Mathematics

Senior Thesis Handbook

2019

2019 Edition; Revision 1.0. (September 10, 2018)

Contents

List of Figures

List of Tables

Acknowledgments

The Harvey Mudd College mathematics thesis handbook is based on the work of many thesis coordinators *and* Clinic directors (as some parts of the handbook are based on similar parts of the Clinic handbook).

Substantial contributions to this document were made by (in alphabetical order) Andrew J. Bernoff, Claire M. Connelly, Lisette de Pillis, Weiqing Gu, Kerry Karukstis, Nicholas Pippenger, Michael Raugh, Barbara Schade, Lesley Ward.

Claire Connelly assembled the original version of this document during the summer of 2008.

Calendar

In addition to the tasks specified in this calendar, you will be expected to give a number of talks about the progress of your research to other thesis students during both semester, generally during the time slot scheduled for thesis. Talks will be scheduled by the Thesis Coordinator and the schedule will be made available to you on the thesis website or by some other means.

Important Note

The following table shows due dates as scheduled; check the shared Google calendar for the latest thesis information.

Date	Description
September 11 (Tue) 11:00 A.M.	First class meeting; Senior Research Expectations form signed and turned in
September 25 (Tue) 11:00 A.M.	Deadline for finding second reader
September 25 (Tue) 11:00 A.M.	Thesis webpage set up
November 27 (Tue) 5:00 P.M.	Midyear report first draft due to advisor and second reader (LaTeX or PDF; check with your advisor)
December 7 (Fri) 5:00 P.M.	Penultimate midyear report draft due to advisor and second reader
December 14 (Fri) 5:00 P.M.	Final PDF version of midyear report due to advisor, second reader, thesis coordinator
January 29 (Tue) 11:00 A.M.	Dates for public presentation of talk or poster due to thesis advisor and thesis coordinator
March 28 (Fri) 5:00 P.M.	First draft of final report due to advisor and second reader(s)
April 19 (Fri) 5:00 P.M.	Second draft of final report due to advisor (optionally, to reader as well)

continued on next page

Date	Description
April 22 (Mon) 5:00 P.M.	Draft of Presentation Days Poster Due to Advisor
April 26 (Fri) 10:00 A.M.	Final Poster Completed and Submitted for Printing
April 29 (Mon) 11:00 A.M.	Final Report and All Other Project Deliverables Due
May 3 (Fri)	Alumni Weekend Poster Session
May 6 – May 8 (Mon–Wed)	Presentation Days
May 10 (Fri) 5:00 P.M.	Final Work Products Archived Electronically

Contact Information

Computing Help

Contact the department's information technology analyst at `mreeves@hmc.edu`.

People

Name	Office	Extension	E-Mail Address
Lisette de Pillis, Thesis Coordinator	Shanahan 3426/3418	18975	depillis@math.hmc.edu
Lisette de Pillis, Chair	Shanahan 3426/3418	18975/18023	depillis@math.hmc.edu
Jocelyn Olds-McSpadden	Shanahan 3428	18023	jocelyn@math.hmc.edu
Molly Reeves, Information Technology Analyst	Shanahan 2404	18754	mreeves@hmc.edu
CIS A/V Services	Sprague 5	77777	av-request@hmc.edu

Facilities

Name	Room Number	Notes
Math Reading Room/Library	Sprague 3	
Math Workroom	TLB 3433	Networked copier; color laser printer; fax machine: (909) 621-8366

Chapter 1

The Senior Thesis Program

The Harvey Mudd College community places a high value on helping students to develop research skills that they will carry into their future careers in academia, in industry, or elsewhere.

To foster these skills, the Department of Mathematics provides two options you can choose for your "capstone research experience": senior thesis and Clinic.

Both programs allow you to experience the rigors of scientific and mathematical research—constructing a unique research problem; combing the literature for relevant work; clearly stating one or more hypotheses and developing proofs or computational programs to test those hypotheses; and, finally, writing up the results in a professional form and giving both a formal talk and participating in a poster session to communicate your results.

The Clinic program gives you the chance to work as part of a team on a problem posed by an outside institution, often a company or nonprofit organization.

Whereas Clinic projects are generally centered on solving a problem at the direction of a client, thesis projects often involve "pure" research problems—the exploration of ideas under your own direction or under the direction of a faculty supervisor.

1.1 Objectives

The primary goal of our thesis program is to give you the experience of working as a professional research mathematician. Meeting that goal requires you to achieve two essential objectives during the course of your

project:

1. Acquiring an in-depth familiarity with some area of mathematics.
2. Presenting your work at a professional level in both written and oral forms.

Although some research problems may prove to be intractable, we hope that you will also be able to achieve a third goal,

3. Making an original contribution to mathematical research.

1.2 Thesis Project Requirements

Thesis projects generally take place during the fall and spring semesters of your senior year. A successful project will meet all of the following requirements by the end of this period:

- Finding an advisor for your project, either a Mudd mathematics professor or someone from another department or even another school. (Spring semester of your junior year.)
- Signing and turning in the Senior Research Expectations form.[1] (Due September 11 (Tue) 11:00 A.M.)
- Selecting one or more *second readers* to advise you on your project (a Mudd mathematics professor if your advisor isn't from our department, or anyone qualified if your advisor is one of ours). (By September 25 (Tue) 11:00 A.M.)
- Creating and maintaining a simple webpage that contains a description (abstract) of your project. (By September 25 (Tue) 11:00 A.M.; see Appendix E for details.)
- Meeting regularly with your faculty advisor.
- Making consistent progress on your thesis throughout the school year.
- Attending and participating fully in the senior-thesis seminar, Math 197.

[1] Available from https://www.math.hmc.edu/seniorthesis/guidelines/sre-expectations.pdf

- Completing a thorough review of the scientific literature germane to your research project.
- Keeping a notebook recording your progress on the research (see Appendix D; Purrington (2011)).
- Submitting a first draft of your midyear report, documenting your progress to date, to your advisor and second reader (November 27 (Tue) 5:00 P.M.), and a second draft back to them with changes (December 7 (Fri) 5:00 P.M.). (May be PDF or LaTeX source code; check with your advisor and reader.) See Section 3.1.
- Submitting a PDF of the final version of your midyear report to your advisor, second reader, and the thesis coordinator (December 14 (Fri) 5:00 P.M.). See Section 3.1.
- Submitting a *first draft* of your final thesis report to your advisor, second reader, and Molly Reeves (March 28 (Fri) 5:00 P.M.). The format depends on the recipient's preferences, but you must submit LaTeX source code to Molly Reeves.

  See Sections 3.1, 5.6.2 and 5.7 for how to submit your draft and final thesis.)
- Submitting a *second draft* of your thesis report to your advisor, who may also have you submit it to your second reader for additional feedback (April 19 (Fri) 5:00 P.M.).
- Submitting the final, completed version of your thesis report, including all changes suggested by your advisor, second reader, and other reviewers (April 29 (Mon) 11:00 A.M.).
- Submitting the source code for your thesis and other supporting materials in the proper formats for archiving (due May 10 (Fri) 5:00 P.M.; see Chapter 5 for details).
- Creating a poster presenting the key ideas in your thesis (draft to your advisor by April 22 (Mon) 5:00 P.M.; final version for printing by April 26 (Fri) 10:00 A.M.).
- Presenting your poster during the Alumni Weekend poster session (May 3 (Fri)).

- Making both oral and poster presentations as laid out in the thesis calendar and presenting your work during the college's Presentations Days at the end of the spring semester (May 6 – May 8 (Mon–Wed)).
- Making an oral or poster presentation on your research at a mathematics or other scientific conference (see Section 1.4).

Many of these requirements have specific deadlines associated with them; the Calendar (page 1) lists the dates for this calendar year or check with the thesis coordinator.

1.2.1 Double Majors and Interdisciplinary Theses

If you are a double major, you may choose to have a thesis advisor in another department but still participate in the mathematics senior-thesis program. You may also choose to complete a thesis in an external department and petition to substitute it for the math department's requirements (see Section 1.2.2).

In general, this process requires substantial cooperation between the two departments. If you are considering either of these options, we strongly urge you to talk to the thesis coordinators for *both* departments, as well as becoming familiar with the particular requirements of each department.

If your thesis is interdisciplinary in nature, it must still contain significant mathematical content to satisfy the requirements of the mathematics major. In particular, *your midyear report and final must include at least one separate chapter dedicated to detailing the mathematical aspects of your research*. A member of the HMC mathematics department (typically your second reader) and the mathematics thesis coordinator must approve the mathematical content of both the midyear report and the final thesis.

Joint Math–Physics majors can choose to fulfill their capstone requirements with either a thesis done in either the mathematics or physics departments, and do not need to petition for approval.

1.2.2 Substituting a Thesis from Another Department

If you want to fulfill the thesis requirements for the mathematics major with work completed in another department under the supervision of a faculty advisor from that department, you must meet the following requirements:

1. You must obtain, via petition, the approval of the Mathematics Department Curriculum Committee (MDCC) by *no later than the spring*

semester of your junior year. Your petition to the MDCC should include an explanation of the anticipated mathematical content of your thesis.

Note that an approved petition does not guarantee that your thesis will be accepted as a mathematics capstone, since your final work must satisfy the department's requirements for significant mathematical content to qualify.

2. Enroll in Math 197 and complete its requirements, including attending class, giving presentations, and submitting drafts and final reports as detailed in the *Mathematics Senior Thesis Handbook* (this document).

   Participation in the thesis seminar is a vital part of the mathematics capstone experience, providing you with exposure to the work of your fellow mathematics thesis students and giving you the opportunity to improve your presentation and writing skills while helping your peers with their own.

3. Have an advisor (or joint advisor) or second reader from the mathematics department. Your mathematics advisor or reader will be primarily responsible for determining whether the mathematical content of your work satisfies the requirements for a mathematics capstone project.

4. Present your research in the Mathematics Thesis Seminar (Math 197) during the fall semester.

5. Submit a midyear report at the end of the fall semester and a final thesis report at the end of the spring semester that satisfies the interdisciplinary thesis requirements for mathematical content stated in Section 1.2.1.

6. Present your research in some public forum during the spring semester (see Section 1.4).

7. Present a talk about your research during Presentation Days (May 6 – May 8 (Mon–Wed)).

8. Submit both printed and electronic (PDF) copies of your final thesis to the mathematics department's thesis coordinator that include the mathematics thesis signature page with all appropriate signatures, as outlined in Chapter 5.

1.3 Your Thesis Committee

Your thesis committee will have at least two members—a *faculty advisor* and a *second reader*. At least one member should be from the Harvey Mudd College mathematics department; the other may be from another department at Mudd or even from another college or university.

Note that we expect that you will have found an advisor *before* you apply for thesis, and that you will have found a second reader early in the fall semester (by September 25 (Tue) 11:00 A.M.).

In special circumstances (see Section 1.2.1), your thesis advisor may be from an external department. In these cases, you *must* identify a second reader from the HMC math department *before* the beginning of the academic year.

1.4 Required External Presentation or Poster Session

During the course of the year you will be expected to give a number of presentations about your topic and your progress to various groups of people, including your fellow thesis students and the college community and general public during Presentation Days at the end of the spring semester.

You are also *required* to give a talk or show a poster at one or more conferences or other professional meetings. Some possible venues are the regional Mathematical Association of America (MAA) meetings in the fall (student talks) or spring (poster session)[2] or the American Mathematical Society (AMS) Pacific Coast Undergraduate Mathematics Conference.[3]

Your advisor will also have good advice on suitable venues for your external presentation.

Chapter 4 talks about creating and making presentations, and will help you prepare for talks in off-campus venues as well as those you give at the college.

In addition to the posters and talks for Presentation Days, there is a poster session on Alumni Weekend that students are expected to attend. If you have a poster from your external presentation, you may use that poster, or revise it to reflect new work since that presentation.

[2] See the MAA's website, http://www.maa.org/, for details on these meetings.

[3] http://www.pcumc-math.org/

1.5 Copyright and Fair Use

Under US copyright law, your thesis is automatically copyrighted by you. We ask you for the right to "publish" your thesis on the department's website as well as on the Scholarship@Claremont site[4] (see Section 5.4).

Copyright cuts both ways, of course, and you must cite resources you rely on properly (see Section 3.7). You also need to be aware of restrictions on the use of significant quotations from others' work, graphs, diagrams, or other illustrations that you have not created yourself but want to include in your document. Although "fair use"[5] might apply to some of these uses, it's generally better to err on the side of caution.

If possible, obtain permission from the copyright holder (usually the author or publisher, but you might need to do some digging). In some cases, you can redraw or adapt a diagram or graph, but you must credit the original source.

Some materials may be available under Creative Commons licenses[6] or "in the public domain", but you should still credit the creator whenever possible (even if the license doesn't require attribution, providing it is always a good thing to do).

Copyright issues are complicated and not always entirely clear—for example, the legal definition of fair use is not so precise that you can be sure that your use falls under those provisions (or, more importantly, that the rights holder might dispute your usage). The Claremont Colleges Library has some resources on its website[7] that should be helpful, as well as experts who specialize in copyright and digital publication issues. We encourage you to take advantage of these resources.

1.6 Publication of Your Thesis Work

If you and your advisor decide that your work could be published, we encourage you to do so. Your advisor can help you select an appropriate journal.

Note that publication of your work does not substitute for an external presentation, although giving a talk or presenting a poster about your paper would.

[4] http://scholarship.claremont.edu

[5] http://libguides.libraries.claremont.edu/copyright-resources/FairUse

[6] https://creativecommons.org/

[7] http://libguides.libraries.claremont.edu/copyright-resources

Publication also raise potential copyright issues[8] affecting your final report—consult with your advisor and the thesis staff to ensure that we all understand the copyright requirements of the particular journal you're publishing in.

1.7 Safety and Medical Conditions

If you have a medical condition such as hypoglycemia, hyperglycemia, diabetes, epilepsy, pregnancy, or any other medical condition which may cause sudden loss of consciousness, please inform your instructor of the condition at the beginning of the semester or as soon as you become aware of the existence of the medical condition.

[8]For example, some journals require you to sign over all rights to your paper; some allow web publication of drafts or other versions not including the final version they publish; some impose an embargo on sharing your work elsewhere.

Chapter 2

Computing Resources

The department provides administrative and computing support for thesis projects through its general computing services (mail, fileservice, etc.). Most such resources are made available over a network connection.

2.1 Computing Support

We have an extensive website devoted to our computing resources, how to find them, how to use them, and how to get help, available at https://www.math.hmc.edu/computing/.

Your primary human source for support when using the department's computers, services, and other computer-related equipment and materials (including LaTeX!) is our information technology analyst, Molly Reeves.

If you have any questions or concerns, please feel free to drop by the information technology analyst's office, Shanahan 2404, or send e-mail to mreeves@hmc.edu.

2.2 Accounts

Department machines, including any Linux or Mac systems in Clinic labs, use your math-cluster login name and password for access. You will fill out the necessary forms for accounts during the initial Clinic meeting, but you can (and should) read the appropriate-use policy (also available in a somewhat different form at https://www.math.hmc.edu/computing/policies/). The account-request form is also on the website.

2.3 Computing Equipment

We expect that you will be using your own or college-owned shared computers for the bulk of your work. The department also has some compute servers available for use.

You are also welcome to use department printers on the third floor of Sprague (math Clinic area).

2.4 Shared Machines

As math-account holders, you are also welcome to use other departmental computers. You can access these machines remotely by logging in using the Secure Shell (SSH) protocol. See the computing website for more details.

The department currently has two compute servers that can be used by Clinic teams:

hex Sixteen-core (eight 2 GHz dual-core AMD Opteron 870 processors) parallel system. 32 GB of RAM.

pex 32-core (four 2.4 GHz, eight-core AMD Opteron 6136 processors) parallel system. 128 GB of RAM.

2.5 Storage (Disk) Space

You have a home directory in `/home/students/hmc_`*`gradyear`*`/`*`user-name`* for storing your documents, code repositories, and so forth. These home directories are mounted via NFS (Network File System) on all of the department's machines, so your files can be accessed from any department machine that you can log into (i.e., the machines listed in Section 2.4).

Even if you choose to do most of the work for your thesis on your own machine or on systems provided by another department or organization,

- The webpage describing your thesis work (see Appendix E) that you are expected to maintain is served from a directory within your home directory.
- Your final thesis report and other materials will also end up in your home directory, which will be archived. (See Section 5.7).

We strongly encourage you to set up some form of revision-control system (e.g., Subversion or Git) with the repository stored in a subdirectory of your home directory. You can then work on your code (programming or LaTeX) while avoiding accidental deletions, enabling you to revert changes, and allowing you to experiment without losing your work. See Section 3.8.2 for more details.

2.6 Backups

Your home directories and other NFS-mounted directories on department machines are backed up as part of the department's regular backup cycle. Our tape-backup system runs nightly and goes through a complete set of tapes in roughly a month.

We strongly encourage you to maintain your own backups, as well, either of your whole machine or just of your thesis work.

2.7 Electronic Mail

Your account includes an e-mail address, *username*`@math.hmc.edu`. These addresses are set up to forward to your default (`@hmc.edu`) HMC mail address.

