,0}_{m_2},\ldots. \tag{1.2}$$

For matrices, use the `matrix` environment, again using `&` for column breaks and `\\` for newlines. Use `\left` and `\right` to make brackets big enough to fit around whatever is between them. Use `\cdots`, `\vdots`, and `\ddots` to get various directions of dots:

$$A = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1} \end{pmatrix}, \quad \langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^T \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \mathbf{b}, \quad \text{and} \quad \nabla \times \vec{F} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ P & Q & R \end{vmatrix}.$$

If you want, you can skip typing out the `\left(` and `\right)` parentheses commands for matrices by replacing the `matrix` environment with `pmatrix` for regular (round) parentheses, `bmatrix` for square brackets, `vmatrix` for vertical lines, `Vmatrix` for double vertical lines, or `Bmatrix` for curly brackets. For example,

$$\operatorname{Tr} \begin{pmatrix} 1 & 0 & -4 \\ -2 & 5 & 1 \\ -1 & -1 & 8 \end{pmatrix} = 1 + 5 + 8 = 14,$$

and

$$\det \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1} \end{bmatrix} = \begin{vmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1} \end{vmatrix} = \prod_{1 \le i < j \le n} (x_j - x_i).$$

You can use `\left(` and `\right)` (or `\left[` and `\right]`, or `\left\{` and `\right\}`, or `\left|` and `\right|`, or even `\left(` and `\right]` — they don't have to match) to make brackets around any clump of math symbols. But sometimes you may want big but not huge brackets. For example, the parentheses around

$$\left(\sum_{\substack{1 \le i \le n \\ i \ne 5}} i^2 \right)^3 \qquad \text{and} \qquad (\sum_{\substack{1 \le i \le n \\ i \ne 5}} i^2)^3$$

are way too big and way too small, respectively. Instead, the expression

$$\Big(\sum_{\substack{1 \le i \le n \\ i \ne 5}} i^2 \Big)^3$$

looks much better, even if the subscript sticks out the bottom a little. You can make parentheses like these, anywhere from one to four specific sizes bigger, using the `\big`, `\Big`, `\bigg`, or `\Bigg` commands just before the parentheses.

For integrals, try using the `\,` spacing command to get just the right amount of space before the dx (or in this case, $d\theta$):

$$L_{\mathbb{C}}(r) = r\int_0^{2\pi} f^{\#}(re^{i\theta})\,d\theta \quad \text{instead of} \quad L_{\mathbb{C}}(r) = r\int_0^{2\pi} f^{\#}(re^{i\theta})d\theta.$$

For multiple integrals, you can do

$$\iint_S (\operatorname{curl}\vec{F})\cdot d\vec{S} = \oint_C \vec{F}\cdot d\vec{r} \qquad \text{or} \qquad \iiint_E \operatorname{div}\vec{G}(x,y,z)dV = \iint_S \vec{G}\cdot d\vec{S},$$

but if you want iterated multiple integrals, just use `\int` multiple times:

$$\int_0^1\int_0^{1-x}\int_0^{1-x-y^2} \frac{dz\,dy\,dx}{x\sin(yz)+3}.$$

As you may have noticed, for certain symbols like sums and products, LaTeX treats subscripts and superscripts differently in displayed versus in-line mathematics. For example,

$$\int_a^b f(x)\,dx = \lim_{n\to\infty}\sum_{i=1}^{n} f(x_i)\Delta x_i.$$

becomes $\int_a^b f(x)\,dx = \lim_{n\to\infty}\sum_{i=1}^{n} f(x_i)\Delta x_i$ in in-line math mode, so that it will fit vertically. You can use the `\displaystyle` command to get $\displaystyle\int_a^b f(x)\,dx = \lim_{n\to\infty}\sum_{i=1}^{n} f(x_i)\Delta x_i$ in in-line math mode. But either way, if an in-line math equation is too big, it can make the paragraph hard to read. Big equations should be displayed.

