

Math 490 Style Guide

Matthew Kousoulas

August 16, 2022

This document outlines some basic guidelines for writing `.tex` files so that the whole class maintains a uniform look and Amarit is able to compile the whole answer key at the end of the course.

Installation

Included with this guide is a file, `math490.sty`, that contains code meant to replace the preamble of your `.tex` file (explained below). There are two options for installation:

- You can place `math490.sty` in the same directory as your `.tex` files. This is probably the easier option to get started, but if you want to keep problems in multiple directories, you will need to copy `math490.sty` into each directory and possibly update them over the course of the semester.
- You can also place `math490.sty` in the `texmf` directory associated with your \TeX distribution. The advantage of this option is only having a single file to worry about that all of your projects can access. On Unix based systems the `texmf` is usually `~/texmf/tex/latex/math490/math490.sty`. For Windows it depends on whether you have MiKTeX or TeX Live. Amarit is familiar with MiKTeX and can help you with installation.

Preamble

The preamble of a `tex` file is everything before the `\begin{document}` line. It is important that this be uniform across all the final drafts. (The technical reason is that \LaTeX compilation is weird and we're going to use the *subfiles* package to make it sane. Each file you send in will have its preamble replaced with boiler plate code require by *subfiles* and inherit the preamble from a master file Amarit will write.) The first six lines of your final drafts should be the following:

```
\documentclass{article}
\usepackage{math490}
\title{<title>}
\author{<name>}
\begin{document}
\maketitle
```

Note that it's fine for you to put shortcuts in or otherwise modify your preamble for drafts. Just let Amarit or Matt know about these changes so we can release an updated `.sty` file before you make the final draft. As long as the copy Amarit receives to place in the final answer key has the above format everything will be fine.

Naming Scheme

One thing that will be exceptionally helpful for Amarit when he is compiling the answer key is a consistent naming scheme for files. We decided on `<last-name>-<problem-number>.tex` all lowercase. If you are doing multiple exercises as one assignment, keep them in separate files. As an example we have included our first homeworks: `kousoulas-1.2.v.tex` and `mutharu-1.1.iii.tex`. These are also included as a demonstration of `math490.sty` in addition to the chart below.

Environments

We've predefined all of the environments that Riehl uses in the text, theorem, proof, definition, etc. Unfortunately to get the numbering to work there's a bit of a kludge. \LaTeX automatically numbers theorems and other environments as they appear in the text. In order to provide the appropriate reference to the theorem in Riehl's text we need to override this behaviour.

If you want to copy a theorem or definition from the text you need to proceed it with line `\settheorem{<chapter#>}{<section#>}{<theorem#>}` and follow it with `\popthm`. Use this command regardless of whether you are making a lemma, corollary, definition, or any other environment. The two exceptions are exercises which are explained below and proofs which are not numbered.

Exercises

For readability it would be good for us to preface all of our solutions with a copy of the exercise statement. Please try to copy the problem statement verbatim. Similar to the other environment preface it with `\setexercise{<chapter#>}{<section#>}{<theorem#>}`, however there is no need to follow the environment with anything.

Template

We've included a template file that you can use to start all your projects. It contains a lemma, exercise, and proof environment. You only need to change the body text and numbering. You can also change lemma to a different environment to match the text.

Style Conventions

In the interest of having a professional looking answer key at the end of the course, we have a handful of formatting conventions largely drawn from Riehl's text.

First, there are often a variety of notations to choose from for a single concept. You should defer to Riehl's notation first, and if using something beyond the text Dr. Pardue's notation second. For example, both 1_x and id_x are used to denote the identity morphism on a space x . In keeping with Riehl's notation please use 1_x .

Also, a few things to be careful with in general typesetting. Often \LaTeX will have a bunch of commands that all render the same glyph but have different spacing. These correspond to whether the glyph is being used to symbolize an operator, a relation, delimiters, or punctuation. For example $|$ is an operator while \mid is a relation. Similarly, $:$ is an operator while \colon is punctuation and is used for function definitions. In the latter case we have defined a handful of macros for defining functions that automatically use the correct spacing. This is further documented in the \LaTeX symbol guide that we sent out.

Finally, we want to mention that $\$$ as the delimiter for display style math mode is deprecated. It is a primitive in \TeX that was replaced by \lbrack and \rbrack by the **amsmath** packages. (These are aliases for $\begin{equation*}$ and $\end{equation*}$) The latter creates a much more robust environment that properly handles spacing and won't break on complicated commands. **amsmath** also introduced the matching syntax $\l($ and $\r)$ for in-line math, unlike the above these are just aliases for $\$$, so the choice is stylistic.