2.7.1 Mailing Lists

CIS sets up a mailing list called `math-197-l` with the addresses of all the students registered for the senior-thesis seminar (Math 197) at the start of the fall semester. We use this list to send important messages to all students enrolled in the thesis seminar, so if you join the class later on, make sure that you are added to the list.

Some other lists that you might be interested in include

`linux-l@g.hmc.edu` The `linux-l` list is for discussing the use of Linux at Mudd. It's also a good place to ask for help from a collection of professors, systems staff, and fellow students.

`latex-l@g.hmc.edu` The `latex-l` list is a place to ask (and answer) questions about using LaTeX. As with the Linux list, there's a diverse group of people who can help you out.

We have instructions for subscribing to these lists available on the department's website, at https://www.math.hmc.edu/computing/support/email/mailing-lists/.

Chapter 3

Writing Your Mathematics Thesis

The thesis process requires you to write two documents during the course of the year.

3.1 Written Reports

The first document is a midyear report, due at the end of the fall semester. This document serves several roles: it is a means of updating your advisor on your results to date; an initial draft of your final report; and a gentle introduction to the tools, skills, and outlook required when writing professional reports.

We strongly encourage you to share your LaTeX code with your advisor and second reader (as well as any peers you're working with), which will both ensure that your code will typeset on someone else's system and allow them to give you feedback on your code. An ideal way of sharing your code is to use a version-control system such as Git or Subversion and a code-hosting service such as GitHub or Bitbucket (see Section 3.8.2).

Midyear Report Due Dates

November 27 (Tue) 5:00 P.M. First draft of your midyear report is due to your advisor and second reader.

December 7 (Fri) 5:00 P.M. Second draft addressing their comments due to advisor and reader.

December 14 (Fri) 5:00 P.M. Final version of your midyear report due to your advisor, your second reader, *and the thesis coordinator*.

Submit your LaTeX code or PDF to your advisor and reader (whichever they prefer); e-mail a PDF file to the thesis coordinator.

Although your midyear report serves as a draft of your final thesis, that document will probably still require a great deal of additional work to complete, so you will need to plan ahead to avoid collisions with all of the other activities (final exams, other papers and projects) that typically crowd up at the end of the year for graduating seniors.

In particular, your thesis will be expected to conform to the requirements spelled out in this chapter.

Final Report Due Dates

March 28 (Fri) 5:00 P.M. First draft of your final report is due to your advisor and second reader.

April 19 (Fri) 5:00 P.M. Second draft with revisions due to advisor and reader.

April 29 (Mon) 11:00 A.M. The final version of your thesis is due to your advisor, second reader, *and the thesis coordinator*.

The PDF you send to the thesis coordinator may be used by the department in determining end-of-year awards.

May 10 (Fri) 5:00 P.M. Final PDF, LaTeX source code, poster, data or source code, and other materials must be archived in your math cluster account. (See Chapter 5.)

3.2 Reference Books on Writing and LaTeX

Writing is hard, skilled work, and LaTeX's syntax can be daunting for the newcomer. Obtaining some well-written, respected references to call on when questions arise is a worthwhile investment.

3.2.1 On Writing

We highly recommend three books for guidance on various levels of the writing process.

Lyn Dupré's *BUGS in Writing: A Guide to Debugging Your Prose* (1998) provides an excellent guide to technical writing, including motivations, terminology, formatting (including footnotes, tables, figures, and much, much more), and word choice. Dupré has edited many major works in computing and other sciences, and is acutely aware of the special needs of scientific writing.

Jean-Luc Doumont's *Trees, Maps, and Theorems: Effective Communication for Rational Minds* (2009), has solid advice on structuring documents such as your report, creating presentations and presenting them in front of an audience, and some advice on other topics, such as effective e-mail messages and poster design.

The last book is recommended more as a reference and a stylistic tie breaker than as an immediate influence. *The Chicago Manual of Style* (University of Chicago Press, 2003) is the ultimate distillation of the stylistic rules for the University of Chicago Press, one of the oldest and largest academic presses in the United States. *Chicago* has the down-and-dirty details on every last-minute quibble about how to spell words, capitalize names and titles, number sections, lay out figures and tables, and almost anything else you can think of (and a good deal more you've probably never considered). Note that we have used *Chicago* as a guide to designing and organizing the thesis class and sample reports.

3.2.2 On LaTeX

For LaTeX, we recommend George Grätzer's *More Math into LaTeX*[1] (2007), which is a comprehensive introduction to all of the most frequently used aspects of LaTeX, with a special—and unique—emphasis on the use of the American Mathematical Society's AMS-LaTeX packages, which are almost universally used by mathematicians in preparation of articles and books, even for journals not published by the AMS.

Grätzer's book is especially useful for its mathematical typesetting examples. He presents the typeset equations—which can be skimmed through to look for examples similar to your particular problem equation to use as guides—in conjunction with the code used to typeset the examples.

The first part of the book walks you through the process of creating your first LaTeX document. In the appendices you will find tables of useful symbols and other commands, a brief history of TeX and LaTeX, and instructions for installing a TeX system on your Mac OS X or Windows machine.

[1] *More Math into LaTeX* is available on SpringerLink; see the bibliography for a link.

Note that the third edition of Grätzer's book, *Math into LaTeX*,[2] (2000) is also an acceptable guide.

3.2.3 Additional References

Some other books you might want to consult include

- *Mathematics into Type*, Swanson et al. (1999).

  A guide to writing and typesetting mathematics, with a great deal of useful information on how best to compose your mathematical expressions for accurate, readable, and attractive typesetting.

  Available from the AMS as a downloadable PDF. (See bibliography for link.)

- *The LaTeX Companion*, Mittelbach et al. (2004).

  Compiles the documentation for many LaTeX packages and provides code examples and typeset results to help you choose the best package for a particular use. The *Companion* also serves as an introduction to and reference for LaTeX programming, and as such can help you create your own LaTeX packages and document classes, but will also provide almost everything you're likely to need for composing your own commands, environments, and other LaTeX macros.

- *The Elements of Typographic Style*, Bringhurst (1996).

  A guide to the intricacies of typesetting, page layout, typeface design and choice, and how to format almost any textual material.

3.2.4 Obtaining References

BUGS in Writing is relatively inexpensive and is the sort of book that you'll find useful no matter what you go on to do in your post-Mudd career. We strongly encourage you to buy yourself a copy—not only will your thesis benefit, but so will your other writing this year and during the rest of your career.

Similarly, one of Grätzer's books will help you with the technical aspects of writing your thesis and papers or homework or results for other classes now, and will continue to serve as a solid reference for any LaTeX writing

[2]The third edition of *Math into LaTeX* was edited by the department's system administrator, making her a useful resource as well.

you'll do in the future. (We believe that once you've used LaTeX, you'll never want to use a word-processor for anything more complicated than a quick letter or shopping list again.)

Justifying *Chicago* requires a bit more of a commitment to creating long-form documents (such as this one!) in your future career. You might choose to avail yourself of a library or institutional copy rather than buy it now.

The department should have copies of *Bugs*, *MMiL/MiL*, and *Chicago* for reference; check with the department's administrative aide or your advisor if you can't find a copy.

The other books we've mentioned should be available from the Libraries of the Claremont Colleges, bookstores, and other sources.

3.3 Formatting Your Reports with LaTeX

As discussed in Section 1.2, you are expected to use LaTeX to produce your midyear report and final thesis. The department has created a document class, templates, bibliography style, and other material to help you write your thesis. Appendix A provides documentation for the hmcthesis document class, and the rest of this chapter should help you with the rest.

For still more resources on using LaTeX, see Section 3.9.

3.3.1 Report Template

Included in the distribution of the hmcthesis class is a template file for use in formatting your reports. The file is called `hmcthesis-template.tex`, and should be copied to your working directory and modified to include information relevant to you and your thesis project.

The template file contains extensive comments explaining what the included commands do and how they should be filled out. Section A.1 covers the various commands provided by the class file for specifying the title of your report; the names of your advisor(s) and reader(s),[3] and so forth.

3.3.2 The Sample Thesis

The template file gives you the basics to get started, but we also have a sample thesis report that can be typeset with the hmcthesis class to give you an idea of what a mostly filled-out report should look like. The sample

[3]Be sure to use the "official names" that your advisor(s) and reader(s) use in their own publications; see Appendix G for more information and a list.

report contains examples of most commonly used LaTeX commands and environments, along with some tips and tricks for writing LaTeX documents.

In addition to its value as a visual example, the code for the sample document is meant to serve as an example for you to emulate when writing your own LaTeX code. It contains extensive comments, and also shows best-practices layout of various LaTeX structures such as tables, figures, lists, and so forth.

The sample-report code is available from the department's website at https://www.math.hmc.edu/computing/support/tex/sample-report/ in both `tar.gz` (compressed tape-archive files) and ZIP files. That page also includes links to the document as typeset with both the hmcthesis thesis class and the hmcclinic class file; these examples show how the same material can look somewhat different when typeset with a different class file.[4]

Important Note

Don't use the sample thesis/Clinic report as the basis of your document—use the provided template file instead. The sample document loads several packages that you won't need, defines some commands you probably won't use, and so forth.

Otherwise, feel free to crib from the sample report where doing so is useful!

---

[4] As you will probably notice, the differences in the documents are fairly minor, as the hmcclinic and hmcthesis class files are very similar. A more dramatic change can be seen by typesetting an article-length document with both the LaTeX article class and the amsart class created by the American Mathematical Society.

3.4 Getting Started

When you're staring at a blinking cursor, the task of writing your thesis may seem impossible. Your thesis may be the longest and most complex document that you've ever worked on. But tackling the report is a lot easier if you look at it as a series of smaller tasks whose results combine to create the final document.

The best way to write your thesis is to do so incrementally—adding elements as they become available. And, yes, that does mean that writing your report *while* you're doing your research is a (very) good way to work.

Start early in the fall semester by setting up your thesis template, adding in all of the appropriate names and the title of your thesis project. Then, within the first two to three weeks, you can fill in the sections that outline your research problem and approach. You probably have some idea of the scope of your project, maybe even preliminary notes, even at this early stage.

Similarly, you're probably aware of some background publications in the library and on the Internet, so you can write a brief discussion of each reference while starting to compose your (annotated) bibliography (using BibTeX, Section 3.7, of course).

Viewed in this manner, your thesis becomes a cumulative work that is subject to evolutionary improvement at every step of the project. By working on your thesis throughout the course of the year, most of it will already be complete by the time it's due, and you can concentrate on the chapters and sections that detail your research results.

Your faculty advisor can be a great resource while you're writing your thesis. Keep your advisor closely involved as you work on drafts to assist you with matters of style as well as content, and be sure to also solicit comments and suggestions from your reader(s), other interested faculty, and your peers.

3.5 The Structure of Your Thesis

LaTeX provides a series of nesting *structural commands* to specify the structure of a document. By separating and organizing related material, you can make the organization of your report clearer to your readers. A nice side effect of this approach is that it supports the division of a larger document into individual files that can be worked on independently.

The highest level of structural division in your thesis is that of the *front matter*, *main matter*, and *back matter*, which are set in your source code by

the placement of the `\frontmatter`, `\mainmatter`, and `\backmatter` commands. If you're using the template file (Section 3.3.1), these commands are already in the right places, with comments explaining what material belongs in each part.

We will provide a brief summary of the content of each part in the following sections; this material is drawn primarily from *The Chicago Manual of Style* (University of Chicago Press, 2003).

Important Note

Note that the order of these elements matters. If you're not using the template (see Section 3.3.1), be very sure that the order of elements in your thesis matches those in the template or the sample thesis.

If you're not sure, ask.

3.5.1 The Front Matter

The front matter is the material that appears at the—wait for it!—front of your document! If you look at a published book, the front matter generally corresponds to the parts of the book that are numbered using roman numerals.

In the front matter of your thesis, you should have the following elements:

Title Page Formatted for you by the document class.

Copyright Page Formatted for you by the document class.

Abstract A *brief* (one to four paragraphs), high-level description of the main thrust of your report. You want to explain what the problem is, how you went about solving it, and perhaps have a very brief summation of your solution.

Table of Contents, List of Figures, List of Tables These elements are produced by LaTeX's automatic numbering system and their placement is specified using the commands `\tableofcontents`, `\listoffigures`, and `\listoftables`, respectively. (See also Section 3.6.)

Important Note

The ordering of these commands is important—the table of contents should appear first, followed by an optional list of figures and an optional list of tables.

If you have other elements that you think should be similarly itemized, talk to your advisor about how to do so.

Acknowledgments A place to thank individuals, organizations, or even things that were particularly helpful in completing your project. You should also thank anyone who read your drafts and made suggestions for improving them (advisors, readers, others).

3.5.2 The Main Matter

As its name suggests, the main matter is where the real content of your report is found. For a report of this significance, we expect to see several chapters dealing with the key elements of your project.

3.5.2.1 The Introduction

You should start with an introductory chapter (usually just called "Introduction") that explains the problem, talks briefly about previous research, and then sketches out the remainder of your report with references to the in-depth discussions.

If at all possible, your introduction should be written using *non-technical* language. Write for an intended audience of *non*–math-major HMC seniors, without assuming any specific knowledge of the area of your thesis work.

3.5.2.2 Other Content

You should then have chapters covering topics such as

- The theoretical background of the problem, previously done related research, and how it applies to your problem
- Your theories and approach to solving the problem
- How you implemented your solution
- The result of applying your solution to the problem
- Other related material

3.5.2.3 The Conclusion

The most important chapter in your thesis is the conclusion. Many readers will only read your introduction and conclusion. Some might even skip the introduction, which means that your conclusion is even more important.

Your conclusion shouldn't be just another summary of the whole process. You need to say what it is that you accomplished over the course of the year, and why what you did was meaningful. You may want to discuss new insights you've had into the problem as a result of following the approach that you did. You might even want to include some criticisms of your work in light of what you've learned.

3.5.2.4 Future Work

You should talk about some of the possibilities for future research based on the work that you've done. For your midyear report, this section will be the place to talk about the work that is currently in progress and where you expect it to lead. For your final thesis, this section might grow to be its own chapter, outlining several possible future approaches to your problem and why they might be worth pursuing.

3.5.2.5 Appendices

Appendices are extra bits of information that aren't necessary for understanding the key ideas of your report, but that may be interesting for people who are especially interested in a particular topic.

Good uses for appendices are in-depth discussions of how a particular theory works, why an algorithm was constructed one way and not another, and other background material about your project or about a problem that may provide some additional insight into your project.

3.5.2.6 Source Code and Algorithms

You might be thinking that an appendix would be the perfect place to include the thousands of lines of MATLAB, C++, or Java code you wrote to implement a solution to your project's problem. It isn't.

By and large, people don't read printed source code. And if they are interested in your code, they probably want to run it, not just read it, and they certainly won't want to have to type it all into a computer.[5]

[5]Note that the typesetting process can often make cutting and pasting from a PDF version of your thesis impossible, too, as some fonts and font encodings can result in glyph

```
\begin{figure}
  \centering
  Content Appears Here
  \caption[An Example Figure]{An Example Figure.}
  \label{fig:example-figure}
\end{figure}
```

Figure 3.1 Code for an Example Figure.

If you have source code that you think people will find useful, we can make it available through the website that we will maintain for your thesis. See Section 5.7.1 for more information about this option.

If you want to describe one or more algorithms in detail, especially if they are too complex to easily handle within the normal flow of your implementation and results chapters, you may do so in one or more appendices. But no source code!

3.5.2.7 Figures and Tables

We expect that you will use LaTeX's `figure` and `table` environments to format your figures or tables. The basic form of these constructs is shown in Figures 3.1 and 3.2.

Some things to notice in these examples:

Optional Float Arguments Note the lack of the optional placement argument to the float environments. In general, you should allow LaTeX to put tables where it wants to put them. If you absolutely have to have a float appear in a particular location, then you can add this argument.

In general, floats should appear at the top or bottom of a page; if you have them embedded in the middle of a page, you should think seriously about whether that placement is absolutely necessary for a reader to understand your work.

It's also a good idea to leave details like the placement of floats until very near the point that you're ready to submit your final thesis. Until then, changing other parts of your document may cause everything to reflow and make your specification incorrect.

substitutions that produce corrupt code when pasted into an editor.

```
\begin{table}
  \centering
  \begin{tabular}{lll}
  \toprule
  Header  &  Header  &  Header \\
  \midrule
  Cell    &  Cell    &  Cell   \\
  Cell    &  Cell    &  Cell   \\
  \bottomrule
  \end{tabular}
  \caption[An Example Table]{An Example Table.}
  \label{tab:example-table}
\end{table}
```

Figure 3.2 Code for an Example Table.

Header	Header	Header
Cell	Cell	Cell
Cell	Cell	Cell

Table 3.1 An Example Table.

Optional Argument to `\caption` In both figures and tables, you should provide an optional argument to the `\caption` command. The optional argument is used (when present) in the list of figures or list of tables. Note that here we drop the period from the required caption argument. It's especially important to use the optional argument if your caption is more than just a few words. (In general, lengthy discussions should take place in the text that references the figure rather than in the figure caption, although it's also reasonable to have a title sentence followed by one or two brief sentences describing the content of the float. Consistency is the key—if you have one or two floats with very different captioning, we'll call you on it.)

Use of the booktabs Package In Figure 3.2, our table uses rule commands provided by the booktabs package, which provide horizontal rules as shown in Table 3.1. Additional (and more complex) table examples can be seen in other parts of this document as well as the sample thesis/Clinic report.