The same sizing issue comes up with fractions, even in displayed math. For example,

$$\lim_{h\to 0}\frac{\frac{x+h+2}{(x+h)^2}-\frac{x+2}{x^2}}{h} \quad \text{and} \quad f(x,y) = \begin{cases} \frac{x^2+3xy^2}{x^2+y^2} & \text{if } (x,y)\neq(0,0) \\ 0 & \text{if } (x,y)=(0,0) \end{cases}$$

have tiny, unreadable fractions. You can force those fractions to be bigger by using `\dfrac` instead of `\frac`, to obtain

$$\lim_{h\to 0}\frac{\dfrac{x+h+2}{(x+h)^2}-\dfrac{x+2}{x^2}}{h} \quad \text{and} \quad f(x,y) = \begin{cases} \dfrac{x^2+3xy^2}{x^2+y^2} & \text{if } (x,y)\neq(0,0) \\ 0 & \text{if } (x,y)=(0,0). \end{cases}$$

Here are some other miscellaneous LaTeX commands to be aware of:

- `\varepsilon` gives a different-looking epsilon: ε instead of the default ϵ.
- `\varphi` gives a different-looking phi: φ instead of ϕ.

- `\smallsetminus` gives a smaller set subtraction sign than `\setminus` — compare $A \smallsetminus B$ with $A \setminus B$.
- `\varnothing` gives a more circular nullset sign than `\emptyset` — compare $\varnothing$ with $\emptyset$.
- `\nmid` gives the "does not divide" sign. In general, if you want to put a slash through a symbol, precede the symbol with `\not`, as in $x \notin A$ or $x \not\leq 5$. (For example, `\neq` is effectively just an abbreviation for `\not =`, albeit a very useful one.) But if you try `\not |` or `\not \mid`, you get the ugly $\not|$ or $\not\mid$, whereas `\nmid` gives the much nicer $\nmid$.
- `\binom{n}{m}` makes the "n choose m" binomial coefficient symbol, giving

$$\binom{n+1}{k+1} = \binom{n}{k} + \binom{n}{k+1}$$

  for displayed math mode, and $\binom{7}{5}$ for in-line math mode.

The bullet list above was produced by an `itemize` environment. (To get the symbol $\bullet$ by itself, use `\bullet` in math mode.) LaTeX also has two other built-in list environments: `enumerate`, which gives numbered lists, and `description`, which gives lists where each item begins with a different label. There's also the much more flexible (but harder to use) `list` environment, but you are unlikely to need it for a thesis.[2]

1.2 Sectioning Commands and Other Comments

1.2.1 Subsections

Besides chapters and sections, there are also subsections. But don't have only one subsection in a section, or only one section in a chapter. Why break something up into one piece?[3]

1.2.2 Hats and Bars

To put a hat on a math symbol, use `\hat`, as in $\hat{h}(x)$ or $\hat{f}(\zeta)$. But sometimes `widehat` looks better, as in $\widehat{G}$ or $\widehat{\mathbb{R}}$. Similarly, `\bar` gives $\bar{f}$, but `\overline`, like `\widehat`, will stretch, as in $\overline{f}$, or even $\overline{abcd}$.

---

[2] I used a `list` environment to get the alphabetically enumerated list in the statements of Proposition 2.1.2 and Lemma 3.2.2 of `samplethesis.tex`.

[3] The hierarchy of sectioning commands actually continues further to subsubsection, paragraph, and subparagraph. But let's not get carried away. Besides, these lesser cousins of sections aren't numbered, so you can't refer to them with `\label` and `\ref`.

By the way, `\label` assigns to that label-name the number of the current structure—chapter, section, subsection, theorem, equation, etc.—that we're deepest inside **and that has a number**. If you use `\label` in a lesser cousin like a subsubsection, you'll get the number of the subsection it's inside.

1.2.3 Quantifiers

Although LaTeX provides the $\exists$ and $\forall$ symbols, you should *not* use them, unless you're actually writing a logic thesis and they appear as part of symbolic sentences, or something like that. If you want to say that there is some $x \in \mathbb{R}$ such that blah blah blah, then actually **write out** "there is some $x \in \mathbb{R}$ such that…," not "$\exists x \in \mathbb{R}$ s.t.…."

