Mathematical Foundations for Computer Science (15-151)

CMU Summer Academy for Mathematics and Science, Summer II 2017

Class meets on weekdays at 8:00–9:30am in Porter Hall A19

`http://math.cmu.edu/~cnewstea/teaching/15-151-N17/`

Instructor: **Clive Newstead**
`cnewstead@cmu.edu`
Wean Hall 8205

TA: **Alp Müyesser**
`nmuyesse@andrew.cmu.edu`
Doherty Hall 1217 (Sun–Thu 6:30–9:00pm)

1 Course content

Leaning objectives

Upon successful completion of this course you should be able to:

(1) Accurately use standard mathematical notation and terminology in mathematical writing, including symbolic logic, sets and set operations, functions, relations, binomial coefficients and factorials;

(2) Identify feasible techniques for proving a proposition based on its logical structure;

(3) Write correct, clear and precise mathematical proofs, in an appropriate level of detail, of both familiar and unfamiliar results from pure mathematics in the areas covered;

(4) Recognise and apply standard proof techniques, including direct proof, contradiction, contraposition, weak and strong mathematical induction, and the well-ordering principle;

(5) Accurately recall definitions and state and prove the main theorems in the mathematical areas covered;

(6) Solve standard unseen problems in the mathematical areas covered by identifying which results from the course are appropriate and applying them accurately;

(7) Typeset basic mathematical documents using LaTeX, including the mathematical notation used in the course, sections, references and labels, and basic formatting.

List of topics (tentative)

- **Fundamentals.** In the first couple of weeks, we will set the scene for the rest of the course—and, indeed, the rest of pure mathematics. We will start with an introduction to some basic proof techniques, including direct proof, contradiction, contraposition and induction; then we will hitch up the abstraction with a formal introduction to symbolic logic and the fundamental notions of *sets* and *functions*.

- **Enumerative combinatorics.** The first topic we will explore in some detail is *enumerative combinatorics*, which is a fancy term for 'counting'. We will develop techniques for counting sets of objects based on their descriptions, using the mathematical machinery of functions. Enumerative combinatorics is particularly useful in theoretical computer science, for example in the analysis of algorithms.

- **Relations and order.** Relations allow us to compare mathematical objects. We will treat relations as mathematical objects in their own right, with a particular focus on order relations, which behave in some ways like the '$\leqslant$' relation on the number sets. Of particular interest to us are Boolean algebras, which can be used to study electrical circuits in the abstract.

- **Induction.** Mathematics and computer science are littered with structures that can be built up out of basic elements by repeatedly applying certain rules. An example is the set of natural numbers, which is built up from the number zero by repeatedly adding one. Other examples are lists, streams, strings, trees and logical formulae. We will familiarise ourselves with such structures, and develop a systematic way of proving facts about them using *proof by induction* in various guises, including structural induction and well-founded induction.

- **Additional topics.** If time permits, the last few days of class will be a survey of some areas of pure mathematics and theoretical computer science that you will likely encounter in future courses. The choice will be yours—we will vote on a topic as a class. Example topics are number theory, probability theory, infinite sets, recursion theory, computational complexity, or abstract algebra.

- **LaTeX.** Typesetting mathematical documents is very difficult in most Office-style 'what you see is what you get' (WYSIWYG) editing software—it is easier to input mathematical notation as code. The industry standard for typesetting mathematics is LaTeX (pronounced 'lay-tek' or 'lah-tek'), in which all formatting and mathematical notation is entered as code. Throughout the course, we will have LaTeX workshops, and you will typeset some of your mathematical work using LaTeX.

Reading materials

As course notes we will use my book (in progress):

- *An infinite descent into pure mathematics* by Clive Newstead.

This book, and any other required reading material, will be provided to you free of charge.

The following texts are also useful, but are not required reading:

- *A Concise Introduction to Pure Mathematics* by Martin Liebeck;
- *How to Prove It: A Structured Approach* by Daniel J. Velleman;
- *Invitation to discrete mathematics* by Jiří Matoušek and Jaroslav Nešestřil.