Important Note

Do not use the table style with vertical rules shown in some LaTeX books.

3.5.2.8 Image Formats

The department will be typesetting your thesis using pdfLaTeX, which only understands images in PDF, PNG, or JPEG formats.

In addition to using the accepted formats, it's important to understand the difference between *vector* and *raster* or *bitmap* images.

Vector images are encoded such that they can be rescaled with no loss of quality. Examples of vector image formats include PostScript, Encapsulated PostScript (EPS), and PDF.

Raster images are stored as pixels. They generally have a set *resolution*, often specified in *dots per inch* or *dpi*. Because of this set resolution, raster images do not scale up well. For example, a typical image on a website has a resolution of 72 or 96 dpi (screen resolution). In contrast, the standard resolution of a modern laser printer is 1200 dpi.

a. A Raster Image (72 dpi PNG image).

b. A Vector Image (PDF).

Figure 3.3 A low-resolution raster image (a) compared with a scalable vector image (b). Both are the same size, but scale very differently.

Scaling such an image up requires it to attempt to fill a larger space with the same number of pixels. Depending on the scaling method in use, the image will degenerate so that it either looks dotty (like an image in a newspaper) or blurry (as the program tries to fill in the gaps with guesses). Figure 3.3a shows a 72 dpi image scaled up slightly from its original size.

Raster images *can* be used for high-resolution printing, but you generally need to have a high-resolution (minimum of 300 dpi) image to use. Figure 3.3b shows a 600 dpi PDF image scaled up slightly from its original size.[6]

Finally, note that some applications will allow you to save a low-resolution raster image in a vector format. Unfortunately, that generally means that the application is creating a vector "wrapper" around the low-resolution bitmap image.

3.5.2.9 Recommended Graphics Applications

We expect that most of the images you include will be diagrams illustrating some process or graphs or charts showing your results. You may also need to include photographs that document some particular aspect of your project (or, for some projects, may actually be examples of images that are the subject of your investigations).

The following suggestions may help you find an appropriate application;

[6]Note that the resolution issue causes particular problems for screenshots taken of computer applications. If you need to include screenshots, you should explore the possibility of mocking up the screenshot using an application that can produce vector graphics. Alternatively, you may be able to take a screenshot of a large window and reproduce it at a smaller size, but you can still expect to see some blurring when printed.

feel free to consult your advisor or the department's system administrator for additional suggestions.

Diagrams For diagrams, we strongly recommend the use of OmniGraffle Pro, a drawing application specifically designed for creating diagrams. OmniGraffle allows you to save your final images in various formats; you should generally use PDF for the best results.

You may also find the PGF and TikZ LaTeX packages useful for creating diagrams and illustrations right in your reports. This option is particularly useful if your diagram is more easily generated by a script or program rather than drawn by hand.

Graphs Many of the mathematics applications available on the department's computer systems can be used to draw graphs, either as their main focus or as one of many output options. In particular, Maple, Mathematica, MATLAB, R, and SAGE can create graphs. On the Macs, Plot and Abscissa are graphing applications. Grace and Matplotlib are Linux applications for graphing. And, of course, spreadsheet applications, such as Calc, Excel, and Pages, can generate graphs.

As with other diagrams, in general you will want to get vector output of your graphs for inclusion in LaTeX documents. In many cases you will be able to save the graphs directly as PDF, which is the best option. If you cannot generate PDF, you may be able to generate PostScript or Encapsulated PostScript (EPS), which can be converted to PDF. Avoid bitmap formats (BMP, GIF, JPEG, PNG, etc.).

Photographs For the most part, your use of photographs should be fairly basic, and may even be limited to simple color adjustments or cropping. In such cases, we strongly recommend that you use the iPhoto or Preview applications on Mac OS X, which will allow you to do these simple manipulations easily.

More complicated editing may require a dedicated bitmap editing application, such as Photoshop or the GIMP (on Linux).

For photographs, you will want to retain the highest resolution image or the most pixels that you can. Do your cropping without changing the resolution or size of the image, and adjust the image size in your document with the commands provided by the graphics and graphicx packages (Carlisle, 1999).

3.5.3 Back Matter

The back matter comprises elements such as your bibliography, an index, a glossary, a colophon, and so on. Of these, the only one you're likely to bother with is the bibliography, which we cover in Section 3.7.

If you're interested in including one or more of the other commonly seen back-matter components, please talk to your advisor and the department's system administrator about the best way to do so.

3.6 The LaTeX Label and Reference System

LaTeX provides commands that allow you to refer to other parts of your document without keeping track of the numbers that it assigns to various structural elements during the typesetting process.

In addition to the numbering for chapter, sections, and similar structural elements, LaTeX also numbers mathematical equations (when using the `equation` and similar environments); figures, tables, and other floats (see Section 3.5.2.7); and other elements, such as list items and pages.

By using a `\label` command, you can give yourself a textual handle that you can later use to refer to the labeled item using a `\ref` command.

For example, the text "Figure 3.1" is produced by typing

```
Figure~\ref{fig:example-figure-code}
```

In the `figure` environment for that figure, we have a `\caption` command, and at the end of the command (or on the following line without whitespace between them), we have a `\label` command that looks like `\label{fig:example-figure-code}`.

3.6.1 Formatting Labels

In general, we recommend that you format your labels in a consistent, but generic, way, usually in the form *`kind:label text`*. Using `fig` for figures, `tab` for tables, `ch` for chapters, `sec` for sections, `eq` for equations, and so on, means that you can easily distinguish between labels (and, perhaps, have the same label text for both a section or equation and an illustration showing a visual example of what you describe). Some text editors take advantage of these "kind" labels to help you find and insert references, and allow you to limit your search by reference type. There are even LaTeX packages that can automatically fill in the labels for you, formatted correctly.

Object Type	Abbreviation	Reference Example
Chapters	`ch`	`Chapter~\ref{ch:`*thing*`}`
Sections	`sec`	`Section~\ref{sec:`*thing*`}`
Subsections	`sec`	`Section~\ref{sec:`*thing*`}`
Subsubsections	`sec`	`Section~\ref{sec:`*thing*`}`
Parts	`pt`	`Part~\ref{pt:`*thing*`}`
Figures	`fig`	`Figure~\ref{fig:`*thing*`}`
Tables	`tab`	`Table~\ref{tab:`*thing*`}`
Equations	`eq`	`Equation~\ref{eq:`*thing*`}`
Theorems	`thm`	`Theorem~\ref{thm:`*thing*`}`
Lemmas	`lem`	`Theorem~\ref{lem:`*thing*`}`
Definitions	`def`	`Theorem~\ref{def:`*thing*`}`

Table 3.2 Suggested label kinds and reference forms.

Table 3.2 shows some suggested label type abbreviations and reference examples. Note that for sectional commands beneath the chapter level, we recommend simply using the `sec` type and the `Section` label in references; doing so allows you to change the level of a sectional element without having to worry about fixing the label and all the `\ref` commands throughout your document.

3.6.2 Nonbreaking Spaces and References

Notice that all the examples in Table 3.2 have a tilde character (~) connecting the label to the `\ref` command. That tilde is called a *tie*, and it acts as a nonbreaking space to prevent the label from being separated from the reference (e.g., at a line or page break). You should almost always use a tie with your `\ref` commands.

3.6.3 Page References

You can also refer to the page on which a `\label`ed element appears by using the `\pageref` command with the same argument you would give to a `\ref` command. For example, Table 3.2 appears on page 33.

3.6.4 Equation References

For your thesis, we require you to use references of the form `Equation~\ref{eq:thing}`. In other publications, you may be required (or want) to use an abbreviation instead of the full word, or even drop the label altogether.

The amsmath package includes a `\eqref` command for formatting equation references. This command formats equation references as a number surrounded by parentheses, as in "(1.3)". In the thesis class, this command has been overwritten so that it produces our desired label form.

3.6.5 Multiple References

If you're referring to more than one `\label`ed item, you don't need to label each `\ref` command. For two, use a tie to connect the first number to the label followed by the word "and" and the next reference. For example,

```
Equations~\ref{eq:thing1} and \ref{eq:thing2}
```

For a range of items, say several figures illustrating some point, you might do

```
Figures~\ref{eq:thing1}--\ref{eq:thing4}
```

where you're referring to `thing1`–`thing4`, and would end up with something like

Figures 1.3–1.7. . . .

If you don't have a continuous range, write something like

```
Figures~\ref{eq:thing1}, \ref{eq:thing3},
and \ref{eq:thing4}
```

which would give you something like

Figures 1.3, 1.5, and 1.6. . . .

3.7 Bibliographies and Citations

Formatting bibliographies and maintaining citations is yet another difficult task that LaTeX can help you handle. The BibTeX system provides a (command-line) tool, a language to specify the structure of bibliography entries (used in `bst`, bibliography-style, files), and a format for creating a database of

bibliography entries. Combined with the natbib package (see Section 3.7.2), BibTeX allows you to create flexible citations whose format can be changed as easily as the layout of the document.

BibTeX and natbib are reasonably complex; we won't cover their details here. Instead, see the rest of this section and the articles "Managing Citations and Your Bibliography with BibTeX" (Fenn, 2006) and "Tame the BeaST: The B to X of BibTeX" (Markey, 2005), as well as the appropriate chapters of any LaTeX references you may have to hand (such as Chapter 16 of Grätzer's *More Math into LaTeX* or Chapter 10 in his *Math into LaTeX*).

3.7.1 The Author-Date Style

The department uses an author-date style for citations and bibliographies. The author-date style provides more information about references than other styles, such as bracketed numerals, and is thus more accessible for a casual reader.

Examples of author-date style citations can be found in this document as well as the sample thesis/Clinic report (see Section 3.3.2).

3.7.2 The **natbib** Package

The natbib package, written by Patrick Daly, supports a wide variety of bibliographic citation formats, and also provides support for leveraging the power of LaTeX and BibTeX's substitutions to allow you to place text inside citations or create parenthesized comments using a single command, such as "(see Grätzer, 2007: chapter 10)", produced by

```
\citep[see][chapter~10]{gratzer-mmil}
```

The natbib commands also allow you to insert references as names, years, or other forms. Table 3.3 shows several variants. Note that the most common usages are `\citep` or `\citet`; if you're converting an existing document that simply used `\cite`, you can often use

```
\let\cite\citep
```

in the preamble of your document to convert all the citations to use the `\citep` form, and then adjust the citations that need special treatment. (Obviously you can achieve the same thing by doing a global search and replace to change the citation commands throughout your document.)

Form	Command	Result
Compact author lists	`\citeauthor{MathWrite}`	Knuth et al.
Full author lists	`\citeauthor*{MathWrite}`	Knuth, Larrabee, and Roberts
Just publication year	`\citeyear{gratzer-mmil}`	2007
Year with parentheses	`\citeyearpar{gratzer-mmil}`	(2007)
Author names, but with years in parentheses	`\citet{gratzer-mmil}`	Grätzer (2007)
Whole citation in parentheses	`\citep{gratzer-mmil}`	(Grätzer, 2007)

Table 3.3 Examples of natbib Commands.

Documentation for natbib can be found in the files `natbib.`*`ext`* and `natnotes.`*`ext`*, with *`ext`* being dependent on the exact form of the documentation on your system; some systems use DVI, some PostScript, and some PDF formats for documentation. On Unix-like systems such as Linux or Mac OS X, you may be able to easily access this documentation using the `texdoc` command; as in `texdoc natbib` or `texdoc natnotes`.

`natnotes` is a handy "cheatsheet" for the various commands provided by the natbib package.

3.7.3 The hmcmath Bibliography Styles

The format of your citations and bibliography are provided by the bibliography style file. The department provides three bibliography styles for use with theses or Clinic reports:

hmcmath.bst A basic bibliographic style modeled after the style recommended by the fifteenth edition of *The Chicago Manual of Style* (University of Chicago Press, 2003).

hmcmathlabelledannote.bst The same style, but with annotations (recorded in the `annote` field) turned on.

hmcmathannote.bst The same style with the `key` field and the label for the `annote` field removed.

In most cases, you should use hmcmathannote when you're asked for an annotated bibliography, and hmcmath when you don't need annotations.

One of the key features of these bibliography styles over some of the standard BibTeX styles is their support for URLs, which are increasingly common in today's Web-oriented research community.

These bibliography style files are available on the department's Linux systems and can be used on department machines without any additional work on your part.

If you want to use these styles on your own system, or on a system not managed by the math department, they are stored in `/shared/local/texlive/texmf-local/bibtex/bst/hmcmath` and can be copied from there or downloaded from https://www.math.hmc.edu/computing/support/tex/bibliography-styles/.

3.7.4 Constructing Your Bibliographic Database

Once you've read some of the references suggested in Section 3.7, you should be able to start creating bibliographic databases for use with your report.

3.7.4.1 BibTeX Tools

There are many tools that can make composing BibTeX bibliography databases easier. The MacTeX TeX system we recommend for use on Macs includes BibDesk, a graphical bibliography tool. Windows tools include BibDB. A cross-platform (Java) tool is JabRef.

A Google search will help you find more tools, but we would also encourage you to consider another cross-platform solution: Emacs.

Emacs (Gnu Emacs and XEmacs) has excellent support for bibliography databases with `bibtex-mode`, which is included with Emacs. It helps you enter new records, clean up existing records, validate records, sort records, and can even convert some non-BibTeX records to BibTeX format. Combined with AucTeX and RefTeX, Emacs provides a complete editing environment that allows you to compile your document, track changes, interact with version-control systems (see Section 3.8.2), navigate your multifile documents, and insert and track `\label`, `\ref`, and `\cite` commands.

3.7.4.2 BibTeX Tips

A few less-than-obvious tips that might come in handy include

- Some of the web-based bibliographic databases include the option to export citation information in BibTeX format. Sometimes these citations are everything you need, sometimes they need some additional work to make them as useful as possible, but they're usually a good start. Such BibTeX records may even include annotations, notes, or other useful information (although you should create your own annotations if you're producing an annotated bibliography).

- Choose meaningful citation keys (and be consistent!). Citation keys are the strings that you use to refer to bibliographic records in your LaTeX document. Your keys should give you enough information that you can identify—at least roughly—the source from the key alone when reading through your LaTeX code. Many of the searchable databases have keys based on some obscure coding, such as a DOI[7] or the Mathematics Subject Classification. These identifiers work great within the system they're part of, but they're meaningless to most human beings.

  Instead, choose keys that *you* will recognize on sight; for example, `smith-1990-english-cocoa` for an article by John Smith, published in 1990, about the varieties of cocoa available in the English market; or `turing-computable` for Alan Turing's "On Computable Numbers, with an Application to the *Entscheidungsproblem*" (1936).

- When entering names, use the *`last name`*`,`*`first name`* format for all the names, separating multiple authors' names with `and`. Being consistent about this data-entry method makes it easier for you to avoid mistakes with extra commands, reversed names, and so forth. It can also help BibTeX deal with some issues with sorting names with multiple particles (e.g, "Per Brinch Hansen", whose last name is "Brinch Hansen"; or "Charles Louis Xavier Joseph de la Vallée Poussin", whose name should be entered as `de la Vall\'{e}e Poussin, Charles Louis Xavier Joseph` (Walsh, 2002)).

  This method can also help you be clear about the correct order of names in languages that place the family name first, such as Hungarian and many Asian languages.

- Try to find the most complete name you can for an author. If a

[7] Digital Object Identifier, a globally unique identification system meant to label digital content. See http://www.doi.org/.

BibTeX record you find online only has "A.N. Author" (or "Author, A.N.", as we recommend here), try to find out the author's first name. (Middle names are a bonus.) When a last name is common, there could be several first names that abbreviate down to the same initials; distinguishing between "Sally Smith", "Samantha Smith", "Samuel Smith", and "Steven Smith" when all *you* have is "S. Smith" is not an easy task.

Note that the formatting of the author names in the bibliography is handled by the bibliographic style you choose. So one journal might want full names, and another abbreviations, but BibTeX will handle the formatting for you.

It's always better to have more information than you need and throw some of it away than to not have enough.

- Enter titles using title caps; that is, rather than entering, "An article about important topics"; enter "An Article About Important Topics". Here, again, BibTeX handles the formatting, and it will downcase words when needed. But BibTeX can't add capitals.

- Protect required capitals within entries. Sometimes you have proper names, countries, acronyms, or other strings that should always be capitalized, even if the rest of the string is downcased. Stop BibTeX from changing them by enclosing the capital letters in braces. Examples include

```
An Exploration of {C}harleston's {T}heorem
{S}wedish Mathematics in {N}orwegian Classrooms
How the {U}nited {N}ations Formed {UNICEF}
On Computable Numbers, with an Application to
  the \emph{{E}ntscheidungsproblem}
```

3.7.5 Citing HMC Theses and Clinic Reports

You might find yourself wanting to cite a previous student's thesis or Clinic report. BibTeX provides a number of different possible ways of doing so, but we suggest the following:

Theses Cite theses with the `@Misc` entry type. Use the `title`, `author`, and `year` from the thesis, and use

Harvey Mudd College Mathematics Senior Thesis

as the content for a `howpublished` field.