Command	Output
$x y$	$x y$
$x \mid y$	$x \mid y$
$x : y$	$x : y$
$x \colon y$	$x \colon y$

Commands

We've also defined a bunch of formatting commands to make typesetting easier. These build on the shortcuts that Dr. Pardue has provided with his own \LaTeX examples. Included below is a table of commands with examples of their use. We've done our best to include everything that people regularly use. If there is a shortcut that we've missed and you would like to use, let one of us know and we will update the `.sty` file and redistribute it as necessary.

Examples

To help use these we have a table demonstrating the use of some of the commands. We've only put in a representative sample. There are a lot more in the `.sty` available to use. Please look it over and feel free to make suggestions. Also note that with the exception of the category names, these shortcuts are only available in math mode.

Command	Output	Notes
<code>\Top</code>	Top	works in text mode and math mode
<code>\Setp</code>	Set_*	works only in math mode
<code>\Mod{R}</code>	Mod_R	the subscript is the issue
<code>\func{f}{A}{B}</code>	$f: A \rightarrow B$	functions
<code>\func{\inv{f}}{PB}{PA}</code>	$f^{-1}: PB \rightarrow PA$	inverse functions
<code>\mono{g}{A}{B}</code>	$g: A/\ker f \rightarrow B$	monomorphisms
<code>\epic{h}{A}{B}</code>	$h: A \twoheadrightarrow \text{range}(A)$	epimorphisms
<code>\Func{F}{\sc^{\text{op}}}{\cat{D}}</code>	$F: \mathbf{C}^{\text{op}} \Rightarrow \mathbf{D}$	Riehl uses doubled arrows for functors
<code>\incl{io}{S^1}{\RR^2}</code>	$\iota: S^1 \hookrightarrow \mathbb{R}^2$	inclusions
<code>\idfunc{NN}</code>	$1_{\mathbb{N}}: \mathbb{N} \rightarrow \mathbb{N}$	identities
<code>\oper{+}{\RR}</code>	$+: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$	operations
<code>\nper{\norm{\cdot}}{\CC}{n}</code>	$\ \cdot\ : \mathbb{C}^n \rightarrow \mathbb{C}$	n-ary operations
<code>\de\defeq\recip{ep}</code>	$\delta := \frac{1}{\epsilon}$	
<code>\ZZ\fps{x}</code>	$\mathbb{Z}[[x]]$	automatically inserts <code>\left</code> & <code>\right</code>
<code>2\isinvb3</code>	$2 \in 3$	everyone's favourite relation
<code>\qty{\frac{\len f}{\rtwo}}</code>	$\left(\frac{\text{len } f}{\sqrt{2}}\right)$	
<code>\inlnmat{al&\be\\gm&\de}</code>	$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$	useful for typesetting permutations
<code>\angl{\lst{x}{n}}</code>	$\langle x_1, \dots, x_n \rangle$	optional argument changes the first index
<code>\fm{y}{\nil}</code>	$\{y_i\}_{i \in \emptyset}$	empty list is the best list

Packages

One of the limitations of using subfiles to compile the final project is that we have to all be using the same set of packages for the whole document. We've tried to include an expansive list of packages from the start to give everyone flexibility while still avoiding conflicts and packages that don't come with both MiKTeX and TeX Live. Here is a list of the packages we've included along with a brief description:

mathtools is a wrapper for *amsmath* that fixes several problems and adds some useful functionality.

xypic is a simple drawing library suited to typesetting commutative diagrams. It is considerably less powerful than *PGF/TikZ* which some of you use, but it is more than enough to typeset every diagram in Riehl's book.

enumitem allows for customising the enumerate, itemize, and description environments. Don't go crazy here, the goal is to match the look of the text. We've included a common example.

array updates the implementation of the array and tabular environments making them extensible and more robust.

amssymb provides a handful of extra symbols for typesetting math.

graphicx provides the `\scalebox` command and allows embedding of images.

If you look at the `.sty` file you will see we have included a handful of other packages. You shouldn't need to interact with these, but we've included a description for the sake of completeness.

ntheorem is a replacement for *amsthm* allowing more flexible typesetting. (Matt couldn't figure out how to get *amsthm* to give him output that looked like Riehl's textbook.)

xspace is a very simple library that makes some of the shortcuts simpler to use. It adds a context aware trailing space to text mode shortcuts. See the code for details.

fontenc is included along with a bunch of other font packages to match the look of Riehl's text. As best as Matt can tell the text uses Nimbus Roman for the body text. Some of the spacing is off which annoys him to no end, but he can learn to cope.

geometry handles the page margins.

Documentation for all of these packages can be found on CTAN, the repository for \TeX packages. We've also included documentation in a separate archive.