1.2.4 Punctuating Mathematics

Don't forget that you need to punctuate math. All of the math in `samplethesis.tex` and `latextips.tex`, whether in-text or displayed, includes punctuation (period, comma, or whatever) when appropriate. For displayed math, make sure the punctuation is **inside** the math mode delimiters; LaTeX is pretty smart and will (usually) do a good job in putting the punctuation in the right place, with the right amount of spacing. For in-line math, it can be *either* inside or outside math mode; but if it's outside, you can't leave a space between the ending `$` or `\)` and the punctuation.[4]

1.2.5 Spacing

LaTeX is generally very good about getting spacing correct. For example, it puts a more space after a period than it does after a comma or between words, regardless of how many times you hit the space bar. However, if you only want a regular space after a period (say, because you're using an abbreviation), put a backslash just before the space. For example,

```
Serre et al. proved
```

prints as "Serre et al. proved", whereas

```
Serre et al.\ proved
```

prints as "Serre et al. proved".

Meanwhile, to prevent LaTeX from breaking a line between two words, use a tilde `~` instead of a space. For example, a linebreak in the middle of a reference to, say, Section 1.2 just looks weird. Instead, I've been using tildes whenever I make a `\ref` reference to Chapters, Sections, equations, etc., so as to force LaTex to print such references as Section 1.2. Note that the spacing between words on the last few lines was a little wider than usual; that's LaTeX rearranging things to avoid the forbidden linebreak.

---

[4]Note that this paragraph contains both *italicized* and **boldfaced** text, produced with `\emph` (short for "emphasis") and `\textbf`, respectively. Although you can also use `\textit` to get italics, it is usually better to use `\emph`. For example, using `\emph` inside a theorem environment, where the surrounding text is already italicized, makes the following text **non**-italicized, so that it actually stands out as intended; look at Proposition 2.1.2.d of `samplethesis.tex`. Other related commands include `\texttt` for `typewriter text` (useful for displaying program code) and `\textsl` for *slanted text*.

FYI, in case you see it somewhere: there are also legal but older-style LaTeX commands `\bf`, `\em`, `\tt`, `\sl`, etc., but they are a little clunkier and are used slightly differently. Still, `\textbf{hello world}` and `{\bf hello world}` both produce **hello world**, for example.

Week of	Sections	Topics and Comments
Feb 2	13.5, 13.6, 14.1	Lines, Planes, Quadrics, Vector-valued Functions
Feb 9	14.2–14.4	Calculus for Vector-valued Functions
Feb 16	15.1, 15.2	Multivariable Functions, Limits **FIRST MIDTERM: Wednesday, Feb 18**
Feb 23	15.3–15.5	Partial Derivatives, Chain Rule
Mar 2	15.6, 15.7	Gradient, Extreme Values
Mar 9	15.8, 16.1, 16.2	Lagrange Multipliers, Double Integrals
Mar 16	—	(Spring Break; no classes)

Table 1.1: A portion of a Math 13 syllabus

1.2.6 Tables

If you need a table in your thesis, you can make one with the `tabular` environment. I'll put one right here. (Well, I'm *typing* it here, but LaTeX decided to put it at the top of this page; more on that in a second.) The letters `l`, `c`, and `r` inside the brackets right after `\begin{tabular}` specify whether each particular column of text is left-justified, centered, or right-justified. The | symbols, of course, say where the vertical lines go.

Note that because I put Table 1.1 inside a `table` environment, LaTeX automatically gave it a table number, so it would show up in the list of tables (if I had a `\listoftables` command in the front matter; see Section B.2), and I can refer to it with `\ref`, as I did earlier in this sentence. In addition, I was able to give it a caption. The `table` environment also turns the table into a "floating object", meaning that LaTeX will put it where it thinks best, which will probably be in a different place or even on a different page. For example, Table 1.1 appears at the top of the page, even though I typed the code that created the table immediately after the second sentence of this section. You can (try to) force LaTeX to put the table in a particular place by putting an option right after that `\begin{table}` command: `[h]` to put it right here, `[b]` to put it at the bottom of the page, `[t]` to put it at the top of the page, or `[p]` to put it on a different page. (But if the table simply won't fit where you want it, LaTeX will just do the best it can.)