Clinic Reports Cite past Clinic reports with the `@TechReport` entry type. Fill in the `title` and `year` from the report, and use the Clinic team members' names (in alphabetical order) for the `author` field. Specify

Harvey Mudd College Mathematics Clinic

as the `institution`, and, if the sponsor's name isn't mentioned in the title of the report, add a `note` field with something similar to

Project sponsored by *`Sponsoring Corporation`*.

3.7.6 Citing Webpages

More and more useful material is appearing on websites every year. Where this material is a clearly defined "article" or "technical report"—especially when it's a PDF offprint that you can download and view—you can (and should) use the appropriate BibTeX entry type. But sometimes the material really is just a page on a website. For those cases,

1. Use the `@misc` entry type.
2. Set the `howpublished` field to "online; last viewed *`date`*".
3. Set the `url` field to the URL for the webpage.

If possible, you should also set the `year` and `month` fields to the publication date of the webpage (e.g., the copyright year or the date in the header of the webpage's HTML code).

3.7.7 Citing Other Electronic Documents

The `hmcmath` and `hmcmathannote` bibliography styles also support an `eprint` (for, e.g., arxiv.org prints) and `doi` (Digital Object Identifier) field, both of which are currently treated the same as a `url` field (i.e., set as a URL and made into a hyperlink).

3.8 Handling LaTeX Source Code

The following sections suggest some ways of handling a document of the size and importance of your thesis.

3.8.1 Splitting Your Document Into Smaller Files

For a report of this size, we generally recommend that you have

- A *master file* that contains the identifying information for your report, along with all command and environment definitions, package loading, and other code that should be available throughout the report
- *Separate chapter files* containing the material for each chapter

You should also, of course, have a

- *Separate bibliography database* file

Although the separate files approach is often seen as an advantage for groups of collaborators, as each writer can work on a separate piece of the larger document in its own file, this approach also has some advantages for people working on their own.

The organizational benefits are obvious, but if you're constantly retypesetting your document while you're trying different formatting approaches to a particularly troublesome structure or equation, you can actually save time by not typesetting the entire document each time. The `\includeonly` command allows you to decide which `\included` files are typeset. After typesetting your whole document a couple of times (to populate the auxiliary files with labels, entries for the tables of contents and lists of figures and tables, etc.), you can add an `\includeonly` command in the preamble of your document and only typeset the file you're working on, which will just give you the content of your chosen files, but with the cross-references still filled in. (By listing all your `\included` files in the `\includeonly` command and commenting out the files you're not working on, you can easily switch files on and off without having to make other changes.)

While you can also split the content of each chapter into multiple files (e.g., one for each section), it's usually easier to keep the level of chunking at the chapter level. But if your particular needs make smaller chunks more useful, talk to your advisor or the department's system administrator about whether additional splits are a good approach.

3.8.2 Using Version Control

We also encourage you to use some form of version (or revision) control system (VCS), such as Subversion or Git. These systems keep track of changes that are made to files within them so that you can compare different versions

and even revert some changes. You can also add comments to explain what you've changed and why you changed it, split off a separate copy to do some disruptive work ("branch"), and later merge those changes back into your main tree (the "trunk"), and so on.

Version control systems also provide a way to allow multiple people to work on a project by giving each person their own copy of the code. Changes made to one individual's copy of the code can't affect anyone else's checked-out copy until they check in their changes and your collaborators update their copies from the repository. (While coordinating multiple people's work is more likely to be a concern when working on a project for publication, you might also find it useful for sharing your LaTeX code with your advisor, second reader, or others (e.g., when asking for help in diagnosing a problem, or advice on how to typeset a specialized part of your thesis).

The major difference between Subversion and Git is that Subversion uses a central repository, and you need access to that repository to commit your changes (e.g., over a network connection). Git is a *distributed* VCS, which, amongst other things, allows you to make commits without direct access to that central repository.

3.8.2.1 Subversion Resources

More details on using Subversion (which was our required VCS for Clinic projects until recently) are available on the department's website at https://www.math.hmc.edu/computing/support/version-control/. The "Subversion book", *Version Control with Subversion* (Collins-Sussman et al., 2004), written by some of the developers of the system, is available online[8] as well as in printed form. Another good book on best-practices use of Subversion is Garrett Rooney's *Practical Subversion* (2006).

3.8.2.2 Git Resources

The best resources for Git are

- http://git-scm.com/: Git's home, with lots of links to documentation, tools, and more.
- http://git-scm.com/doc/: Git's official documentation, including Chacon and Straub's *Pro Git* book (2015), which is available under a Creative

[8] At http://www.svnbook.org/.

Commons license and can be purchased as a physical book, or downloaded for free in various e-book formats (PDF, ePub, mobi, HTML) from https://progit.org/.[9]

- http://stackoverflow.com/questions/tagged/git: Latest questions about Git on http://stackoverflow.com/, a technical support site for developers.

3.8.2.3 Hosted Git Repositories

Free (to students) repositories are available from a number of hosting services on the Web. While Github (https://github.com/) is the best known and most popular, Atlassian's BitBucket (https://bitbucket.org/) (which also supports the Mercurial VCS), and GitLab (https://gitlab.com/) offer similar services and are more generous to academic users. In particular, Github offers a free plan that includes five private repositories (where you control access) plus unlimited public repositories; whereas BitBucket offers unlimited private and public repositories to users with `.edu` e-mail addresses and GitLab offers unlimited private and public repositories to anyone.

3.8.2.4 VCS Tools

Command-line tools are included with most or all Linux distributions and with Mac OS X (with the (free) XCode developer tools installed). For Windows, you can install Git from https://git-scm.com/download/win/ or Subversion from https://subversion.apache.org/packages.html#windows.

If you're not comfortable with the command line, or you want to get a visual overview of your code, both VCSs have a number of different GUI tools. You can probably find more by searching online, but here are several well-known and well-regarded options:

Git

gitk Included with Git (free; cross-platform)

Github Desktop Github's Git client; specifically optimized for working with the Github service (free; Mac/Windows; https://desktop.github.com/)

SourceTree Git/Mercurial client from Atlassian (free; Mac/Windows; https://www.sourcetreeapp.com/)

[9] Further references to *Pro Git* will use page numbers taken from the PDF.

TortoiseGit Integrates into Windows File Explorer (free; Windows-only; https://tortoisegit.org/)

GitX Another GUI (free; Mac-only; http://gitx.frim.nl/)

Subversion

RapidSVN (free; cross-platform; http://rapidsvn.tigris.org/)

TortoiseGit Integrates into Windows File Explorer (free; Windows-only; https://tortoisesvn.org/)

Various commercial options. (Note that it seems like the free client development for Subversion has slowed or stopped.)

`<opinion>`

As of 2015, Git appears to have won the popularity contest for best version-control system, and maybe rightfully so. It was originally written by Linus Torvalds to manage the source code for the Linux kernel, and thus got a community of serious users right out of the gate, which helped it get even more buy-in from other free-software developers.

Git's distributed-repository model and cheap branching makes it particularly nice for exploring alternatives—wonder what a big mathematical display might look like using `flalign` or `alignat` instead of `align`? Just create a new branch and try it out. If you hate it, pop back to your original branch and delete the other one. If you like it, merge it in and delete the branch. If you're not sure, well, keep that branch around—you can always switch or delete it later.

Although the department still uses Subversion for some things (notably the code for web site), we've moved to Git for large parts of the scripts, configuration files, and other bits and pieces that get used behind the scenes. We're also experimenting with the (official) use of Git with Clinic this year, after discovering that many of the teams had used Github unofficially last year.

On top of that, with Git's mindshare and solid roots in the free-software community, not to mention its monetization by various web-based services, there are many excellent GUI clients available for free—although the command-line tools are incredibly powerful and give you even more control over every aspect of its

operation (the whole system is scriptable, and you can get very far down inside its data structures), for most day-to-day use, a GUI will do you proud.

I (Claire Connelly) mostly use Macs these days, and while I do most of my Git work on the command line, I also use SourceTree to do cherry-picking when doing commits—the ability to select which bits of a file or files you want to commit together, independent of other changes, was the big thing that made me switch. (Subversion only allows you to check in whole files, which means you have to be very careful about only making one related change before committing, or you'll have to juggle files around to get just the set of changes you want to commit.

`</opinion>`

3.9 Getting More Help with LaTeX and Mathematical Typesetting

At first glance, LaTeX may seem incredibly baroque and difficult to understand. After some experience (and judicious reading of documentation!), you should be able to do most basic tasks in LaTeX without having to think about them too much.

But there are always more complex tasks that you'll know you should be able to do, but whose exact coding won't be obvious. For those situations, consult references (the examples in Chapter 5 of *Math Into LaTeX* (Grätzer, 2000) or Chapter 9 of *More Math Into LaTeX* (Grätzer, 2007) are particularly useful for complex math displays), and consult (in no particular order) the department's TeX support pages, your advisor, the department's system administrator, Google, or the `latex-l` list hosted by the college. (See https://www.math.hmc.edu/computing/support/mailinglists/ for more information on subscribing.)

3.10 Vetting and Submission of Your Report

Before your final thesis is accepted by the department, it will undergo a rigorous review process by your advisor, the thesis coordinator, and the department's system administrator. You may be asked to revise your report as a result of any or all of these review steps.

Note that you can seek out comments on your draft at any time, and doing so earlier rather than later can help streamline the process.

3.10.1 Checking Your Code

Appendix C includes a list of frequently seen problems to avoid, which will give you an idea of the sorts of things that the code/format review will be looking at.

Running the LaCheck program on your code[10] will help you find and fix many (but not all) of the problems discussed in Appendix C. It reads in your entire document (including all the subfiles), and gives you feedback about the code. LaCheck is particularly helpful for finding issues with spacing (use or nonuse of the tie command; extra spacing in environments or command arguments; missing `\@` before sentence-ending punctuation where a word ends with a capital letter); issues with ellipses; italic corrections; correct use of quotation marks; and so on.

Note that LaCheck is not perfect, and that it may complain about things that, while problematic in the general case, may, in your particular usage, be correct. So you will need to look at each warning in context and decide whether the "problem" needs to be addressed or not.

LaCheck is available on any departmental Mac OS X or Linux system.

[10] With `lacheck` *`username-year`*`-thesis.tex`.

Chapter 4

Presentations and Posters

A major component of your thesis experience is training and practice in giving good talks, an art form that requires practice and discipline to master.

Practice sessions are held in class from the start of the semester, when the material you'll be presenting is relatively simple. Your presentations will become more complex and demanding as your project moves forward and you have much more to say within that limited time slot.

You will be giving at least three talks at the college as part of your thesis work.

1. During the fall semester you will be giving two introductory talks to the thesis-seminar class about your project (a 10-minute talk and a 20-minute talk);

2. During the spring semester you will give a talk, present a poster, or otherwise share your work with the mathematics community during a conference or other mathematics meeting;

3. During Presentation Days (May 6 – May 8 (Mon–Wed)), you will give a 20–25 minute talk about your research to a general audience composed of the college community and the general public.

Depending on how you fulfill the public presentation component of your thesis work (Section 1.4), you may be giving additional formal talks or poster presentations at a professional meeting.

4.1 Your Presentation

A typical presentation will be given about 30 minutes, with 25 minutes for the talk itself, and 5 minutes set aside for questions and answers; the exact balance you choose will depend on the nature of your project and may change throughout the year.

Remember that your presentations should be aimed at a general audience, or, at most, an audience that is only vaguely familiar with the ideas behind your work. Avoid technical jargon and excessive detail unless you absolutely must include such details to make your project's goals and accomplishments clear. It's better to give a good general overview and leave the details for the question period or for other discussions than to leave your audience in stunned incomprehension or utter boredom.

You should work closely with your advisor when preparing your talks. Your advisor should review your slides, and, if possible, attend your practice presentations and provide feedback. You may also want to get other people—friends, other thesis or Clinic students, innocent bystanders—to watch your presentation and give you feedback.

We recommend you consult Doumont (2009) for details about constructing and giving presentations.

4.1.1 Content

Your presentation should have several major parts, each split into smaller parts, perhaps represented by separate slides. We encourage you to maintain the same format for all the talks that you will give during the year. Doing so allows you to become comfortable with your presentation and to fine-tune it as your work evolves.

Introduction Begin with an introduction, covering about 5 minutes. Include

- A title slide
- A *brief* and *relevant* overview of your areas of interest as related to the project
- A brief description or explanation of the problem you're working on

The Heart of Your Talk About 15 minutes. Include

- Your approach, in descriptive, *nonspecialist* language
- What you've accomplished—discuss your analysis and results

Conclusion About 5 minutes. Include

- A discussion of work remaining to be done or negative results
- Recommendations for future research

Questions and Answers About 5–10 minutes. Open the floor to the audience. You may want to end your talk with a slide that makes it clear that you're welcoming questions.

Slides Covering Additional Material You may want to include some slides that cover some of the more advanced material that you didn't include in your talk, but that you think people might ask questions about. Although doing so is optional, actually having slides covering those questions can really wow an audience. (Hint: Figuring out what questions you might want to cover is best done by giving practice presentations and paying attention to the questions that people ask you during the practice sessions!)

4.2 Slide Content

Slides should act as reinforcements or illustrations of what you're saying rather than as a distraction. *You* are the main show; you want your audience to focus on you and what you're saying rather than having them struggle to read large swaths of tiny text on a slide, reading all your bullet points and getting ahead of where you are in your talk, or being dazzled by brightly colored, clashing borders from your slide style.

Your slides act as supplements, illustrating complex ideas or allowing you to summarize information with a graph. See Edward Tufte's essay, *The Cognitive Style of PowerPoint: Pitching Out Corrupts Within* (2006b) for more details on the negative impact of "PowerPoint-style presentations" on communicating information.[1]

[1]Tufte's website features a discussion thread about this essay, with posters contributing stories from their own lives.

Tufte's other books, notably *The Visual Display of Quantitative Information* (1983) and *Beautiful Evidence* (2006a), include excellent examples and guidelines on constructing powerful, *meaningful* documents, posters, or presentations to communicate scientific information.

Many of Colin Purrington's guidelines for designing posters (2016) are also relevant when designing talks.

4.2.1 Textual Material

Keep the textual content of your slides brief and to the point. One increasingly popular approach—developed in response to the text-rich but information-poor style described by Tufte (2006b)—is to limit the content of slides to just a few words. Some presenters limit the text on a slide to a single word.

Stanford law professor Larry Lessig is a pioneer in using this style of presentation. Garr Reynolds discusses Lessig's style and provides links to some examples (2005). Reynolds's site, Presentation Zen, also describes many other presentation styles—if you don't like Lessig's approach, you may find another that you're comfortable with using for your own talks there.

4.2.2 Presentation Graphics

The key with graphics is to *keep it sane*. It's easy to get caught up in the idea of providing lots of shiny pictures for your audience to look at, but it's far better to keep graphics to a minimum, which increases their impact on the audience dramatically—if all they've seen so far is sparse text, seeing an image tells them that there's something important happening.

In most cases, you can describe everything you need to talk about in words. But if the best way to get a point across is to use a graphic—a graph, a diagram, a chart, even an animation—do it.

4.2.3 Handouts

You might want to consider creating handouts for your talk, especially if you're giving it as part of a larger program where there are many speakers and the schedule is tight. Handouts should include basic information—the abstract for your thesis or project, your name, e-mail address, and the for your website—along with some other interesting, memorable, or illustrative material that will help someone remember your talk and want to read your whole thesis or report.

Unless it's very short, you probably won't want to include an entire article. For a poster session, though, a miniature version of the poster might be just the thing—but be sure that your contact details are legible.

Be careful about making handouts to be distributed *before* your talk—if your handout covers your whole presentation, people will read ahead and might not pay as much attention to your talk as you'd like.

4.3 Feedback and Critiquing Presentations

The talks you give in class are meant to encourage critiques on your presentations from your peers. We encourage (nay, *require*) you to attend these sessions and to think of good questions that you can ask during the Q&A session at the end of each presentation you see. By asking (and being asked) good questions, you will learn about things that you may have left out of, underemphasized, or glossed over in your presentation. This information will help you adjust your presentation before you make it again.

If you have a sudden flash of insight into someone's problem, share it! You may help your classmate find a new way of exploring their problem and help them get better results than they might have done without your question.

4.4 Creating Research Posters

Most of the issues we've discussed about creating and giving effective presentations are also important when creating a poster—with the additional need for the poster to stand on its own when you're not there to answer questions.

Research posters have to balance the main goal of communicating your work with that of having an attractive poster that encourages people to take a closer look. In particular, while you might be tempted to restate everything in your report on your poster, you need to keep the font size large enough that people can read it from several feet away, but there are many other things to consider as well.

Our LaTeX poster class, hmcposter,[2] takes care of many of the basic decisions you might otherwise need to make, and also happens to produce posters that we can easily print.

[2] https://www.math.hmc.edu/computing/support/tex/classes/hmcposter/

The hmcposter class has some disagreements with Colin Purrington's "Designing Conference Posters" (Purrington, 2016)[3], but Purrington's article is an excellent discussion of how to make a good poster.

4.5 Good Resources for Presentation and Poster Design

In addition to Purrington's articles and examples, there are a number of other good resources that are worth checking out.

