

## Syllabus

21 January 2010

The goal of this course is to understand the fundamental limits on what can be efficiently computed in our universe and other possible universes. These limits reveal deep and mysterious properties about information, knowledge, and processing, as well as practical issues about what can and cannot be computed.

Two fundamental questions about any problem are:

1. Can it be solved using a given abstract machine? (*computability*)
2. How much time and space are required to solve it? (*complexity*)

We explore these questions by developing abstract models of computing machines and reasoning about what they can and cannot compute efficiently.

**Class Meetings.** Tuesday and Thursday, 2-3:15pm in Olsson 120.

**Course Blog.** All course materials and announcements will appear on the course website, <http://www.cs.virginia.edu/cs3102>. Students should visit it frequently. Most course materials will not be distributed on paper. Students should use the course blog to post questions and comments about assignments and lectures.

**Expected Background.** The official prerequisites for cs3102 are cs2102 (Discrete Mathematics) and either cs2110 (cs201) or cs2220 (cs205), all with grades of C- or better. Students entering cs3102 are expected to be comfortable with proof techniques involving first order predicate logic and induction, reasoning about finite and infinite sets, recursive definitions and problem solving, and programming. Students are also expected to have had some previous exposure to asymptotic notation and algorithm analysis (from either cs1120 or cs2220). If you do not satisfy the prerequisites, you should meet with me to discuss whether you should take the class.

**Textbook.** The required textbook for the course is: Michael Sipser, *Introduction to the Theory of Computation* (Second Edition).

Other recommended books that cover much of the same material are:

- John Hopcroft, Rajeev Motowani, and Jeffrey Ullman, *Automata Theory, Languages, and Computation*. (Third Edition)
- Thomas Sudkamp, *Languages and Machines: An Introduction to the Theory of Computer Science*. (Third Edition)
- Charles Petzold, *The Annotated Turing*. (<http://www.theannotatedturing.com/>)

Two fun general audience books that cover material related to this course are:

- Apostolos Doxiadis and Christos H. Papadimitriou, *Logicomix: An Epic Search for Truth* (<http://www.logicomix.com>), 2009.
- Martin Davis, *The Universal Computer: The Road from Leibniz to Turing*, 2000.

**Topics.** We will cover most of the Chapters 0-5 and Chapter 7 of Sipser's book as well as some topics not covered in the book. The plan for the course lectures is:

- Classes 1-6: proof techniques, finite automata, nondeterminism, regular languages (Chapters 0-1 of Sipser)
- Classes 7-11: pushdown automata, context-free languages, grammars (Chapter 2)
- Classes 12-18: Turing machines, computability (Chapters 3-5)
- Classes 19-24: complexity, NP-completeness (Chapter 7)
- Classes 25-28: hot topics (possibly including cryptography, on-line algorithms, game theory, social networks, randomization, and quantum computing)

This course is a theory course and our primary focus is on abstract, theoretical ideas. We will, however, occasionally touch on relevant applications such as virus detection, bioinformatics, ad auctions, and virtual machines.

**Teacher.** David Evans, <http://www.cs.virginia.edu/evans>, [evans@cs.virginia.edu](mailto:evans@cs.virginia.edu). Office: Olsson 236A.

**Office Hours.** I will have office hours Monday afternoons, 1:15-3:00pm and Thursdays right after class (3:15-4:30pm) in Olsson 236A. I am available at other times by email.

**Problem Sets.** There will be seven problem sets, each containing about 5-10 problems of varying degrees of difficulty. The problem sets are intended primarily for learning, and consistent effort on the problem sets is the most important requirement for succeeding in the class. In most cases, students are encouraged to collaborate in small groups on the problem sets. Each assignment will state a clear collaboration policy, which everyone is expected to follow honorably. The expected topics and due dates for the problem sets is below:

- PS1: Proofs, Finite Automata (Chapter 0, Section 1.1) — due Tuesday, Feb 2
- PS2: Regular languages (Chapter 1) — due Tuesday, Feb 16
- PS3: Context-free languages (Chapter 2) — due Tuesday, Feb 23
- PS4: Turing machines (Chapters 3-4) — due Tuesday, March 23
- PS5: Undecidability (Chapters 4-5) — due Tuesday, March 30
- PS6: Complexity (Chapter 7) — due Tuesday, April 20
- PS7: Hot topics — due Tuesday, May 4

**Exams.** There will be two exams and a final.

- Exam 1 (covers material from Problem Sets 1-3 and Classes 1-9): in-class (one page of notes allowed), March 2
- Exam 2 (covers material from Problem Sets 1-5 and Classes 1-18): take-home (out Thursday, April 8, due Tuesday, April 13)
- Final Exam (covers entire course): Thursday, May 13, 9am-noon (scheduled by registrar)

**Honor.** We assume all students are honorable, and will behave according to the letter and spirit of the honor code: no lying, cheating, or stealing. We expect all students to take responsibility for the honor system and the privileges that it provides. We also expect students to do what they can to help their fellow classmates learn, and to help the course staff improve the class.

For many assignments in this course, you will be encouraged to work with other students, but there may be rules about what other resources you can use. In particular, problem sets may include questions from previous versions of this course and it would be detrimental to your experience in this course, as well as unfair to the other students, to use found answers rather than solving the problems yourself. A high expectation of trust and honor allows us to reuse well known problems, which is to the benefit of everyone involved in the course.

The assignments will state the collaboration and resource use policy clearly and you are expected to follow it carefully. If you are unsure if something is permitted, consult with the course instructor before doing it. For exams, students are required to work alone and follow the stated rules exactly.

**Grading.** Students will be evaluated based on their understanding of and ability to apply key concepts from the course as demonstrated through the problem sets, exams, contributions in and out class, and extra credit efforts. If necessary, there may also be occasional quizzes. There is no curve for the course or required grade distribution.

The approximate range of possible weightings is:

	Minimum	Default	Maximum
Problem Sets	25%	40%	60%
Exam 1	10%	15%	20%
Exam 2	15%	15%	30%
Final Exam	20%	30%	50%

Grades will be calculated using the maximum value achieved using several different weightings, including the default weighting. There will also be many opportunities to earn extra credit by solving challenge problems, producing especially insightful answers, or making exceptional contributions to the class.