If you use the `tabular` environment without the `table` environment, the table will appear just where you put it, but with no label, caption, or appearance in a list of tables, like this:

x	$(-\infty, -1)$	$(-1, 1/\sqrt[3]{2})$	$(1/\sqrt[3]{2}, \infty)$
$f'(x)$	$+$	$+$	$-$
$f(x)$	$\nearrow$	$\nearrow$	$\searrow$

1.2.7 Figures

If you need figures in your thesis, use the `figure` environment and the `\includegraphics` command. You'll also need to have the command `\usepackage{graphicx}` somewhere in your preamble, so that LaTeX actually *knows* the `\includegraphics` command. (See Section B.1). The `graphicx` package knows a lot of file formats, including .png and .pdf. (Some

graphics utilities produce .eps files (Encapsulated PostScript); depending on your latex implementation, you may need to convert those to .pdf with a utility like epstopdf.) Figure 1.1 gives an example.

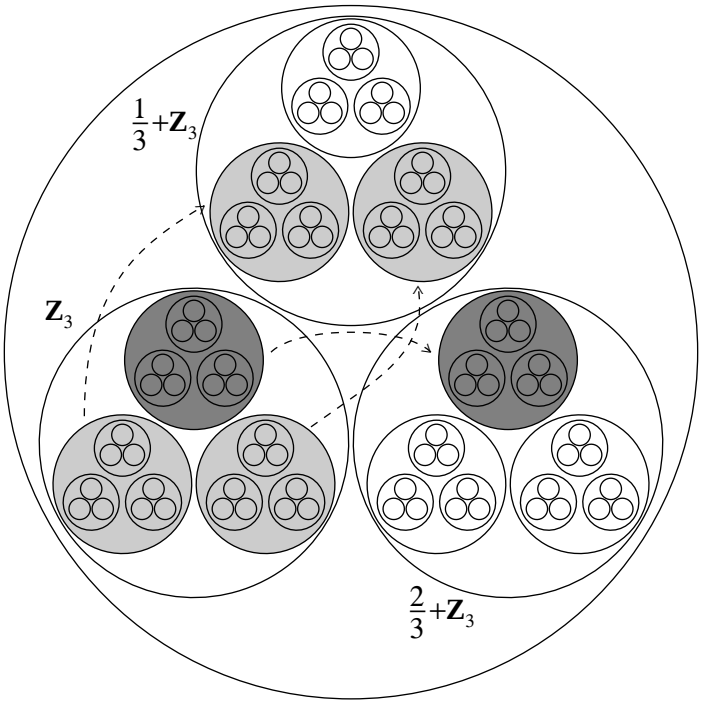

Figure 1.1: An Example of a Figure.

Figures, like tables, are also floating objects. As with the `table` environment, you can use the `[h]`, `[b]`, `[t]`, and `[p]` options with the `figure` environment, if you want. I used the `[h]` option on Figure 1.1, to prod LaTeX into putting it just where I typed it.

I also put Figure A.1 later in this document. Figure A.1 uses the same graphics file as Figure 1.1 (namely, `sample_fig.pdf`). However, for the sake of variety, I had LaTeX rescale the size of Figure A.1 with the `\scalebox` command; specifically, I used `\scalebox{0.8}`, to make the figure 80% of its actual size.

Appendix A

Other little things

A.1 How to make appendices

Just put `\appendix` in the `.tex` file, and presto, all "chapters" that follow are Appendices!

A.2 Comments

As you've probably noticed, lines in the `.tex` file beginning with `%` are comments. LaTeX ignores everything from the `%` symbol until the end of the line. So for one thing, you can leave yourself notes explaining what various technical declarations do, or to highlight where various chapters or sections start. For another thing, if you decide to cut some chunk of your thesis that you've already written, it's probably safer to comment it out (i.e., precede each line of it with a `%` symbol) rather than delete it. That way, if you decide later that you actually want to keep some or all of what you cut, it's easy to recover.

A.3 Linebreaks

LaTeX treats a linebreak (i.e., hitting the Enter key) essentially the same as a space. The only real exception is that two or more consecutive linebreaks (i.e., leaving at least one blank line in the `.tex` file) tells LaTeX to make a paragraph break. I think it's a good thing to have a lot of linebreaks in your `.tex` file. Linebreaks make the file itself much easier to read and to edit, without changing the appearance of the printed version.

However, a clump of two, or seven, or seven hundred consecutive blank lines is treated the same as one. If you want extra vertical space for some reason, use the `\vspace` command. For example, I got the big space just above by putting `\vspace{0.3in}` between two paragraphs.