- *Trees, Maps and Theorems: Effective Communication for Rational Minds*] by Jean-Luc Doumont (Doumont, 2009) is a fantastic book that covers writing, presentations, graphic displays, and more.[4]
- Garr Reynolds's web article, "Presentation Zen: The 'Lessig Method' of Presentation" Reynolds (2005), gives an inspirational argument for a minimalist presentation style, with the emphasis kept on the speaker(s) rather than the slides, using the slides as visual background or for specific illustrations rather than including much of the text of the talk on the slides themselves. (See also his more detailed book of the same title, published in 2011.)
- Edward Tufte's books on design and data visualization, *Beautiful Evidence* (Tufte, 2006a), *The Visual Display of Quantitative Information* (Tufte, 1983), *Envisioning Information* (Tufte, 1990), *Visual Explanations: Images and Quantities, Evidence and Narrative*, and his essay "The Cognitive Style of PowerPoint: Pitching Out Corrupts Within" (Tufte, 2006b), are good in so many ways, with fantastic examples from the history of visualization and strong advice.

4.6 Dress Codes for Presentations

Your in-class presentations during the year are informal, and your regular clothing will be acceptable.

[3]Article: http://colinpurrington.com/tips/poster-design. "Good" example: http://colinpurrington.com/wp-content/uploads/2011/09/scientific-poster-advice-purrington.pdf. "Bad" example: http://colinpurrington.com/2012/example-of-bad-scientific-poster/.

[4]There are also some videos and other materials viewable on line at http://www.principiae.be/X0302.php.

For formal practice presentations, however, we may require you to wear "business-casual" clothing, which basically means

- Neat and clean appearance
- For men, long-sleeve shirts, clean slacks, and dress shoes
- For women, formal skirts or slacks and blouses, sensible shoes (i.e., low heels or dressy but subdued flats)

For Presentation Days talks or poster sessions (May 6 – May 8 (Mon–Wed)), we will expect more formal office-appropriate clothing:

- For men, coat and tie
- For women, suits or formal business dresses

4.7 A Last Word: Sensible Paranoia

Don't tempt fate by incorporating new, untested graphics between your penultimate and your final presentations.

And *always*, even during practice sessions, check your audio-visual equipment before your presentation and be prepared for a last-minute disaster by keeping your presentation on a USB key, accessible from a website, or even having a set of overhead slides handy, just in case.

Chapter 5

Completing Your Thesis

There are a number of final tasks that must be completed so that you can receive a grade for your thesis and be eligible for department awards, and we can print, archive, and mail the final copies of your thesis. This chapter explains what those tasks are, how to perform them, and where to look for more information.

5.1 Due Dates

The exact due dates for the components of your thesis are specified in the calendar on page 1 of this document or in the online version of the calendar, available from https://www.math.hmc.edu/seniorthesis/current/calendar/.

5.2 Address Form

We need to have an address to which we can send your copy of your bound thesis. The bound theses will be ready sometime in mid- to late summer or early fall, so be sure that the address you give us will be valid for that period. If your address changes during that time, please notify us.

The form is available on the senior-thesis website, at https://www.math.hmc.edu/seniorthesis/tools/address-form.pdf.

If your address changes, please send the changes to Jocelyn Olds-McSpadden at jocelyn@math.hmc.edu as soon as you can.

5.3 Signed Signature Pages

You must deliver two (2) copies of the *signature page* of your thesis to the thesis coordinator by the deadline listed in the calendar.

Each signature page must be

- Printed on the department's color laser printer (gaspode) on the designated archival paper supplied by the department[1]
- Individually signed in blue or black permanent ink by
  - Your thesis advisor(s)
  - Your reader(s)

Remember that you may need to start the signature-gathering procedure early if one or more of your advisors or readers is not resident in Claremont.

The signature pages will be used for your copy and the department's library copy.

Obtaining the Signature Pages

The current version of the hmcthesis class file does not generate the signature page by default to facilitate the creation of ready-to-use copies of your thesis.

To get a signature page, you should temporarily modify your master LaTeX document by adding the `sigpage` document-class option to the `\documentclass` command in the preamble of your document.

The `\documentclass` command should look like

```
\documentclass[sigpage,mathematics]{hmcthesis}
```

The easiest way to have the signature pages printed on the correct paper is to submit a PDF file containing only your signature page to the mathematics department's administrative aide, Jocelyn Olds-McSpadden.

5.4 Copyright Form

Starting in 2006, the department's theses include a copyright statement that confirms your copyright in the work (which you are entitled to under U.S.

[1] We use 28-pound Mohawk Superfine Ultrawhite Smooth as our archival paper, which is acid-free and includes an alkaline buffer to further protect the paper from acid-related deterioration.

and international copyright law) and adds a note to the effect that you are granting the department a license to share your thesis for educational and nonprofit use. As we also send your thesis to the Claremont Colleges Library for inclusion in their Scholarship@Claremont digital library site, this form will also cover their copyright needs.

Our copyright deposit agreement form allows you to confirm that you are granting the department and the Claremont Colleges Library the rights stated in the copyright blurb that will appear in your thesis. The form is available on the department's website, at https://www.math.hmc.edu/seniorthesis/tools/deposit-agreement.pdf.

5.4.1 Mathematics Subject Classification

The CCDL takes part in a larger indexing effort for online materials, and part of its metadata is a keyword field. For theses, having up to three (3) classfications from the Mathematics Subject Classification (MSC) will make your thesis more easily findable and categorizable.

The Mathematics Subject Classification is available online at http://www.ams.org/mathscinet/msc/msc2010.html.

5.5 Your Thesis Itself

Although we will be typesetting your thesis to obtain a final copy for printing and archiving, we also expect you to typeset your thesis and to provide an electronic (PDF) copy—for use by the department for grading and reference.

5.6 Scholarship@Claremont

Theses are made available both from the department's website and the Claremont Colleges Library's Scholarship@Claremont site (http://scholarship.claremont.edu).

You will need to fill out the thesis submission form, which records a bunch of metadata about your thesis. Start by going to

http://scholarship.claremont.edu/cgi/ir_submit.cgi?context=hmc_theses

which will allow you to create an account. Once you get your account approved, fill out all the information on the form.

The answers to most questions are probably obvious, but just in case. . .

- Your name should show up as the author, but if not, you'll need to click the person with a pencil for the author to add your name and institution (just Harvey Mudd College is fine).
- The Date of Award should be May, `year`, where `year` is your year of graduation. (Ignore Season and Day.)
- Mudd only gives out Bachelor of Science degrees.
- Department is Mathematics.
- Second Department is blank unless your thesis was a joint thesis with an advisor from another department.
- The First Advisor should be your advisor and the Second your reader. Add a third if you had one.

  Be sure you're using the names of your advisors as they use them on publications—check with your advisor or reader. These are also the names that should be on your thesis.

  Don't include titles or degrees (e.g., "Prof." "Ph.D", "BSC").
- Rights Information should be your name again.
- Terms of Use & License Information can be the default license set by Scholarship@Claremont or a Creative Commons license; we only support the Attribution–NonCommercial–ShareAlike variant at this time.
- Keywords should be the MSC codes from above—both numbers and words, separated by commas.
- Subject Categories—please pick appropriate subject categories from the list (this field is one of the ones that's really difficult for us to fill out).
- Abstract—your abstract, with any citations expanded. Ideally you'd massage it to look right here (e.g., put variables in italics, set super- or subscripts).
- Comments—leave this field blank.
- Streaming media—leave blank unless you have some.

- Upload Full Text—you have to do that, but the published version of your thesis will be one that we typeset. So we recommend that you choose "Link out to file on remote site" and enter the path to your final draft of your thesis on your HMC math cluster thesis webpage.

You will be able to go back and edit your submission if you need to; later on, you'll be able to log in and see how many readers and downloads you've had for your thesis.

Unfortunately, there's no way for us to set defaults for these fields, so you'll need to fill them in even if there's only a single possible/reasonable answer.

5.6.1 Grading/Reference Copy

You must submit a Portable Document Format (PDF) file containing your complete thesis, typeset by you, which can be used by your advisor and the department for grading and during its awards process.

5.6.2 LaTeX Source Code

Even more important than the reference copy is the LaTeX source code needed to typeset your thesis. We will archive your source code and we will also use your source code to typeset the version of your thesis that will appear on the department's website and be used to print copies of your thesis for binding.

We will retain the source code indefinitely and can supply you with a copy should you wish to continue working on your topic and need to use that code as the basis of any future articles, books, or presentations. We do, however, also encourage you to make your own copies of your source code (see Section 2.6).

Section 5.7 explains the steps you should take to prepare your thesis code for archiving. You are also expected to use these instructions to submit the first draft of your final report (due March 28 (Fri) 5:00 P.M.).

5.6.3 Copies for Binding

We will print four copies of your thesis on archival paper and have those copies bound. Printing is usually done sometime during the summer.

Once we have the bound theses back from the bindery, we will distribute the copies as follows:

- One copy to you at the address you have provided (see Section 5.2)

- One copy for your thesis advisor
- One copy for the mathematics department's library

The fourth copy may be used to nominate your thesis for external prizes, such as the Morgan Prize.

If your thesis is chosen for one of the department awards, we will also print and bind a fifth copy for the award collection.

5.7 Archiving

The department's systems administrator will archive your thesis, its LaTeX source code, and, optionally, other source code, data, or other components, in consultation with the thesis coordinator. To make this process as easy as possible, follow the instructions in this section. If you have questions, ask.

5.7.1 Thesis Source Code

For your thesis, we want the *minimal set* of LaTeX code and supporting figures to allow us to compile your thesis correctly on a departmental machine.

Important Note

You can safely assume that the hmcthesis class and all its supporting files will be available on the department machines through the mechanisms supplied by our TeX systems. *Do not include any of these files.*

Also, be sure that you don't have any files from the sample thesis or any files produced by the typesetting process, including auxiliary and log files *and* DVI, PDF, or PostScript versions of your typeset thesis.

5.7.1.1 Thesis Directory

Start by creating a directory at the top level of your home directory on the math Linux cluster to contain the LaTeX code for your thesis. The directory must be named `username-year-thesis`; where `username` is your math department Linux cluster login name and `year` is the year of completion of your thesis.

Inside this directory, put

- The LaTeX files that you created to create your thesis—your master file, chapter files, any images, your bibliography database file, and so on.
- The source code for any images or figures that appear in your thesis that must be processed during the typesetting process to create the images that are included in your thesis.
- A file called `README` that includes instructions about any nonobvious requirements for typesetting your thesis: generating figures, glossaries, or indexes; scripts that have to be run, and so forth (i.e., anything much beyond "Run pdfLaTeX on my master file"). We should be able to type

  ```
  rubber -d username-year-thesis
  ```

  and get a compiled PDF.

Your "master" file should be named to match the name of the directory—*username*`-`*year*`-thesis.tex`, where the variables are the same as those for your thesis directory. Naming your file in this way allows us to typeset your thesis and get a PDF file with the correct name. (It also helps us script the process.)

Do *not* include any data files, extra images, source code not being typeset in your thesis, e-mail messages, movies, or other files or directories in the thesis directory except those files needed to typeset your thesis.

Nonstandard LaTeX Packages If you had to download any additional LaTeX packages to format your thesis, you may need to include copies of those packages in your thesis directory.

Try typesetting your thesis without the additional packages in your working directory on a department machine. If everything works without those packages, you don't need to include any additional files. If you get errors related to missing packages, include those packages in your thesis source-code directory and note their inclusion in your `README` file. If you obtained some of these packages from a source other than Ctan, the Comprehensive TeX Archive Network, be sure to tell us where you found the packages.

If you created any packages yourself, and those packages are required to typeset your thesis, please be sure to include those files as well.

5.7.1.2 Data Directory

If you have data, code, or other materials related to your thesis that you think we should hang on to (in case, for example, you're planning to work more on the same topic with your advisor(s) or readers(s)), create a directory in your home directory named `username-year-thesis-data`, and place those materials in that directory.

Once your thesis is archived, the contents of your data directory will made available from a URL similar to

> `https://www.math.hmc.edu/seniorthesis/archives/`
> `year/username/username-year-thesis-data.zip`

where, of course, `year` is your thesis-completion year and `username` is your username on the math cluster. You may refer to this URL in your thesis (e.g., in an appendix describing or documenting your code).

5.7.2 Postponing Archiving

Under certain circumstances, such as when your thesis work might be eligible for patents or otherwise requires secrecy for some amount of time, you may be able to arrange to postpone the publication of your thesis for up to a year. You will need to get permission from both your advisor and the Thesis Director to postpone publication.

Note that you are still expected to submit your source code and other materials just like every other student. We will publish your thesis as it was turned in when next year's theses are archived.

5.7.3 Thesis Website

We also archive parts of your thesis website (see Appendix E), especially the `index.html` file. To help us archive this material,

- In the `thesis` directory that you created inside your `public_html` directory at the beginning of the year, put a PDF of your thesis. Name it `username-year-thesis.pdf`, and make sure that any links in your `index.html` file point to that new filename.
- Remove any extraneous material from your `~/public_html/thesis/` directory—if it isn't linked to from your `index.html` file, it shouldn't be in that directory.

- Remove any links (and the associated files) to drafts, midyear reports, or other material from your `index.html`.

5.8 Personal Backups

You may want to burn archives of your files to CD-ROM or DVD-ROM or copy them to your own machine. The department does not supply blank media, but suitable media is available from Huntley, Target, your favorite office supply store, and quite possibly your preferred grocery store.

There's lots more information about burning CD- and DVD-ROMs available on the math computing website, including detailed instructions, at https://www.math.hmc.edu/computing/support/burning-cds/.

You can also copy your materials from your math account to your own machine using a command-line or GUI SCP or SFTP client. Command-line programs (`scp`, `sftp`) are included with most Linux or Unix-like operating systems (including Mac OS X). Clients with graphical interfaces are also available, including Fugu and Cyberduck on Mac OS X and Cyberduck and WinSCP for Windows.

Appendix A

The hmcthesis Document Class

LaTeX uses *document-class files* to define many fundamental aspects of a document, including font sizes; typefaces; predefined commands and environments; the formatting of various elements (e.g., title pages, lists); and other customizations.

The use of a document-class file allows the writer to concentrate on the *content* of their document, leaving the formatting to the document class. It also provides a layer of abstraction, as in most cases it is possible to change only the document class and obtain a typeset document that appears very different from the same document typeset with another document class.

To help you format your midyear report and your final thesis, the department has created the hmcthesis document class, which provides a title page for the midyear and final reports, font selection, and loads a set of core LaTeX packages to provide support for other aspects of the writing process.

The hmcthesis class also allows you to create a signature page for your advisor(s) and reader(s) to sign. (See Sections 5.3 and A.2.2.1.)

The hmcthesis class can be used on any mathematics department Linux system without any special configuration. If you would like to work on your own machine, you will need to get your own copies of the class and its support files and install them where your TeX system can find them. Appendix B explains how to obtain and install the necessary files.

The hmcthesis distribution also includes the `hmcthesis-template.tex` that you should use as the base for your "master file" when writing your midyear and final thesis reports.

A.1 Using the hmcthesis Document Class

The hmcthesis class includes a number of commands for entering metadata about your thesis (title, author, etc.) and also has some options that affect the formatting of the typeset document.

A.2 Options for the `\documentclass` Command

LaTeX documents begin with a `\documentclass` command, which specifies the document-class file to be used to format the document.

Document classes typically define *document-class options* to turn some functionality on or off.

The hmcthesis class provides several document-class options. Some of these are mandatory—typesetting your document without them will trigger an error message and stop compilation—and some of them are optional, but help format your document correctly for specific purposes.

A.2.1 Departmental Document-Class Options (Mandatory)

Because the hmcthesis class is used by multiple departments, you must specify a document-class option to set your department in the `\documentclass` command. For math, you'll put

```
\documentclass[mathematics]{hmcthesis}
```

The available (supported) department document-class options are

- `biology`
- `computer-science`
- `cs`
- `chemistry`
- `engineering`
- `math`
- `mathematics`
- `physics`

A.2.2 Document-Class Options (Optional)

In addition, hmcthesis provides two document-class options to help your format your reports for specific uses.

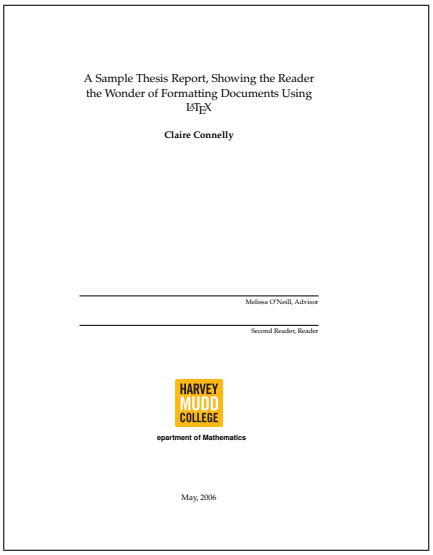

Figure A.1 A Sample Signature Page.

A.2.2.1 The `sigpage` Option

The `sigpage` option creates a page with signature lines for your advisor(s) and reader(s) to sign, indicating their acceptance of your thesis.

Use it as

```
\documentclass[sigpage,mathematics]{hmcthesis}
```

and produces a page similar to that shown in Figure A.1.

