CMSI 385

Introduction to the Theory of Computation

http://dondi.lmu.build/fall2018/cmsi385

Fall 2018—Doolan 222
TR 2:40–3:55pm, 3 semester hours
Office Hours: TR 11:15am–12:15pm, WR 4–5:30pm, or by appointment

John David N. Dionisio, PhD
dondi@lmu.edu
Doolan 106; (310) 338-5782

Objectives and Outcomes

This course introduces you to the very foundation of computer science: the theory and reasoning beneath the pervasive activity of *computation*. At this level, we are concerned with computation in its purest form, independent of technology, application, or media. Long after you finish this course, my hope is that you will be able to:

1. **Reason about classical automata, especially finite automata, pushdown automata, and Turing machines.**
2. **Work with formal grammars and languages, especially regular sets, context-free languages, recursive, and recursively enumerable languages.**
3. **Understand and describe formal notions of computability and decidability.**

In addition to the course-specific content, you are also expected to:

4. **Follow disciplinary best practices throughout the course.**

Prerequisites/Prior Background

You must have previously completed CMSI 281 Data Structures and Algorithms I as well as MATH 248 Introduction to Methods of Proof in order to take this course.

Materials and Texts

Unlike other aspects of computer science, the theory of computation changes very little over time (barring seminal new ideas or long-sought proofs). Thus, most textbooks on the subject will likely work well. But just to name something specific…

- John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*, 3rd Edition, Addison-Wesley, 2007.
- Assorted handouts and articles may also be distributed throughout the semester.

Course Work and Grading

Course work will consist of multiple in-class and take-home exercises, each with a point value that will be totaled up at the end of the semester. Your final grade will be based on the percentage of the points you get against the total possible.

Percentages ≥ 90% get an A– or better; ≥ 80% get a B– or better; ≥ 70% get a C– or better. I may nudge grades upward based on qualitative considerations such as degree of difficulty, effort, class participation, time constraints, and overall attitude throughout the course.

Workload Expectations

In line with LMU's *Credit Hour Policy*, the workload expectation for this course is that for every one (1) hour of classroom instruction (50 scheduled minutes), you will complete at least two (2) hours of out-of-class work each week. This is a 3-unit course with 3 hours of instruction per week, so you are expected to complete $3 \times 2 = 6$ hours of weekly work outside of class.

Attendance

Attendance at all sessions is expected, but not absolutely required. If you must miss class, it is your responsibility to notify me about this and keep up with the course. The last day to add or drop a class without a grade of W is August 31. The withdrawal or credit/no-credit deadline is November 2.

Academic Honesty

Academic dishonesty will be treated as an extremely serious matter, with serious consequences that can range from receiving no credit to expulsion. It is never permissible to turn in work that has been copied from another student or copied from a source (including the Internet) without properly acknowledging the source. It is your responsibility to make sure that your work meets the standard of academic honesty set forth in:

http://academics.lmu.edu/honesty

Special Accommodations

Students with special needs who require reasonable modifications or special assistance in this course should promptly direct their request to the Disability Support Services (DSS) Office. Any student who currently has a documented disability (ADHD, autism spectrum, learning, physical, or psychiatric) needing academic accommodations should contact DSS (Daum 224, x84216) as early in the semester as possible. All requests and discussions will remain confidential. Please visit *http://www.lmu.edu/dss* for additional information.

Topics and Important Dates

Correlated outcomes are shown for each topic. Specifics may change as the course progresses. University dates (italicized) are less likely to change.

August/ September	Introduction to the course; background and implications of the theory of computation
August 31	*Last day to add or drop a class without a grade of W*
September 3	*Labor Day*
	Finite state automata; closure and nondeterminism; the pumping lemma; pushdown automata
October	Grammars and languages: regular, context-free, recursive, recursively-enumerable
November	Turing machines; computability and decidability
November 2	*Withdraw/credit/no-credit deadline*
November 21–23	*Thanksgiving; no class*
December	Wrap-up: a peek into complexity theory; open questions
December 11	*Registrar-scheduled final examination slot, 2–4pm*

You can view my class calendar and office hour schedule in any iCalendar-savvy client. Its subscription link can be found on the course web site (it's too long to provide in writing).

Tentative Nature of the Syllabus

If necessary, this syllabus and its contents are subject to revision; students are responsible for any changes or modifications distributed in class or posted to the course website.

Course Outcomes

1 Reason about classical automata, especially finite automata, pushdown automata, and Turing machines.

1a	*Know the characteristics and capabilities of each machine such that they can be constructed and "executed" for various computations and their substantive differences, particularly in terms of computing power, can be described concisely*	The theory of computation is build upon so-called *models* of computing—automata or machines that distill the essence of computational activities independent of physical devices or technology. Furthermore, their conceptual "purity" allows us to work with them as mathematical entities, and thus facilitates *proofs* about their characteristics and behavior. Knowing these in broad strokes constitutes fulfillment of this outcome.
1b	*Be familiar with the formal definitions of each machine, such that they can be reasoned about with the rigor of mathematical proofs*	

2 Work with formal grammars and languages, especially regular sets, context-free languages, recursive, and recursively enumerable languages.

2a	*Be able to specify the grammar for a particular language, and to recognize strings produced by that grammar*	One of the foundational observations of the theory of computation lies in how models of computation map quite directly into languages of certain categories. Fulfilling this outcome involves knowing how to define and work with languages at this level as well as understanding how they correspond to the different machines seen in the course.
2b	*Recognize the correspondences between different language categories and computational models for recognizing those languages*	

3 Understand and describe formal notions of computability and decidability.

3a	*Be able to state the formal meanings of computability and decidability, give clear examples of both types of problems, and describe the known limits or boundaries of such problems*	The course serves solely as an introduction, and necessarily stops at the broader ideas and implications that the theory of computation uncovers once a sufficient foundation has been laid down. It is hoped that students will gain an initial sense and appreciation of the overall and potential impact of theory on real-world technologies and approaches.
3b	*Have a sense for current open questions surrounding computability and decidability, including their implications for existing technologies*	

4 Follow disciplinary best practices throughout the course.

4a	*Express automata, grammars, languages, and proofs using clear visuals and notation, adhering to standards where applicable*	The theory of computation traffics almost completely in thought and the abstract. Thus, it is of utmost importance that when these ideas are given tangible form, they are expressed in a way that minimizes misunderstanding and maximizes conveyance of these ideas.
4b	*Use available resources and documentation to find required information.*	The need to look things up never goes away. Remember also that the course instructor counts as an "available resource," so this outcome includes asking questions and using office hours.
4c	*Meet all designated deadlines.*	