Incidentally, paragraph breaks (i.e., double linebreaks) are not allowed in the middle of math mode; they will give you an error message. Single linebreaks in math mode are perfectly fine, though — as noted above, they are treated just like spaces.

Oh, by the way, I'm going to put Figure A.1, which I promised back in Subsection 1.2.7, right here. Recall that it's a floating object; LaTeX decided to put it on the next page.

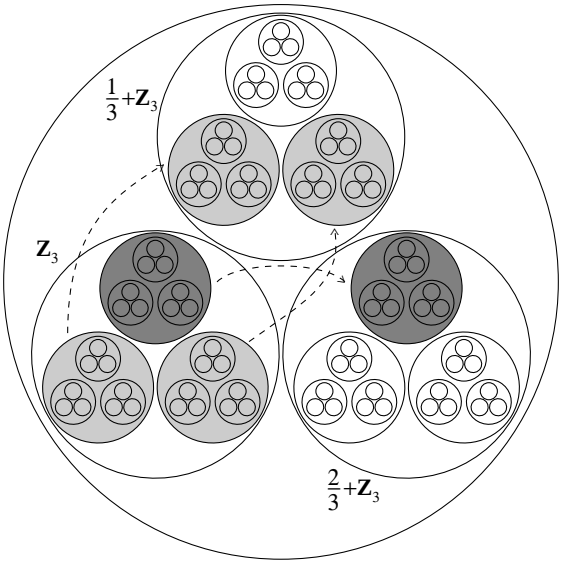

Figure A.1: Another random figure.

A.4 Dashes, Ellipsis, and Quotes

LaTeX handles these punctuation marks a little differently than you might expect.

There are three kinds of dashes: the hyphen -, for intra-word dashes; the en dash –, for number or letter ranges, like "pages 55–70" or "parts (b)–(f)"; and the em dash —, which is, of course, a punctuation mark — the kind that's used in much the same way as a semicolon. Code them in LaTeX with `-`, with `--`, and with `---`, respectively.

An ellipsis is the "dot-dot-dot" punctuation ..., used for partial quotatations. (For example, you will probably need it when writing the "Corrections" portion of your thesis.) Three periods like this... gives the screwy spacing found in the middle of this sentence. Instead, use the `\ldots` command, to get..., as desired.

For double quotes, do **not** use the `"` symbol. (About the only LaTeX use of `"` is for umlauts; `H\"{o}lder` produces Hölder.) Instead, open with two **left** single-quotes `` ` `` (found to the left of the 1 key on most keyboards) and close with two **right** single-quotes `'`. Similarly, for single quotes, use the left and right quotes as appropriate. For example,

```
Cohen notes, ``These `G\"{o}del numbers' are large.''
```

gives: Cohen notes, "These 'Gödel numbers' are large." However, both

```
Cohen notes, ''These 'G\"{o}del numbers' are large.''
```

and

```
Cohen notes, "These 'G\"{o}del numbers' are large."
```

give the ugly: Cohen notes, "These 'Gödel numbers' are large."

I'd recommend you edit your `.tex` file using the `emacs` text-editor. Among its many advantages, `emacs` automatically does what you want when you hit the `"` key; it figures it out whether you need the umlaut symbol, two left quotes, or two right quotes, based on whether the previous character was a backslash, space, or something else. So you actually end up using the `"` *key* after all, even if you don't end up using the `"` *symbol*.

Appendix B

LaTeX File Structure

The `.tex` file has roughly four parts: the **preamble** (setting margins, etc.), the **front matter** (title page, abstract, etc.), the **main matter** (the thesis itself and appendices), and the **back matter** (bibliography and, in the corrected version, list of corrections). In this Appendix, we'll describe the four parts in more detail. In particular, we'll explain how the commands found in `samplethesis.tex` and `thesis.tex` make thesis documents that conform to the College's and the department's regulations.

B.1 Preamble

First, we specify the type of document (`report` class for Amherst math theses), set the font (11 point), and format for two-sided printing.

```
\documentclass[11pt,twoside]{report}
```

If you're planning to print single-sided, then replace `[11pt,twoside]` with `[11pt]`. (See Footnote 1 on page 12 for more information.)

Next, we import certain packages with fancier symbols and theorem styles we'll need.

```
\usepackage{amsmath}
...
```

(Note: if you change anything here, **make sure that the amsmath package is listed BEFORE the amsthm package**.) If you're going to have figures, don't forget to add a line like `\usepackage{graphicx}` here, as noted in Subsection 1.2.7.