A.2.2.2 The `cclicense` Option

The `cclicense` option modifies the copyright statement on the back of the signature or title page of your thesis to license your thesis using the Creative Commons Attribution–NonCommercial–ShareAlike license.

See http://creativecommons.org/licenses/by-nc-sa/3.0/ for a summary of the rights given, withheld, and reserved by this license and http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode for the full legal details.

If you want to use the `cclicense` option, you will need to check the appropriate box on the Copyright Form we ask you to sign to allow the college and the Claremont Colleges Library to publish your thesis in various electronic archives. (See Section 5.4 for more details on the copyright-assignment form.)

A.2.3 Cover-Page Commands (Mandatory)

The hmcthesis document class supplies a number of commands for specifying information about your thesis project. As these commands are used to format the cover page for your thesis, these commands *must* be specified.

A.2.3.1 `\title` (Mandatory)

A title for your thesis. Probably something that captures the essence of your project. Used as

```
\title{An Exploration of Complex Interactions
  Between Mathematical Phenomena}
```

A.2.3.2 `\author` (Mandatory)

Your name. You probably want to use a formal variant; the version of your name that you would want to appear on any other academic publication. So, for example,

```
\author{Richard J. Petersen}
```

or

```
\author{Caroline R. Chanew}
```

A.2.3.3 `\advisor` (Mandatory)

The name of your faculty advisor (should you have more than one, use the `\and` command to separate them). Note that you should only include their name, *not their title*. So,

```
\advisor{George Washington}
```

You should use your advisors' "official names": the names they use on their own publications. See Appendix G for details and a list of HMC mathematics faculty official names. If your advisor isn't on the list, ask them what you should use.

A.2.3.4 `\reader` (Mandatory)

The name(s) of your second reader(s). As above,

```
\reader{Thomas Jefferson \and Alexander Hamilton}
```

You should use your readers' "official names": the names they use on their own publications. See Appendix G for details and a list of HMC mathematics faculty official names. If your reader isn't on the list, ask them what you should use.

A.2.4 Exceptions to Standard LaTeX Usage

1. Most LaTeX documents define a `\date` command to specify a date. The hmcthesis class does not support the use of this command—you can use it, but its contents will not be used when typesetting your thesis.

   Instead, for most theses you should specify the *year* that your thesis is being submitted by using the `\thesisyear` command, as in

   ```
   \thesisyear{2009}
   ```

   For spring–fall theses, you may also need to specify the month that your thesis was completed. You can do so with the `\thesismonth` command, as in

   ```
   \thesismonth{December}
   ```

   The `\thesismonth` command defaults to "May", so either specifying it with no argument (`\thesismonth`) or not specifying it all (as we recommend), will use May as the month.

2. The `\eqref` command provided by the amsmath package ordinarily formats equation references as the equation number surrounded by parentheses—for example, as "(1.3)". The hmcthesis class redefines this command to conform to our standard reference formatting expectations, so `\eqref` will instead create references that look like "Equation 1.3".

A.3 Default Output Format

Recent releases of TeX systems use the pdfTeX engine as the basic processing application for TeX and LaTeX documents. As pdf (Portable Document Format) files are also smaller than PostScript versions of the same document, and pdf documents are also better supported by free or integrated viewer

applications across all computing platforms, we encourage you to develop your documents exclusively using the `pdflatex` program for compilation.

Important Note

Note that no matter what format you may choose to use while writing your thesis, the final versions of your thesis must compile with `pdflatex`.

The pdfTEX engine only supports graphic file inclusions in PDF, JPEG, or PNG formats. If you have EPS (Encapsulated PostScript) images, you will have to convert them to PDF before they can be used. The `rubber` tool, a command-line script that should be available on most of the Linux/Unix systems in the department, can handle this conversion for you automatically. You can also do the conversions with `ps2pdf` on Linux systems or `pstopdf` on a Mac OS X system.

A.4 Basic Appearance of Documents

As the same questions crop up year after year, we will describe some basic fundamentals about the appearance of documents produced with the hmcthesis class. These are key elements of the class and *should not be adjusted or overridden* by your own LaTeX code. If you're curious about the specific choices that have been made, please feel free to discuss them with the department's system administrator.

Note that this document and the sample Clinic/thesis report are exemplars of the correct appearance of your final document.

A.4.1 Fonts

We use the newpxtext package to typeset the document using a typeface based on Palatino and a matching set of mathematical fonts (newpxmath and mathalfa).

- The sans-serif typeface is Source Sans Pro; adjusted to work with the x-height of the New PX fonts.
- Double quotes (“ ”) may look reversed; this appearance is a “feature” of the roman typeface.
- `The "typewriter" or monospace typeface is Source Code Pro.`

- The base font size is 11 points, with leading set to 13.6 points (by the standard bk11.clo file).

A.4.2 Page Layout

Pages are laid out using the basic parameters provided by the standard LaTeX book class. Documents are typeset for printing on both sides of the page ("duplexing"). All the laser printers available for student use within the department are duplexing printers.

Chapters and other major sections begin on right-hand pages; that is, it is possible *and correct* for there to be a blank page on the left-hand page in a two-page spread. The blank pages are completely blank—line numbers and headers should not appear on such pages.

There is no vertical whitespace between consecutive paragraphs; paragraph breaks are indicated by *indentation* of subsequent paragraphs rather than blank lines. Paragraphs immediately following a section header *are not indented*.

A.5 Packages Loaded by hmcthesis

The hmcthesis class loads a number of packages in addition to the standard LaTeX kernel code that's loaded by the LaTeX format and the code loaded by the standard LaTeX article or book class that hmcthesis uses as its base.

The packages loaded are:

amsfonts Support for additional fontsets provided by AMS-LaTeX.

amsmath The base package for the AMS-LaTeX system.

booktabs Support for *Chicago*-style tables.

caption Allows us to define the appearance of captions for figures, tables, and other floats. We also provide formatting information for use with the subfig package, but don't actually load subfig in the class.

fancyhdr Used to format the headers and footers of the document.

fontenc Support the T1 font encoding.

graphics Support for including graphics files in various formats in LaTeX documents. Note that the graphicx package provides some enhancements to the commands provided by the graphics package, including

the ability to use key=value arguments in the optional arguments of those commands, allowing you to reduce the amount of typing that you would have to do to insert nested commands. If you prefer the graphicx approach to graphics inclusion, just load graphicx in your document's preamble.

helvet Use Hevetica as the sans-serif typeface.

ifthen Support for `if...then` constructs in LaTeX code. Used in some of the class's definitions.

mathalfa Use the STIX BOONDOX calligraphic typeface.

natbib Use the natbib package for citation commands and bibliography formatting.

newpxtext Use New PX as the base (roman) typeface.

newpxmath Use New PX as the math typeface.

textcomp Provides encoding subsets to supply missing characters using the TS1 font encoding.

url Load the url package with the `obeyspaces` and `spaces` document-class options.

verbatim This package provides an enhanced version of the standard LaTeX `verbatim` environments.

xspace Provides the `\xspace` command, which provides a "smart space" that is used at the end of many of our defined commands. (By ending a command definition with `\xspace`, the command can be used as `\foo` rather than `\foo{}` to avoid having the following space eaten by LaTeX.

Appendix B

Installing the **hmcthesis** Class On Your Own Machine

The mathematics department's computers have the hmcthesis class and other support files installed so that they work without your having to do anything special. You can copy the template file to a new directory and start working immediately—the college's logo and seal, the bibliography style, and some of the configuration options "just work".

If you want to work on your own machine, however, you'll need to have local copies of several different files available for use by your machine's TEX system.

Although it's possible to use LATEX packages or classes without installing them (by placing them in your working directory), we encourage you to install the files into a "TEXMF tree" so that they will be usable by any document you create on your system. Installing support files also allows you to update them in one place, which means it's easy to bring all your documents using a particular class or package up to date, and to be sure that each document is using the latest version of these files that you have installed.

For thesis, having the class and support files installed in a TEXMF tree also makes easier for you to meet the requirements for submitting the electronic archive of your thesis source files—see Section 5.7.1 for details.

TEX System	Path
TEX Live (Linux/Unix)	`/usr/local/share/texmf`
MacTeX (Mac OS X)	`/usr/local/texlive/texmf-local`
MikTEX (Windows)	`C:\LOCALTEXMF`

Table B.1 Local TEXMF tree locations for some common TEX systems.

B.1 TEXMF Trees

Modern TEX systems contain thousands of files, including classes and packages, but also several different kinds of font and font-related files; support and format files for bibliographies, indexes, and other document elements; and support for different macro systems built on TEX, including Plain TEX, LATEX, ConTEXt, and others. On the machine being used to write this appendix, there are roughly 66,000 files in the TEX-Live installation.

Each program in a TEX system uses a different combination of these files, and if each program had to search through all the files to find the files it needed, the system would be quite slow. To make the task easier, TEX-system developers created the TEX Directory Structure, or TDS, which defines a standard hierarchical file system to be used to sort components into standard locations.

Furthermore, most TEX systems support multiple TDS-compliant trees that are searched in a particular order, which means that the TEX system can have its own set of files that can be managed separately from those installed by a system administrator or by individuals.

The TEX Directory System is defined and described in a document maintained by the TUG Working Group on a TEX Directory Structure (TWG-TDS, 2004). The whole tree is fairly complex, but most people only need to worry about implementing a small subset of the whole tree.

B.1.1 Local TEXMF Trees

Local TEXMF trees are usually shared by all users of a computer. Their exact location depends on the TEX system in use, but some common locations are shown in Table B.1.

On departmental Mac OS X and Linux systems, the local TEXMF tree is at `/shared/local/texlive/texmf-local`.

TEX System	Path
TEX Live (Linux/Unix)	`~/texmf`
MacTeX (Mac OS X)	`~/Library/texmf`

Table B.2 Personal TEXMF tree locations for some common TEX systems.

B.1.2 Personal TEXMF Trees

Personal TEXMF trees are supported on some systems by default, and on others with special configurations. They're most useful when working on a system shared by other users (such as the math workstations), especially when end users don't have access to system directories, or might want to override the choices made by system administrators.

For Unix-like systems such as Linux and Mac OS X, personal TEXMF trees can be set up in the user's home directory and are automatically searched. The locations of these trees are shown in Table B.2.

B.1.3 MikTEX Root Directories

The MikTEX system for Windows doesn't support local or personal TEXMF trees by default, although you can set up such directories by using its configuration tools. The Settings tool (in the MikTEX program group) includes a "Roots" tab. You can create a directory (such as `C:\LOCALTEXMF`) and then use the Roots tab to add that directory. The directory you create will have to have subdirectories conforming to the TEX Directory System standard.

Once you've added your new root, you should use the Up and Down buttons to ensure that your root is searched before the other roots. (Alternatively, you can put your root last, and you will always get the version of packages that come with or were installed by MikTEX, but packages you've installed that aren't included in MikTEX's system will be found.

If you add files to your local root after you've added the root to MikTEX's list of root directories, you need to refresh the filename database by hand so that MikTEX's component applications can find the new files.[1]

To update the database, start the MikTEX Settings application and select the General tab. You should see a pane labeled Maintenance, and inside

[1] MikTEX's filename database is equivalent to the `ls-R` files that Unix TEX systems use; on Unix systems, local and personal TEXMF trees are searched, so you don't need to index them.

that pane a button labeled Refresh FNDB. Click the button, and wait while MikTEX rescans its file roots. Once it's done, you should be able to use your new files.

Note that you shouldn't need to refresh the database by hand after installing LATEX packages with the MikTEX package manager; the database should be rebuilt as part of the installation process.

B.2 Obtaining the Class and Support Files

There are two basic elements that you will need to produce your final thesis:

1. The class and its support files, which include the HMC seal and logo and some configuration files; and,

2. The `hmcmath.bst` bibliography style file.[2]

You can either download a `tar` file or ZIP archive containing the necessary files or copy them from a mathematics department computer.

B.2.1 Downloading the Class and Support Files

The hmcthesis class is available from https://www.math.hmc.edu/computing/support/tex/classes/hmcthesis/.

The bibliography style files can be downloaded from https://www.math.hmc.edu/computing/support/tex/bibliography-styles/.

There are also archives that contain the class and its support files and the bibliographic style files already arranged into the proper directories. If you don't already have your own TEXMF tree, we recommend that you get one of these archives rather than the separate pieces. The `hmcthesis-texmf` archives are available from https://www.math.hmc.edu/computing/support/tex/classes/hmcthesis/.

B.2.2 Copying the Class and Support Files

The class, bibliography-style, and support files can also be copied from any departmental Linux system. The latest version of the class file and support files can always be found in `/shared/local/texlive/texmf-local/`

[2]There is also a `hmcmathannote.bst` file that provides an annotated version of the hmcmath bibliography style, which might be useful if your advisor wants an annotated bibliography; see Section 3.7.3 for details.

```
texmf/
   bibtex/
      bst/
         hmcmath/
            ...
   tex/
      latex/
         hmcthesis/
            ...
         hmcclinic/
            ...
```

Figure B.1 Thesis and Clinic-related TEXMF subdirectories.

`tex/latex/hmcthesis/`. The bibliography-style files can be found in `/shared/local/texlive/texmf-local/bibtex/bst/hmcmath/`.

B.3 Installing the Class and Support Files

As noted in Section B.1, there are many possible subdirectories for local and personal TEXMF trees. For installing the basic support files for use with the hmcthesis class you only need a small number of these subdirectories, as shown in Figure B.1.

The `hmcthesis-texmf` archives provide these directories for you, allowing you to copy the `texmf` directory they contain into the correct location on your system. (The archive contains more detailed installation instructions.)

Alternatively, you can create the directory tree shown in Figure B.1 in your local (see

Section B.1.1) or personal (see Section B.1.2) TEXMF tree, copy the contents of the `hmcthesis` archive to *`TEXMF`*`/tex/latex/hmcthesis` and `hmcmath.bst` to *`TEXMF`*`/bibtex/bst/hmcmath`. You can also `cd` to the destination directory and unpack the archives there.

B.3.1 Testing Your Installation

If you want to test your installation, you can try TEXing a document using the class or package. If it works, you're good!

If not, you can use some command-line tools to see whether TEX can find your new files. Open a terminal or command window and `cd` into any directory except the one in which you installed the files (your working directory is a good choice, but the test should work in any directory). Run the command

```
kpsewhich hmcthesis.cls
```

The program should print the full path to your copy of the hmcthesis class file. If it doesn't, check to be sure that you have the same directory tree shown in Figure B.1 and that all the directory names are spelled exactly as shown.

B.3.2 Debugging Your Paths

Linux or Mac OS X systems include a `kpsepath` command that can show you the full set of paths that your TEX system will search when looking for a file.[3] So you might run

```
kpsepath -n latex tex
```

to show the full path that the `latex` program will search when looking for TEX source files (which includes class and package files). The output will look something like the following (note that we have split the output at the colons for clarity; the actual output runs everything together on one line):

```
.:
/home/students/username/.texliveyear/texmf-config/tex/latex//:
/home/students/username/.texliveyear/texmf-var/tex/latex//:
/home/students/username/texmf/tex/latex//:
!!/shared/local/texlive/year/texmf-config/tex/latex//:
!!/shared/local/texlive/year/texmf-var/tex/latex//:
!!/shared/local/texlive/year/texmf/tex/latex//:
!!/shared/local/texlive/year/../texmf-local/tex/latex//:
!!/shared/local/texlive/year/texmf-dist/tex/latex//:
/home/students/username/.texliveyear/texmf-config/tex/generic//:
/home/students/username/.texliveyear/texmf-var/tex/generic//:
/home/students/username/texmf/tex/generic//:
!!/shared/local/texlive/year/texmf-config/tex/generic//:
!!/shared/local/texlive/year/texmf-var/tex/generic//:
```

[3] On a MikTEX system, all the search paths are visible in the MikTEX Settings application.

```
!!/shared/local/texlive/year/texmf/tex/generic//:
!!/shared/local/texlive/year/../texmf-local/tex/generic//:
!!/shared/local/texlive/year/texmf-dist/tex/generic//:
/home/students/username/.texliveyear/texmf-config/tex///:
/home/students/username/.texliveyear/texmf-var/tex///:
/home/students/username/texmf/tex///:
!!/shared/local/texlive/year/texmf-config/tex///:
!!/shared/local/texlive/year/texmf-var/tex///:
!!/shared/local/texlive/year/texmf/tex///:
!!/shared/local/texlive/year/../texmf-local/tex///:
!!/shared/local/texlive/year/texmf-dist/tex///
```

The output of this command may be a bit confusing. Each colon-separated component is a directory that is searched by the program. The `.` at the top of the list is equivalent to the current directory (which is why you can put package files in your working directory and TEX will find them; it's also why we recommended that you run the `kpsewhich` command from a different directory in Section B.3.1). The double exclamation marks (`!!`) at the start of some paths indicate that the program should look for a file called `ls-R` at the root of that tree instead of searching the filesystem for matching files.[4] The doubled slashes at the end of the paths tell the search library to search subdirectories, not just the specified directory.

So in our example, the `latex` program will first look in the current working directory. Then it will look in private directories used by the TEX Live TEX system in your home directory (`~/.texliveyear/texmf-config/tex/latex` and `~/.texliveyear/texmf-var/tex/latex`; where `year` will be the year of release).

