Syllabus

Summary

CS 81: An introduction to some of the mathematical foundations of computer science, particularly logic, automata, and computability theory. Develops skill in constructing and writing proofs, and demonstrates the applications of the aforementioned areas to problems of practical significance.

Prerequisites: (CS 60 $\vee$ CS 42 $\vee$ CS 52) $\wedge$ ($\neg$Math 55 $\rightarrow$ CS 55) $\wedge$ $\neg$CS 81.

Classes: MW 11am–12:15pm, JA B132

Web Page: `http://www.cs.hmc.edu/cs81`

Course Staff (Office hours to be announced soon)

Prof. Chris Stone	`stone@cs.hmc.edu`
Joseph Dearden	`jdearden@hmc.edu`
Sam Gray	`sgray@hmc.edu`
Sean Laguna	`slaguna@hmc.edu`
Matt Johnson	`mjohnson@hmc.edu`
Bea Mititiri	`bmetitiri@hmc.edu`
Matthew Toal	`mtoal@hmc.edu`
All of us!	`cs81help@cs.hmc.edu`

Course Goals

By the end of this course, you should be able to:

- Learn and use new formal systems;
- Understand the correspondence between formal logic and natural-language proofs.
- Write clear and logically valid proofs;
- Read and write regular languages and context-free (BNF) grammars;
- Reason about code using preconditions, postconditions, and loop invariants.
- Recognize when regular expressions, context-free-grammars, and computation will and will not work for specific problems;
- Explain how to translate among regular expressions, nondeterministic finite automata, and deterministic finite automata;
- Explain the Church-Turing thesis and list models of universal computation;

And more!

Selected Course Topics

First we'll look at logic, both propositional and predicate forms, with an emphasis on the distinction between validity (truth) and provability (derivation). We'll gain some experience in doing proofs in a style known as "natural deduction", both for propositional and predicate logic. This will be a helpful way to outline proofs for the rest of your careers.

We will next review finite-state automata and their relationship to languages and regular expressions. We'll look at the applications and limitations of these machines, then move on to the more powerful pushdown automata, which correspond to context-free languages (generated by grammars of the type you studied in CS 60).

Then we'll look at Turing machines, which are even more computationally powerful. As with other families of machines, Turing machines have their imitations. We will show how to prove that certain problems cannot be solved algorithmically, even ones not apparently related to Turing machines.

Throughout the course we'll be looking at the connections between logic and computability, including the the relationship between logical formulas and program correctness, approaches to automated theorem proving, model checking, decision problems, and Gödel's famous Incompleteness Theorem.

Textbooks

- Michael Huth and Mark Ryan, *Logic in Computer Science, Second Edition*, Cambridge University Press, 2004. ISBN 052154310X.
- Elaine Rich, *Automata, Computability, and Complexity: Theory and Applications,* Pearson Prentice Hall, 2008. ISBN 0132288060.

Getting Help

If you're stuck, *please* contact a grutor and/or the professor. Don't waste time just spinning your wheels.

In addition to Professor Stone's regularly-scheduled office hours, you can drop by his office (Olin 1251) any time his office door is open, or make appointments for other times. (See `http://www.cs.hmc.edu/~stone/schedule.html`.)

You can also use e-mail, especially for short, easily answered questions or statements. You will get the fastest response by sending your message to all course staff via the alias `cs81help@cs.hmc.edu`, so please don't send mail to the professor directly unless the topic is too private for the grutors!

Your Responsibilities

Reading E-mail

E-mails will be sent to the course mailing list **cs-81-l@hmc.edu** (which generally goes to your main campus account) and directly to you. You are responsible for reading all e-mails.

Attendance

I expect you to attend every class and participate actively. Because you will often be asked to do in-class exercises, bring blank paper and writing materials to every lecture.

Collaboration

You are encouraged to discuss the lecture and reading topics with any or all of your classmates. This can range anywhere from informal chats in the hallway to formal study groups.

You are even encouraged to discuss homework problems with other students in this class. However, the caveat here is that **you must come away from homework discussions with understanding in your head, not solutions in physical or electronic artifacts.** Thus you are not allowed to meet as a group and leave with notes or solutions on paper or in your computer; anything you turn in must be written solely by you, on a separate occasion. Finally, **homework discussions must take place on an equal basis**, e.g., not just giving away solutions.

Remember, when it comes to the exams, you're on your own. True understanding will always beat rote learning of specific homework problems.

Course Work and Grading

50%	Homeworks
20%	Midterm
25%	Final Exam
5%	Participation

There will be one homework assignment every week or so. Late assignments will not be accepted except in rare special cases where prior arrangements have been made with the instructor, or in case of unforeseeable crises such major illness.

Preliminary Course Calendar

Week		Topic(s)
1	8/31	Formal Systems and Structural Induction
		Logic
2	9/5, 9/7	(Classical) Propositional Logic
3	9/12, 9/14	(Classical) Predicate Logic
4	9/19, 9/21	Semantics of Predicate Logic
5	9/26, 9/28	Tableaux and Sequents
6	10/3, 10/5	Constructive, Linear, and Modal Logics
7	10/10, 10/12	Reasoning about Code
		Computability
8	10/19	Cardinality
9	10/24, 10/26	Decision Problems, Strings, Languages
10	10/31, 11/2	Regular Languages and Finite Automta
11	11/7, 11/9	Non-regular and Context-Free Languages
12	11/14, 11/16	Push-Down Automata and Turing Machines
13	11/21, 11/23	Reductions and Undecidability
		Logic and Computability
14	11/28, 11/30	TBA
15	12/5, 12/7	TBA

- The above order and timing of topics is subject to change.
- The take-home midterm will be distributed around Week 6 and take the place of a weekly homework assignment.
- The take-home final will be distributed by Friday, December 9 at the latest, and be due back at noon on Friday, December 16.