An interesting introductory-level survey of topics in pure mathematics, theoretical computer science and electrical engineering, with emphasis on their importance in computer science, is:

- *The New Turing Omnibus: Sixty-Six Excursions in Computer Science* by Alexander K. Dewdney.

A fun, non-technical discussion about how the skills you acquire when studying abstract mathematics relate to the real world is:

- *How Not to Be Wrong: The Power of Mathematical Thinking* by Jordan Ellenberg.

2 Assessment and grades

All work you do in this course will receive some amount of credit, with a total of 1000 points available. Grades will be assigned according to how many points you are awarded. Below are descriptions of the course assessments and grade assignments.

- **Classwork.** 2 points $\times$ 25 classes = 50 points total.
  - ◇ **What?** Various activities and problem-solving exercises in class.
  - ◇ **Why?** To keep you engaged with the material as it is presented and to put into practice the social aspect of mathematics.
- **Homework** 35 points $\times$ 10 homeworks = 350 points total.
  - ◇ **What?** Several challenging questions covering material covered in class. Graded for mathematical correctness and proof-writing quality.
  - ◇ **Why?** To give you the opportunity to show off what you've learnt, and to allow me to give you individual feedback on your progress.
- **Quizzes.** (30 points $\times$ 5 Wednesdays) + (50 points $\times$ 5 Fridays) = 400 points total.
  - ◇ **What?** Exercises and problems to be completed individually in class or recitation.
  - ◇ **Why?** To test your knowledge of the material and to provide structure to your (limited) study time out of class.
- **LaTeX project.** 50 points.
  - ◇ **What?** Short write-up of a document using LaTeX, putting into practice the mathematical typesetting skills you'll learn in the first two weeks.
  - ◇ **Why?** To familiarise you with using LaTeX to typeset mathematics, and to prepare you for the course project, which must be typeset.
- **Course project.** 150 points.
  - ◇ **What?** Mathematical investigation and write-up about mathematical topic of your choosing.
  - ◇ **Why?** To bring together the skills you have learnt in the course, and to expose you to the kind of mathematics done by mathematicians.

The preliminary grade borderlines are as follows:

A: 900 and over **B:** 800–899 **C:** 700–799 **D:** 600–699

These borderlines might be lowered, but will not be raised; for instance, a score of 800 guarantees you a B grade, even if the borderlines change.

3 Schedule

The following schedule is subject to change.

Date			Sec.	Topic	Hw	Qz
Week 1	Monday	3rd July	—	Placement test		
	Tuesday	4th July	—	*No class*		
	Wednesday	5th July	—	Introduction		
	Thursday	6th July	1.1–1.2	Basic proof techniques	✓	
	Friday	7th July	1.3	Weak and strong induction		✓
Week 2	Monday	10th July	2.1	Symbolic logic	✓	
	Tuesday	11th July	2.1–2.2	Symbolic logic and sets		
	Wednesday	12th July	2.2	Sets		✓
	Thursday	13th July	2.3	Functions	✓	
	Friday	14th July	2.3	Functions		✓
	Sunday	16th July	—	**LaTeX project due**		
Week 3	Monday	17th July	4.1	First look at counting		
	Tuesday	18th July	4.1	More functions	✓	
	Wednesday	19th July	4.1	Even more functions		✓
	Thursday	20th July	4.2	Counting	✓	
	Friday	21st July	4.2	More counting		✓
Week 4	Monday	24th July	5.1	Relations	✓	
	Tuesday	25th July	5.1	Equivalence relations		
	Wednesday	26th July	5.2	Order relations		✓
	Thursday	27th July	5.2	Boolean algebras	✓	
	Friday	28th July	5.2	More boolean algebras		✓
Week 5	Monday	31st July	1.3, 5.3	Inductively defined sets	✓	
	Tuesday	1st August	5.3	Inductive structures		
	Wednesday	2nd August	5.3	Structural induction		✓
	Thursday	3rd August	5.3	Well-foundedness	✓	
	Friday	4th August	5.3	More well-foundedness		✓
	Sunday	6th August	—	**Course project due**		
Week 6	Monday	7th August	TBA	Additional topic		
	Tuesday	8th August	TBA	Additional topic	✓	
	Wednesday	9th August	TBA	Additional topic		✓
	Thursday	10th August	TBA	Additional topic	✓	
	Friday	11th August	—	*No class*		
	Tuesday	15th August	—	**Final grades posted**		

Hw: homework due. **Qz**: in-class quiz.