Next it will look in your personal TEXMF tree (`~/texmf/tex/latex`). Then it will use the `ls-R` file in `/shared/local/texlive/texmf-local/` to get a list of possible completions in subdirectories of `/shared/local/texlive/texmf-local/tex/latex/` (the `shared/local` tree is available on all math department Mac OS X and Linux systems, and is where the latest version of the hmcthesis class and other support files are kept). After that, the program looks at the main system TEXMF tree, again consulting the `ls-R` file.

After searching the `tex/latex` trees, the program will look for files in

[4] These `ls-R` files are generated by the `texhash` program, and are similar to the output of the `ls` command with the `-R` (recursive) flag. Having the system's files be indexed in this way makes TEX much faster.

the `tex/generic` subtree (files that can be used by LaTeX or TeX), and then for files meant for use by TeX alone.

If your local or personal TEXMF tree doesn't appear in the list, you may need to do some additional configuration. Please consult the department's system administrator or the `latex-l` list (see Section 3.9) for help.

B.3.3 More Information

If you're really curious about how TeX searches for files, you should read the article "A Directory Structure for TeX Files" (TWG-TDS, 2004) and the "Kpathsea Library" manual (Berry and Weber, 2008).

Appendix C

Commonly Seen Problems

Unfortunately, year after year we see some of the same issues with the LaTeX code submitted by students. To help you not make the same mistakes others have made, this appendix explains some of the most commonly seen problems and how to avoid them.

Some of the issues discussed here are fairly obvious, but some deal with the intricacies of writing LaTeX code and may not be at all obvious.

C.1 Spell Checking

A surprising number of final reports contain spelling errors.

Use a spell checker to check your entire document. There are LaTeX-aware spell-checking programs available on all platforms. Some editors also include spell-checking as you write, which is especially useful. If you need advice on choosing an editor or a spell checker, ask!

C.2 General Style and Code Format

The sample thesis/Clinic report code provides an example of how best to format your LaTeX source code for easy reading and editing. It also includes many examples of complex LaTeX structures and has pointers to relevant references. More information about obtaining the sample thesis/Clinic report is available in Section 3.3.2.

C.2.1 Hacking at the Document Format

A major problem that we see occasionally is an attempt to make the typeset thesis or Clinic report look like a Word document. Examples include adding line-break commands to force space between paragraphs; use of `\noindent` or `\indent` to suppress or alter LaTeX's paragraph indentation; and the addition or suppression of blank pages between chapters.

Don't. If you're not sure whether your document looks right, download one of the typeset sample thesis or Clinic reports from https://www.math.hmc.edu/computing/support/tex/sample-report/. If you're still not sure, put a copy of your LaTeX code in your math account and send e-mail to the department's sysadmin, who will take a look at it and let you know if it's okay.

C.2.2 The LaTeX Reference System

We sometimes see hard-coded references to equations, sections, chapters, figures, tables, and other elements for which LaTeX provides the `\label` and `\ref` mechanisms. Coding by hand works fine so long as you always remember to go back and correct the references after moving things around, adding new elements, and so on. It's much, much simpler to let the computer do the tedious work of tracking these changes, especially as these values can be tricky to keep track of.

C.2.3 Recreating Existing Structural Commands/Environments

Some people construct proofs and other mathematical structures by hand rather than using the handy environments and other tools provided by the AMS classes and packages.

Don't. If you're trying to do something that seems like the sort of thing that other people would be likely to want to do, but you're not sure where to look for their code, ask.

Google is a good source for answers to questions about formatting. You can also consult any of the books that we recommend on LaTeX. George Grätzer's *Math into LaTeX* (2000) and its replacement, *More Math into LaTeX* (2007), include lots of examples with both the typeset version of something and the source code that produced it.

```
... and so we suggest that
\[
E = mc^{2}
\]
requires us to understand...
```

... and so we suggest that

$$E = mc^2$$

requires us to understand...

Figure C.1 A Run-In Equation.

C.2.4 Not Using Whitespace

Whitespace is your friend! There is no advantage to cramming things together when writing LaTeX code—LaTeX formats text according to its own rules, which includes collapsing multiple spaces or blank lines, which allows you to use whitespace in your code to help make things more clear to you or an editor.

Whitespace and indentation will not only help you figure out what's going on, but it will also make it easier for anyone else reading your code (such as your advisor or the department's system administrator (who will be vetting your code prior to your thesis being accepted)).

Get in the habit of using whitespace now. You'll be glad when you have to reuse some of your older LaTeX code, and any collaborators or editors you'll have in the future will also be very grateful.

C.2.5 Whitespace in Wrong Places

Often an equation, a list, or a code example should appear midparagraph, like a series of comma-separated phrases would. In these cases, you should not have blank lines between the introductory phrase and the `\begin` command for your environment or after the `\end` environment command and the beginning of the additional text.

Figures C.1 and C.2 show a run-in equation and a run-in list, respectively. Figure C.3[1] shows a list where the last item is the last thing in the paragraph, and we begin a new paragraph by including a blank line between the end of the list environment and the new paragraph.

[1]The poem is William Butler Yeats's "The Second Coming", first published in 1920. In most cases, extracts from poetry should be set using the `verse` environment.

```
... therefore,
\begin{enumerate}
\item The first supposition;
\item The second supposition;
\end{enumerate}
suggests that we ...
```

... therefore,

1. The first supposition;
2. The second supposition;

suggests that we ...

Figure C.2 A Run-In List Environment.

```
... thus Yeats suggests that
\begin{itemize}
\item Things fall apart
\item The center cannot hold
\item Mere anarchy is loosed
  upon the world
\end{itemize}

We believe that this conclusion
is, however, a bleak
interpretation....
```

... thus Yeats suggests that

- Things fall apart
- The center cannot hold
- Mere anarchy is loosed upon the world

We believe that this conclusion is, however, a bleak interpretation....

Figure C.3 A Concluded List Environment. (With apologies to William Yeats.)

NFSS Font Command	Obsolete Font Command
`\textit{xxx}`	`{\it xxx}`
`\textbf{xxx}`	`{\bf xxx}`
`\textsc{xxx}`	`{\sc xxx}`
`\textsf{xxx}`	`{\sf xxx}`

Table C.1 New-Style vs. Old-Style Font Commands.

C.2.6 Using Old-Style $$ Display Math Delimiters

LaTeX documents use the `\[ \]` delimiters for setting display math environments in place of the older, TeX-style `$$` delimiters.

There are two major reasons to use the LaTeX-style delimiters rather than the TeX-style delimiters. First, LaTeX's spacing is different than TeX's, especially when some of the AMS packages are in use (as they are with our thesis and Clinic classes). Second, it's not possible for LaTeX classes and packages to alter the display math environments created with `$$...$$`, so math set in different environments may not be typeset consistently.

The basic rule here is to always use the LaTeX commands.

C.2.7 Using Old-Style Font Commands

A related problem is the use of older, deprecated font commands rather than the "new" font commands (from the "New Font Selection Scheme" or NFSS, introduced in 1989). Table C.1 provides a quick overview of commonly used font commands.

The NFSS commands are better than the old-style commands because they are designed to operate orthogonally to one another, and can, therefore, be nested and generate the proper results. Thus

```
\textsc{\textsf{\textbf{String}}}
```

should (given the selection of fonts and font packages that include all the necessary font shapes and weights) produce the word "String" set in bold, small-capital, sans-serif type; that is, as

STRING

Getting the same results with the old-style commands is left as an exercise for the reader.

C.2.7.1 Emphasis or Italics?

Ideally, you should use `\emph{xxx}` rather than `\textit{xxx}`, as the `\emph` command does the right thing when nested; that is,

```
\emph{My favorite movie?  Hmm.  Probably
  Wenders' \emph{Until the End of the World}},
  she thought.
```

would typeset *Until the End of the World* using roman glyphs so it would stand out properly from the rest of the italicized type; as in

> *My favorite movie? Hmm. Probably Wenders'* Until the End of the World, she thought.

C.2.8 Typing Punctuation

LaTeX uses several special code sequences to represent some common punctuation characters. It's important to get these right and to be consistent in their use to produce an attractive document.

C.2.8.1 Quotation Marks

"Quotation marks" are produced using the `‘‘` and `’’` ligatures for double-quotes; `‘` and `’` for single quotes.

The quotation mark key, `"`, produces ", or "something" in the typeface used by our classes. In other typefaces the left or opening double-quote character may look odder and more out of place (because it might be set as "straight quotes" or the right or closing double-quote glyph's design might be differ more strongly from that of the left double-quote).

C.2.8.2 Dashes

There are three types of dashes: single dash or hyphen, typed as `-`, used to connect words in a adjectival phrase such as "parallel-processing system"; *en dashes*, typed as `--`, used to connect ranges of numbers, as in "10–20"; and *em dashes*, typed as `---`, and used for longer breaks or interjections—such as this one.

Important Note

Note that none of these dashes should have whitespace surrounding them—they should be run into surrounding text.

C.2.8.3 Periods and Spacing

LaTeX's handling of periods in different contexts can cause subtle spacing issues. By default, LaTeX assumes that a period marks the end of a sentence, and adds a bit of extra "glue" to that space that can "stretch" if needed to help the line fill the available space. But LaTeX can't tell when a period is used to end a title or an initial, and will insert a sentence space instead of a regular word space. Depending on its context, "Dr. John A. Smith" might appear with a wider space than you might expect or might break the line between the title, the initial, or the surname.

You can insert a *tie* to prevent LaTeX from using a stretchy space or splitting things across a line break by replacing the space with a tilde (~) character. Instead of `Dr. A. Smith`, you might type `Dr.~John~A. Smith`, which would produce "Dr. John A. Smith".

C.2.9 Hyphenation Issues

TeX's hyphenation engine is very powerful, but as new words are introduced to the language, the hyphenation patterns that it uses must be updated. In most cases, you won't want to wait until a new release is made, so LaTeX provides some tools to allow you to specify hyphenation patterns for words.

For words that you use a lot, you will want to add a `\hyphenation` command in the preamble of your document. This command takes a space-separated list of words with syllables indicated with dashes; for example,

```
\hyphenation{da-ta-base ho-meo-stat-ic}
```

For words that are breaking improperly (i.e., the word is not hyphenated or not hyphenated properly, and part of it sticks out into the margin (beyond the edge of the text block)), you can try adding "optional hyphenation" commands, `\-`, at the correct locations within the word. If neither approach helps, you may need to rewrite the sentence in order to make it possible to typeset the paragraph correctly.

Note that one way to see "overfull" lines easily is to add the `draft` document-class option to your `\documentclass` command. In draft mode,

LaTeX will put black "slugs" over the text that extends into the margins, making it easier to see.

Fixing hyphenation issues is a fine-tuning step, usually left until very near the end of your writing process, when you worry about where tables or figures fall, how pages break, and so on.

C.2.10 Formatting Tables

Most of the "tables" provided as examples in books and other introductory material on LaTeX are really "tableaus". They use extra vertical and horizontal lines that create distracting visual noise and don't provide any information.

Most published books typeset tables in a sparser, cleaner style, using whitespace to separate columns and lines. Excellent examples and guidelines are presented by *The Chicago Manual of Style* (University of Chicago Press, 2003). This style is supported by the booktabs package for LaTeX, which is loaded by default in our hmcclinic and hmcthesis document classes.

Section 3.5.2.7 talks briefly about the use of the booktabs package for formatting tables. Examples of how to use the booktabs package's commands are included in our sample thesis and Clinic report (see Section 3.3.2) as well as in other LaTeX materials produced by the department.

C.3 BibTeX Problems

Section 3.7 covers our recommended BibTeX usage.

There are three basic categories of issues that we see with the use of BibTeX in theses.

The first, of course, is not using BibTeX at all. The second is issues with formatting the content of BibTeX databases. The third is not fixing problems flagged by the `bibtex` program.

C.3.1 Formatting BibTeX Databases

Section 3.7.4 discusses some recommended ways to format bibliographic databases. The key idea is to record as much information as possible in your database. Ideally, you want to have the most complete records that you can possibly put together, with full names for all authors, and complete, properly punctuated, and capitalized titles.

Bibliographic styles vary by publication, with some expecting full names of authors and editors and some using initials. Having full names supports

either style, but only having initials means that you won't be able to fulfill the requirements of publications that demand full names.

Similarly, some bibliographic styles use whatever capitalization you supply for titles, whereas others might downcase all but the first word. One problem with sloppy titles appears when your bibliography records are all sentence-capped but the house style calls for headline-capping. Another is when proper names are used but aren't protected using braces. For example, `{F}ourier`, not `Fourier` (or, worst, `fourier`).

C.3.2 Ignoring BibTeX Errors and Warnings

When BibTeX is run on an auxiliary file, it pulls records out of the associated bibliography database file and formats them according to the specified bibliography file. Along the way, it will generate warnings about missing fields or misformatted information, and errors when important information is missing or duplicated.

Many submitted theses or Clinic reports included bibliography entries with missing dates, journal titles, or other important information. Some contained more than one entry with the same key. And there were other, similar problems.

BibTeX is pretty good about telling you exactly what's wrong, including the line number and the record that has the problem, so there's really no excuse for not fixing these issues.

Appendix D

Guidelines for Maintaining a Technical Project Notebook

A project notebook is

> the medium by which a researcher organizes and focuses his thoughts, keeps records of daily work, and provides a record of tasks performed. Notebooks record investigative history, both successes and failures, so that research does not have to be repeated. These notebooks represent the factual and legal documentation of work performed by technical personnel within the organization.
>
> —Dawn Duperault, School of Library and Information Studies, Texas Woman's University, 1995

D.1 Why Keep a Project Notebook?

While the ubiquitousness of laptop computers, Wikis, note-keeping software, internal blogs, and other electronic record-keeping technologies has caused the whole culture of handwritten project notebooks to evolve, these physical, paper-and-ink documents still play a vital role in the research process. Notably, the paper-based project notebook is still essential in the patent process and provides a time-honored (and legally admissible) way of documenting scientific discovery.

Early in the semester (by Clinic/Sponsor Orientation Day, xxx., At the latest), you and your teammates, your advisor, your project manager, and

your sponsor's liaison should meet and agree on the exact nature of how project notes will be kept for your Clinic project. The guidelines for written notes included in this chapter are the traditional rules for such documents, and following these rules or similar variants will be expected by many of our sponsors.

Keeping a project notebook can be justified on at least five levels:

- Technical
- Personal
- Academic
- Professional
- Legal

D.1.1 Technical Reasons

A project notebook documents the progress of your experiments (basic research, product, or software development), observations, and results.

A notebook makes the process of archiving data more efficient. It also creates a resource for interpreting your results and planning future experiments.

D.1.2 Personal Reasons

Personal reasons for maintaining a project notebook include

- Developing your skills of observation, ordering evidence, and drawing inferences as well as your descriptive powers
- Helping you "remember" everything that you do for the entire length of the project
- Avoiding duplication of effort
- Avoiding repetition of erroneous procedures
- Insuring against the loss of valuable data

D.1.3 Academic Reasons

Academic reasons for maintaining a project notebook include

- Providing you with a solid record of your participation in a project
- Documenting the contributions of team members on a project
- Providing you with the information you will need to prepare technical reports or journal articles at a later date

D.1.4 Professional Reasons

Professional reasons include

- Supporting the validity of results reported to peers and sponsors
- Providing proof of fulfillment of contracts
- Maintaining continuity when a long-term project is passed from one team to another

D.1.5 Legal Reasons

Legal reasons include

- Establishing a permanent record to prove the authenticity of the work
- Availability as evidence to prove a company's or college's right to obtain a U.S. patent
- Defending a patent against those who contest it
- Defending a patent against infringement by others

D.1.5.1 Inventions and Patent Law

The dates of "conception" and "reduction to practice" are essential to obtaining and defending a patent.

Date of Conception Record the novel idea in the notebook as quickly as possible

Evidence of Due Diligence Continue to record every instance when you return to the idea

Reduction to Practice Established by the actual construction and successful testing of device incorporating the invention

D.2 Notebook Management

In order to serve as evidence, a project notebook must be difficult or impossible to fake. Thus, it should be permanently bound with numbered pages (so that adding pages is difficult or impossible, and missing pages will stand out).

Each team member should have her own notebook, and investigators should only make entries in their own notebooks. Having multiple investigators record the same information is okay—you will have duplicate records should one notebook be damaged or lost.

Legibility is important—the greatest discovery in the world might not be upheld if no one but the investigator can read the description in the project notebook!

All entries should be made in ink. Ideally, you should avoid using different colors of ink on the same page (or even on the same day). Using the same color ink helps make it clear that the entries were made at roughly the same time.

For the same reason, you should avoid leaving blank pages between entries or even significant amounts of blank space on a single page. Blank pages or spaces could be used to modify your records later.

All entries should be made consecutively, in the order in which the events that they describe occurred. Save nonchronological organization methods for your reports or articles.

Reserve the first few pages of your notebook for use as a table of contents—update that section daily with pointers to significant events, when new days' entries start, and so forth.

Don't try to erase or excise errors. Instead, cross them out with a simple line (i.e., don't scribble wildly over the deleted material or black it out with a marker) and write the corrected information in after the deleted section.

If you discover an error in your work on a following day, cross out the erroneous material and add a note that points to the corrected information. Include the page number where the corrected material is written and sign and date the deletion.