After that, we define the theorem types to be used in the thesis.

```
\newtheorem{theorem}{Theorem}[section]
...
```

The `\newtheorem` command takes two *arguments* (in curly brackets): the first is what we'll call that theorem type in the `.tex` file, and the second is the actual text to appear in print. Meanwhile, the *option* (in square brackets) specifies the numbering; `[section]` means that, for example, the first Theorem of Section 4.1 would be numbered 4.1.1, and the first Theorem of Section 4.3 would be 4.3.1. If I had put `[chapter]` instead of `[section]`, then those two

Theorems would be simply 4.1 and 4.2. Or, if I omitted that option entirely, the first Theorem of the **whole thesis** would be numbered 1, the second 2, and so on.

Note that I put the option `[theorem]`, and in a different place, for the other theorem types, to number them as if they were `theorem`s. So if Section 2.3 has a lemma, a theorem, a definition, and then another lemma, they would be numbered as Lemma 2.3.1, Theorem 2.3.2, Definition 2.3.3, and Lemma 2.3.4. If instead you want definitions to be counted separately from the others, change the proclamation line for definition to

```
\newtheorem{definition}{Definition}[section]
```

Then the same set of theorems and definitions would now be labelled Lemma 2.3.1, Theorem 2.3.2, Definition 2.3.1, and Lemma 2.3.3.

Meanwhile, the `\newtheorem*` command is just like the other `\newtheorem` commands, except that the `*` says that there should be no number attached. (Obviously, there is no square-bracketed option to describe the numbering in this case.)

Finally, `\theoremstyle` changes the style of all `\newtheorem` types that follow, at least until the next `\theoremstyle` command. There are three theorem styles to choose from:

1. `plain` style is the most emphatic, with boldface labels and italicized text. It's also the default, so we didn't need to declare it explicitly.

2. `definition` style has boldface labels but regular (upright Roman) text.

3. `remark` style is the least emphatic, with italicized labels and regular text.

I happened not to use `remark` style for any theorem types in `samplethesis.tex`, but you can if you want to. And, of course, there's no requirement that you have to put definitions in `definition` style or remarks in `remark` style.

Next we set margins and things like that; you probably shouldn't mess with these settings:

```
\setlength{\evensidemargin}{0in}
...
```

These are all length variables, which are allowed to be negative but which must have units (even when their value is `0`). LaTeX units include inches (`in`), centimeters (`cm`), points (`pt`), and many others. Most of the commands above set the margins, by giving the location and size of a rectangle on the page where LaTeX is allowed to put text, figures, etc. More precisely, the upper left corner of this rectangle is *one inch plus* `\topmargin` from the top of the page, and *one inch plus* `\oddsidemargin` from the left edge of the page.[1] The rectangle itself is then `\textwidth` wide by `\textheight` high. Meanwhile, `\footnotesep` is the gap just above a footnote, but **footnotes are rarely used in math theses, if ever**.

---

[1] LaTeX ignores `\evensidemargin` if you use the option `[11pt]` instead of `[11pt,twoside]` in the first line of the document. On the other hand, if you **do** use `twoside`, then on even-numbered pages, LaTeX uses `\evensidemargin` instead of `\oddsidemargin`. That's handy, since thesis regulations require a wider margin on the binding side — but if you double-side, the binding alternates between the left and right sides. Please note that whether or not your thesis physically prints on both sides of the page has **nothing** to do with LaTeX's `twoside` option, but instead only with your printer settings.

Immediately after those length settings, two more appear in the preamble; and these are ones that you may actually want to change, depending on the behavior of your printer:

```
\setlength{\voffset}{-0.7in}
\setlength{\hoffset}{0in}
```

Different printers center pages differently, and you can adjust these two lengths to correct for that. So `\voffset` dictates how far to shift the rectangle of text down the paper (or in this case up, since the length is negative), and `\hoffset` is how much to shift it to the right.