4 Policies

Academic honesty and integrity

My stance on academic honesty is simple: **all work you submit should accurately reflect your understanding**, and **any help you receive should be acknowledged**. What this means is that if someone were to ask you to explain something you submitted, then you would be able to explain it, and to say how you came to know it. What follows are some more specific descriptions of what this looks like in practice.

Collaboration. Speaking to each other about the course material and homework problems is one of the most effective ways to learn, so this is encouraged. What I ask is that you:

- **Cite your collaborators.** This means that you're giving them credit for their help, and avoids plagiarism issues. Just write a sentence at the top of your homework or project saying who you worked with, e.g. 'I discussed Q4 with Carl Gauss, who said I should use induction'.
- **Write your work up independently.** If you made any permanent records during collaboration sessions, these records should be destroyed before you write up your solution. For example, any notes on whiteboards should be erased, and notes on paper should be thrown away. Direct copying is absolutely forbidden.

External resources. Sometimes you need a little more guidance than is available from your notes, and doing some research can give you the boost you need to understand the material in the course. If you do use external resources, then please:

- **Cite your sources.** If you used a book or website, other than the course textbook or other assigned reading, please say so—just the book title and author, or web page URL, is fine.
- **Write your work up independently.** Close the book or web page and make sure you've understood what you learnt before you start writing—otherwise, all you're really doing is copying, and then your work doesn't reflect your understanding.

Quizzes. Quizzes are the main so-called *summative assessments* of the course, and are intended to be an opportunity for you to demonstrate the knowledge and skills you have acquired. The only resource you should have available to you going into a quiz is your brain, unless otherwise specified—this means you should not have your notes open, you should not speak to others during the quiz, and you should not be looking at other people's answers.

Homework submission

Homework is designed not just to test your knowledge, but also to help you learn. I set deadlines because it's important that you understand the content on the homework before we move on to new material in class. As such, it is in your own interests to do the homework and submit it on time.

Late homework will typically receive a reduction in credit. Homework submitted by the next day will be graded for 80% credit; homework that is two days late will receive 50% credit; and homework that is any later than that will not receive credit.

If there are any special circumstances that mean you absolutely need to submit the homework late, please speak to me as soon as possible—this includes things like illness or planned absence for religious observances. The sooner you ask, the easier it will be for us to find a work-around.

Attendance

Classes will be awesome and fun so you'll probably want to attend, but if that's not enough motivation, remember that 45% of your grade comes from in-class activities and quizzes. Quizzes will be at the beginning of class, too, so please be on time—I won't wait for you!

Attendance is required by the SAMS program, and I am required to take attendance and report absences. If you know you're going to miss a class, **please let me know in advance**! If you missed a quiz, then we will arrange for you to take a make-up quiz; there will be no make-ups for missed classwork.

Accommodations

It is very important to me that my methods of instruction and assessment are fair to everyone enrolled in the course. To this end, **please ensure I'm aware of any accommodations that need to be made ahead of time**! Examples include: extra time in quizzes, large text versions of notes or quizzes, or requiring that I use a microphone.

To ensure fairness, I will need some kind of supporting documentation, either from SAMS or from the CMU Office of Disability Resources, in order to grant accommodations on assessments.

Talk to me

I want you to learn lots and I want you to enjoy taking this course. So that I can find out if this is happening, I encourage feedback—be it positive or negative—on all aspects of the course at any time during the semester.

For example, if something I'm doing is making it difficult for you to learn, then say something before it's too late; or if you particularly enjoyed something we did in class, say so so that we can do it again.

You can do this by just speaking to me or Alp, by sending one of us an email, or by using the anonymous comments form, details of which will be circulated in class. Please bear in mind that I cannot reply individually to anonymous feedback. Giving feedback will in no way affect your grade, positively or negatively.