Notebook Entries To Date	Contact Information
Meeting Minutes	Project Description
Individual Log	Goals and Objectives
Group Log	Schedule
Name and Location of Computer Files	Short-Term Plan
Correspondence	Long-Term Plan
References	Work Schedule
Team Members	

Table D.1 Notebook Entries to Date.

D.2.1 Signing and Dating Pages

Every page of your notebook should be signed and dated. Table D.1 includes a list of specific types of information that should be dated independent of their position on a page.

D.2.2 Supplemental Material

Materials that are generated by equipment or computer programs that support your work should be permanently inserted into your notebook at the appropriate location. They should be attached using tape or glue along their left-hand margin, and should be signed and dated with the signature extending across the attached material and onto the notebook page.

D.2.3 Computer Records

Because the time and date stamps on computer files can not be guaranteed to be unfalsifiable, you should print a hard copy of your code on a weekly basis, have the code reviewed by a witness, sign and date each page of the code, and archive the printouts.

D.3 What to Record

The project notebook is a daily record of your activities—what you are thinking and doing.

- Ideas, calculations experimental results, and observations should be entered into the notebook on the same date they occur

- Results include more than data. Record questions, reasons for changes in experimental plans, evolving thoughts, awareness of potential of discoveries, and so forth.
- Exactly what was done—protocols, design of experiments, calculations, etc.
- Why it was done (e.g., objectives and goals)
- Who initially suggested the idea
- Who did the investigation/exercise/task/test
- When and where the task was performed
- The results obtained
- The conclusions drawn

At the end of a work day,

- Summarize what you have accomplished
  - Conclusions are not necessary
  - The summary helps maintain continuity, reveals where the work left off and how it might resume
- Periodically have your work witnessed
- Store your notebook in a safe place

D.3.1 Witnesses and Reviewers

Must be adults, preferably over 21, with the technical competence to understand the details of the subject

Must be an impartial person—not a co-inventor

Sign and date every page reviewed:

Witnessed and understood by _____, Date _____.

D.4 Things to Avoid

Problems with your notebook may reduce its value in legal or academic disputes or damage the credibility of your laboratory or team.

These problems include

- Illegible entries
- Unsigned or undated pages
- Pages that have not been witnessed
- a long delay between the signing of the page by the inventor and the witness
- Consecutive notebook pages that are not dated in chronological order
- Missing notebook pages
- Erasures and deletions

D.5 Conclusion

Remember, a project notebook

- Says exactly what was done, when it was accomplished, and by whom it was performed
- Reveals the thought process of the investigator in the conception of ideas and the interpretation of results
- Enables someone else to repeat the work at some future date
- Is durable and verifiable

D.6 Acknowledgments

This chapter is based on slides from a presentation made by Professor Kerry Karukstis to the mathematics Clinic on November 20, 2001. Additional material was written based on BookFactory (2007); Caprette (2005); Garabedian (1997); Purrington (2011).

Appendix E

Creating Your Thesis Website

One of the requirements of the thesis proram is that you maintain a one-page website that includes information about your thesis project. We will later archive this page, so its format is important, and we provide a template file for you to use.

E.1 Setting Up Your Webpage

There are two ways to get started, the easy way (which we recommend) and the hard way (which requires more work within the Linux shell).

E.1.1 The Easy Way

The easy way to set up your webpage is to run the script

`setup-seniorthesis-website`,[1]

which will create all the necessary directories in your home directory and set the permissions correctly so that your page will be viewable with a web browser pointed at a URL similar to

`https://www.math.hmc.edu/~`*`username`*`/thesis/`,

where *username* is your login name on the math department's Linux systems.

[1]This script is located in `/shared/local/bin`.

E.1.2 The Hard Way

If you want to have more control over the process, you can create the various directories you'll need (`~/public_html`, `~/public_html/thesis`) and then download the template file, `webtemplate.html`, from the senior thesis website. Save a copy of that file as `~/public_html/thesis/index.html`.

In order for your page to be readable by the web server, you'll need to make sure that the Unix permissions are set correctly. Directories should be set to `755` (`drwx-r-xr-x`) with `chmod 755 directory`, as in `chmod 755 ~/public_html`. File permissions are set the same way, but they should be set to `644`.

Your home directory is a special case. Because you probably don't want random people able to be able to see the files and directories at the top level of your home directory, you need to set the permissions to `711` (`dwrx--x--x`) with `chmod 711 ~`.)

The script mentioned in Section E.1.1 does all this basic work for you.

E.2 Next Steps

Now that you have the basic directories and the "index" file (the default file the web server uses when displaying a directory URL), you can edit the index file and add a picture of yourself and an preliminary abstract, if you have one.

E.2.1 Editing `index.html`

Once you've run the script, you will still need to edit the `index.html` was created. This file is located in `~/public_html/thesis` (in your home directory; the ~ is a Unix shortcut).

You should change the generic fields in the page to match your name, the title of your thesis, the names of your advisor(s) and reader(s), and so on.

E.2.2 Adding a Photograph

You will also need to supply a JPEG image of yourself named `mugshot.jpg`. We generally take these pictures in one of the first thesis seminar meetings, but if you have a picture of yourself that you prefer, you can use that instead. Pay attention to the sizes, defined in the `index.html` file, and also consider

whether you really want to have a crazy image representing you in the department's archives five, ten, or more years from now.

E.3 The Rest of the Year

As the year progresses, you will be modifying your webpage to add more information and links to updated copies of your thesis. In particular, you will add a copy of your midyear report and a link in the index page.

At the end of the year we will be archiving a version of your thesis webpage. You should check it over to make sure that it includes the most recent copies of

- Your thesis
- An abstract
- Links to posters or other materials related to your thesis that you think people might be interested in

Appendix F

How to Give a Good Talk

By Joseph A. Gallian.
Reprinted from *Math Horizons* magazine (1998).

∞ ∞ ∞

Presentations by undergraduates at professional meetings have increased dramatically in recent years. And more and more undergraduates make presentations in classes, seminars and colloquiums. Learning how to give a good presentation is a valuable skill that many students will find useful in connection with their employment. Baseball manager Joe Torre once said that the teams that are most successful are the ones that do the little things well. Doing the little things well is the secret to giving good talks as well. Here is my advice on how to do the little things well when giving a talk.

Preparation

- Inquire about the target audience.
- Determine the level of knowledge of the target audience.
- Choose a subject that will appeal to the intended audience.
- Don't overestimate what the audience knows about your subject.
- Don't try to do too much.

- Use simple examples and concrete special cases. A "non-example" often helps to clarify a concept. (For instance, if you use the integers modulo 7 as an example of a finite field, be sure to point out that the integers modulo 6 is not a field and why.) Use intuitive definitions rather than technically correct ones. Avoid details. Mention applications.
- Choose a short and informative title. (Cute titles are usually poor titles.) "On a Theorem of Hilbert" is too vague. "On Hilbert's Basis Theorem" is short and informative. In your abstract, indicate the level of the talk. (Examples: "This talk is intended for a general audience"; "This talk is suitable for those who have had linear algebra"; "This talk is suitable for those who have had real analysis.")
- Keep technical terms and unfamiliar symbols to a minimum. When you do use them remind your audience of their meaning from time to time.
- When possible, relate your topic to other fields.
- Provide a context for your talk. Explain how you got interested in the subject. Mention others who have worked on the subject of your talk.
- Use transparencies. A chalkboard talk comes across as slow moving. Make the transparencies up well in advance and number them.
- Use multi-colors in preparing your transparencies. Blue, green, red and purple show up best. Avoid orange and brown. Use permanent ink (water soluble smudges easily). You can fix mistakes with rubbing alcohol.
- Write very large or use a large font (even for a talk in a small room). If you reproduce printed material, enlarge them for the transparencies. Use color photocopy machines to make color transparencies.
- Prepare a crisp beginning. Perhaps start with a question, an application or a prop.
- Don't put much on the transparencies. Use key words and phrases instead of entire sentences. Avoid filling transparencies with equations and formulas. (Don't compute in public.)

Appendix G

Official Faculty Names

Our faculty use particular arrangements of their names for publications, which makes it easier to keep track of their publications across different journals, books, and so on.

Be sure that you use the official names from this list for your faculty advisor and second reader for those commands in your master LaTeX file (see Section 3.3.1) as well as for any bibliographic references to their work. If your advisor or reader isn't listed in this appendix, check with them to find out what name they use for publications.

Jorge Aarão

Arthur T. Benjamin

Andrew J. Bernoff

Robert L. Borrelli

Erin Byrne

Naiomi Cameron

Alfonso Castro

Courtney Coleman

Matt Davis

Lisette G. de Pillis

Robert Flemming

Daniel L. Goroff

Weiqing Gu

Yixin Guo

Sanjay Gupta

Melvin Henriksen

Anette E. Hosoi

Jon Jacobsen

Dagan Karp

Kimberly Kindred

Henry A. Krieger

Greg Levin

Rachel Levy

Alfonso Limon

Thomas LoFaro

Daniel G. Mallet

Susan E. Martonosi

Jeffrey J. Miller

Michael E. Moody

Seema Nanda

John W. Neuberger

Michael E. Orrison

David Panton

Allon Percus

Nicholas Pippenger

Michael R. Raugh

Elissa Schwartz

Francis Edward Su

Lesley A. Ward

Ursula Whitcher

Alvin White

Talithia D. Williams

Darryl H. Yong

Bibliography

American Mathematical Society. 1999. *User's Guide for the amsmath Package*. American Mathematical Society.

Berlin, Daniel, and Garrett Rooney. 2006. *Practical Subversion*. Apress, 2nd ed.

Berry, Karl, and Olaf Weber. 2008. *Kpathsea Library*. Included with TEX-Live installation. For version 3.5.7.

BookFactory. 2007. BookFactory laboratory notebook guidelines. Website. URL http://www.bookfactory.com/special_info/lab_notebook_guidelines.html. Viewed on 2007 August 30.

Bringhurst, Robert. 1996. *The Elements of Typographic Style*. Vancouver, BC: Hartley & Marks, 2nd ed.

Caprette, David R. 2005. Guidelines for keeping a laboratory record. Website. URL http://www.ruf.rice.edu/~bioslabs/tools/notebook/notebook.html. Viewed on 2006 June 14.

Carlisle, David P. 1999. *Packages in the 'graphics' Bundle*.

Chacon, Scott, and Ben Straub. 2015. *Pro Git*. 819, Apress. URL https://progit.org/. Home page for Pro Git, which includes links to download the book in various formats as well as links to purchase a physical copy. Page numbers taken from the PDF edition.

Collins-Sussman, Ben, Brian W. Fitzpatrick, and C. Michael Pilato. 2004. *Version Control with Subversion*. Sebastopol, CA: O'Reilly & Associates. URL http://www.svnbook.org/.

The "Subversion Book".

Connelly, Claire. 2006. TEX and LATEX resources. URL https://www.math.hmc.edu/computing/support/tex/. Accessed on 2006 May 2.

Crews, Kenneth D. 2013. *Copyright and Your Dissertation or Thesis: Ownership, Fair Use, and Your Rights and Responsibilities*. ProQuest. URL http://media2.proquest.com/documents/copyright_dissthesis_ownership.pdf. Licensed to the public under a Creative Commons BY-NC license. Viewed 2016-04-22.

> Discusses copyright issues related to theses and dissertations. Crews was the Director of the Columbia University Copyright Advisory Office, and much or all of this material is available on their website, www.copyright.columbia.edu.

Doumont, Jean-Luc. 2009. *Trees, Maps, and Theorems: Effective Communication for Rational Minds*. Kraainem, Belgium: Principiæ.

Dupré, Lyn. 1998. *BUGS in Writing*. Reading, MA: Addison-Wesley, 2nd ed.

Editor. 1986. Hyphenation exception log. *TUGboat* 7(3):145.

Fear, Simon. 1997. *Publication Quality Tables in LATEX*.

Fenn, Jürgen. 2006. Managing citations and your bibliography with BIBTEX. *The PracTEX Journal* 1(4). URL http://www.tug.org/pracjourn/2006-4/fenn/fenn.pdf. Published online.

Gallian, Joseph. 1998. How to give a good talk. *Math Horizons* .

Garabedian, Todd E. 1997. Laboratory record keeping. *Nature Biotechnology* 15:799–800. URL http://www.wiggin.com/pubs/articles_template.asp?ID=102187242000. Republished at Wiggin and Dana website. Viewed on 2006 June 14.

Grätzer, George. 2000. *Math into LATEX*. Boston: Birkhäuser, 3rd ed.

———. 2007. *More Math into LATEX*. New York: Springer Science+Business Media, 4th ed. URL http://www.springerlink.com/content/978-0-387-32289-6/.

Higham, Nicholas J. 1998. *Handbook of Writing for the Mathematical Sciences*. SIAM, 2nd ed.

Jaffe, Greg. 2000. What's your point, Lieutenant? please, just cut to the pie charts: The Pentagon declares war on electronic slide shows that make briefings a pain. *The Wall Street Journal* URL http://interactive.wsj.com/archive/retrieve.cgi?id=SB956703757412556977.djm. From WSJ Online; originally appeared April 26, 2000.

Knuth, Donald E. 1993. *The TEXbook*. Addison-Wesley.

Knuth, Donald E., Tracy Larrabee, and Paul M. Roberts. 1989. Mathematical writing. *Mathematics Association of America Notes* 14.

Kopka, Helmut, and Patrick W. Daly. 2004. *Guide to LATEX*. Tools and Techniques for Computer Typesetting, Addison-Wesley, 4th ed.

Krantz, Stephen G. 1997. *A Primer of Mathematical Writing*. American Mathematical Society.

Lamport, Leslie. 1994. *LATEX: A Document Preparation System*. Addison-Wesley, 2nd ed.

Markey, Nicolas. 2005. Tame the BeaST: The B to X of BibTEX. Online. URL http://yum.math.hmc.edu/ctan/info/bibtex/tamethebeast/ttb_en.pdf. Accessed 2007 August 30.

> Excellent, in-depth introduction to BibTEX. Not only a good guide to day-to-day use, but also examines many of BibTEX's internals, which helps the reader understand *why* doing things one way rather than another is better.

Mathematics Subject Classification. 2010. Mathematics subject classification 2010. Website. URL http://msc2010.org/. Visited 2010 July 16.

Mittelbach, Frank, Michel Goossens, Johannes Braams, David Carlisle, Chris Rowley, Christine Detig, and Joachim Schrod. 2004. *The LATEX Companion*. Tools and Techniques for Computer Typesetting, Boston: Addison-Wesley, 2nd ed.

Purrington, Colin. 2006. Sample scientific poster. Online; viewed 2006 April 25. URL http://www.swarthmore.edu/NatSci/cpurrin1/postertemplate.ppt.

> A PowerPoint template for a scientific poster, with information about the sorts of things that should and should not be included in such a poster.

———. 2011. Maintaining a laboratory notebook. Online; viewed 2011 September 14.

> Illustrated essay on how and why you should maintain a laboratory notebook. Includes useful information on selecting an appropriate notebook, as well as advice on pens that are insoluable in common laboratory and field solvents (water, methanol, etc.) based on experimentation.

———. 2016. Designing conference posters. Online; viewed 2016 August 25. URL http://colinpurrington.com/tips/poster-design.

> An excellent overview of how to create a poster for a scientific meeting. Also includes information on how to structure your presentation, how to dress, and other good ideas to help you and your poster be remembered.

Reynolds, Garr. 2005. Presentation Zen: The "Lessig Method" of presentation. Website. URL http://www.presentationzen.com/presentationzen/2005/10/the_lessig_meth.html.

———. 2011. *Presentation Zen: Simple Ideas on Presentation Design and Delivery*. Voices That Matter, New Riders, 2nd ed.

Swanson, Ellen, Arlene O'Sean, and Antoinette Schleyer. 1999. *Mathematics into Type*. Providence, RI: American Mathematical Society, updated ed. URL ftp://ftp.ams.org/pub/author-info/documentation/howto/mit-2.pdf.

> Not a LaTeX book, but one that talks about more fundamental questions of how to typeset (or even just type) mathematics. Topics such as breaking mathematical expressions, alternative ways of writing expressions, alignment, copyediting, and general writing advice as well. Available in PDF from the AMS (free).

Tufte, Edward R. 1983. *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press.

———. 1990. *Envisioning Information*. Cheshire, CT: Graphics Press.

———. 1997. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Cheshire, CT: Graphics Press.

———. 2006a. *Beautiful Evidence*. Cheshire, CT: Graphics Press.

———. 2006b. *The Cognitive Style of PowerPoint: Pitching Out Corrupts Within*. Cheshire, CT: Graphics Press, 2nd ed. See http://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=0001yB&topic_id=1 for a discussion thread about this essay on Tufte's website.

Turing, Alan M. 1936. On computable numbers, with an application to the *Entscheidungsproblem*. *Procedings of the London Mathematical Society* 42(2):230–265.

TWG-TDS. 2004. A directory structure for TEX files. Website. URL http://tug.org/tds/.

University of Chicago Press, ed. 2003. *The Chicago Manual of Style*. Chicago, IL: University of Chicago Press, 15th ed.

The standard American work on typesetting style etc. (CHe)

Walsh, Norman. 2002. Help on BIBTEX names. Website. URL http://nwalsh.com/tex/texhelp/bibtx-23.html.