Next we start to declare some macros, or keyboard shortcuts. If you want $\mathbb{C}$ to appear, the LaTeX command is `\mathbb{C}`. But if you're going to use that symbol a lot, it'd be easier to to type something shorter like `\C`. So, use `\newcommand` to define whatever macros you want. Incidentally, besides blackboard-bold, `\mathcal` gives a calligraphic font, `\mathfrak` gives a fraktur (Germanic) font, and there are many more.

In addition, although LaTeX has a bunch of built-in math operators (like sin and log), you might want to define your own. If so, then to get the spacing right and make sure the letters appear in regular type rather than italics, use the `\DeclareMathOperator` command.

```
\newcommand{\C}{{\mathbb{C}}}
...
\DeclareMathOperator{\divop}{div}
```

You can name your macros and math operators whatever you want, as long as they begin with a backslash and don't conflict with other commands or macros.[2]

Finally, `samplethesis.tex` also includes the technical line `\newcounter{bean}`, but it doesn't appear in `thesis.tex`, because you won't need it unless you use a fancy enviroment like `list` in your thesis, as I did in Proposition 2.1.2 and Lemma 3.2.2 of `samplethesis.tex`.[3]

B.2 Front Matter

The preamble is now done. It's time to start the actual document, so we have:

```
\begin{document}
```

(There is also an `\end{document}` command at the end of the file.) Then we need to declare double-spacing (`\baselineskip` is how much space to leave between lines), and, at least for the front matter, have page numbers appear as roman numerals.

```
\setlength{\baselineskip}{21pt}
\pagenumbering{roman}
```

---

[2]For example, LaTeX already has the commands `\P` and `\div` to make the symbols ¶ and ÷. So, after I got an error message the first time I tried each of them, I chose new macro names: `\PP` and `\divop`.

[3]The `list` environment requires a user-defined variable called a counter, so I initialized one with the `\newcounter` command. Any old variable name will work, but the joke choice of "bean" is borrowed from George Grätzer, who uses both "bean" and "sheep" as counter names in his LaTeX manuals.

At last, something to actually print: the title page.

```
\begin{titlepage}
...
\end{titlepage}
```

It's not important what the commands here mean.[4] The only things in the title page you should change are:

1. Your name in the **two** obvious places.
2. Your advisor's name in the one obvious place.
3. The thesis title (with linebreaks as you see fit).
4. The due date.
5. The year in the copyright line.

FYI: both `thesis.tex` and `samplethesis.tex` have two-line titles, but you can use any number of lines you deem necessary, within reason. I'd recommend keeping the title short and sweet, though; a one-line title is usually better than a three-line title.

Next is the abstract. Obviously, write something appropriate to your thesis. And use `\chapter*` rather than `\chapter`, since we don't want a chapter number here.

```
\chapter*{Abstract}
This short example thesis meets all of the formatting
...
```

Acknowledgements are similar:

```
\chapter*{Acknowledgements}
Thanks to Mom and Dad for everything.
...
```

LaTeX will automatically generate a table of contents, based on all the `\chapter`, `\section`, etc. commands that appear in the document, with the simple command:

```
\tableofcontents
```

If you have figures or tables in your thesis, it might be helpful to provide lists of them. LaTeX will generate such lists with the `\listoffigures` and `\listoftables` commands. (These are present but commented out in `thesis.tex` and `samplethesis.tex`.)

It's also helpful for the reader if you include a list of notation, giving the page number at which the notation in question is defined. Because your page numbers are certain to change as you edit, mark the first appearance of each symbol in the `.tex` file with the `\label` command, and then get the page number to appear here with the `\pageref` command.

---

[4] `\vspace` makes a vertical space, and `\mbox{}` makes an empty "box" for `\vspace` to push off of. The `\\` commands are linebreaks in the `center` environment, sometimes with an option in square brackets, specifying an extra amount of space to skip. `\vfill` is a vertical "spring" of whitespace, pushing everything above as high as it will go, and everything below as low as it will go; the two springs have equal strength.

```
\chapter*{List of Notation}
\noindent
\hspace{0.8in}
$\PP^1(\Q)$\dotfill p.\ \pageref{not:P1Q}
\hspace{1.5in}
...
```

Next, for technical reasons, we throw in the ugly LaTeX code

```
\makeatletter
\if@twoside \ifodd\value{page}
\clearpage\mbox{}\thispagestyle{empty} \fi \fi
\makeatother
```

to prevent problems with even-versus-odd page numbering (and hence even-versus-odd page margin settings) in certain cases, by inserting an extra blank page if necessary.[5]

Finally, end the current page and switch back to normal (arabic) page numbers.

```
\clearpage
\pagenumbering{arabic}
```

B.3 Main Matter

Start with `\chapter{Introduction}`, or whatever you call the first chapter, and write. Remember that appendices (see Section A.1 for how to make them) are ignored when the department determines grades and honors levels for theses.

B.4 Back Matter

B.4.1 Bibliography

Begin the bibliography with the lines

```
\clearpage
\addcontentsline{toc}{chapter}{\protect\numberline{}{Bibliography}}

\begin{thebibliography}{9}
```

[5] You don't need to know how this bit of code works, but if you're curious, `\if ...\fi` is just like an `if` statement in a programming language; `@twoside` is a boolean variable (set to true if we are using the `twoside` option), and `page` is a counter (i.e., integer variable) storing the current page number; `\value` retrieves its value. So the if statements do nothing unless we are using the `twoside` option and currently on an odd page. In that case, `\clearpage` ends the current page, `\mbox{}` (which is a blank placeholder) starts a new one, and `\thispagestyle{empty}` prevents the new page from displaying a page number. Meanwhile, the `\makeatletter` command makes LaTeX read `@` as a letter, i.e., a legal character in a variable name; then when we're done, `\makeatother` changes it back to its usual status as an "other" symbol, like `\` or `$`.

If you have ten or more bibliography items, change the `9` to `99`.[6] Then list all works cited.

It should be pretty easy to understand how the bibliography works if you look at the `samplethesis.tex` file. The lines

```
\bibitem{MS1}
P.~Morton and J.~Silverman,
Rational periodic points of rational functions,
\emph{Inter. Math. Res. Notices} \textbf{2} (1994), 97--110.
```

make a bibliography entry that you can cite in the main text with `\cite{MS1}`. Note the different styles used for papers versus books (especially italics and publication years); follow those styles in your own bibliography. Finally, end the bibliography with

```
\end{thebibliography}
```

B.4.2 List of Corrections

A few weeks after theses are due, you may hand in a corrected version, along with a list of the corrections you made. (So this final section will **not** be part of the original thesis document you hand in.) Just start a new unnumbered chapter (*after* the bibliography) with

```
\chapter*{Corrections}
```

and begin with, "When originally submitted, this honors thesis contained some errors which have been corrected in the current version. Here is a list of the errors that were corrected." (The `\chapter*` command and the two sentences are present but commented out in `thesis.tex`; so if you started from `thesis.tex`, you can just uncomment them.)

Then list the changes; the `description` environment is good for this. You can mention *really* minor changes (like spelling) with a rough count, but no specifics. However, any other changes (which should still be fairly minor) should be listed explicitly. Refer to page numbers and line numbers from the *corrected* version, since any future reader will probably be reading from the library copy, which will be the corrected version. The example in `samplethesis.tex` should be self-explanatory.

The figure `sample_fig.eps` originally appeared in J. Benedetto and R. Benedetto, A wavelet theory for local fields and related groups, *J. Geom. Anal.* **14** (2004), 423–456. Large portions of the text of the accompanying `samplethesis.tex` document were taken from R. Benedetto, Preperiodic points of polynomials over global fields, *J. Reine Angew. Math.* **608** (2007), 123–153. Many thanks to David Cox for suggestions and improvements to `samplethesis.tex` and to this `latextips.tex` document.

[6] To explain the code here, first `\clearpage` ends the previous page, so that the correct page number appears in the contents when we invoke `\addcontentsline`, which is needed because LaTeX doesn't want to list the bibliography in the table of contents (since it's an unnumbered chapter, like the Abstract). The string `toc` appears because the table of contents filename ends with `.toc`, `chapter` makes the desired text appear like a chapter in the contents, and `\numberline{}` leaves a blank chapter number. The technical command `\protect` ensures that `\numberline` behaves as it should even though we're having LaTeX copy it into another file.

Meanwhile, the `9` after `\begin{thebibliography}` just means LaTeX should leave enough room for one-digit labels for the bibliography entries. It has to do with the width of the symbol `9`, not the actual number 9 itself — if you have more than nine but still fewer than 100 sources, use `99` instead of `9`.