Course Information

Catalog Course Description: Regular languages and grammars; finite-state machines and transducers; relationships between finite-state automata and regular languages. Context-free languages and grammars; language recognition with stack machines and parsers. Properties of formal languages. Computability and undecidability. Introduction to computational complexity.

Prerequisites: CS-102, Computing and Algorithms II; CS-211, Discrete Mathematics

Course Credits: 4

Contact Hours: 4

Course Topics:

- Regular languages and expressions.
- Deterministic and nondeterministic finite-state machines.
- Equivalence of deterministic and nondeterministic finite-state automata.
- Equivalence of regular languages and regular expressions: Kleene's Theorem.
- Properties of regular languages.
- The pumping lemma for regular languages; nonregular languages.
- Context-free languages and grammars.
- Nondeterministic and deterministic push-down automata.
- Properties of context-free languages.
- Pumping theorems and noncontext-free languages.
- Mathematical models of computation.
- Turing machines and computability.
- Gödel numbers, Universal Turing machines, and Church's Thesis.
- The Halting Problem and uncomputability.
- Reductions and unsolvable problems.
- Introduction to resource-bounded complexity.
- Time and Space complexity classes.
- Deterministic polynomial time and nondeterministic polynomial time classes: **P** vs. **NP**.

Course Objectives: By the end of this course, you should be able to demonstrate the ability to do all of the tasks listed below:

- Given a regular expression, construct a nondeterministic finite-state machine acceptor.
- Given a nondeterministic finite-state machine, convert it to an equivalent deterministic finite-state machine.
- Given a language, determine if it is regular or nonregular language.
- Given a language L, construct a context-free grammar that generates L.
- Given a context-free grammar G, determine the language that G generates.
- Given a context-free grammar or a context-free language, construct a push-down automata that accepts the given language.
- Recognize languages that are not context-free.
- Describe a mathematical formalism for a computing machine.
- Describe the basic characterization of Turing-decidable and Turing-acceptable languages.
- Explain the Halting Problem and the fundamental limitations of computing.
- Describe the basic resource-bounded complexity classes: **DTIME**, **NTIME**, **DSPACE** and **NSPACE**.
- Formulate the complexity classes **P** and **NP**.
- Explain the importance of the complexity classes **P** and **NP**.

Course Schedule (Tentative):

Week 1: Course introduction; finite automata; regular expressions
Week 2: Nondetermistic automata; FA closure properties
Week 3: Non-regular languages; applications
Week 4: Exam 1
Week 5: Context-free grammas; pushdown automata
Week 6: CF closure properties; Non-CF languages
Week 7: Exam 2
Week 8: Turing machines
Week 9: Decidability
Week 10: Undecidability; NP-completeness
Week 11: Exam 3

Administrative Information

Course Time & Location: Tuesday & Friday 1:20pm-3:35pm, AB 4-309

Textbook: Goddard, *Introducing the Theory of Computation*, Jones & Bartlett, 2008, ISBN 9780763741259.

Software: JFLAP, freely available from `http://www.jflap.org`

Instructor: Jim Huggins

Email: `jhuggins@kettering.edu`

Physical Office: AB 2-300G

Digital Office: `http://bit.ly/HugginsOffice`

Student (Office) Hours: Tuesday & Friday 9:00am-10:00am and 3:30pm-4:00pm, *or by appointment*

Office Phone: 810-762-9500, x5439

Course Overview

Theoretical computer science is the study of the mathematical foundations underlying the notions of "computer" and "algorithm" central to the practice of computing. We will investigate several different types of simple computational devices and describe (and prove) exactly what types of algorithms they can and cannot execute. Ultimately, we will present models of computation which can be seen as simple types of digital computers, and thus come to an understanding of the capabilities and limits of these devices.

The material in this course is cumulative in nature; concepts learned in one week will depend on concepts learned in previous weeks. It is thus vital that you keep up with course concepts as the course proceeds; if you find yourself falling behind, *please* see the instructor as soon as possible.

Course Requirements and Grading

Attendance

Attendance at all class sessions is *strongly encouraged.* In accordance with university policy, class sessions will be recorded as supplemental materials; however, the level of quality of those recordings cannot be assured.

University policy requires instructors to report students who appear to have stopped attending class for administrative withdrawal from the course. Consistent failure to complete course assignments, attend class sessions, or respond to instructor email, without making contact with the instructor, may result in administrative withdrawal.

Exams

There will be three examinations during the course; each will be worth 25% of the final course grade.

The first two examinations are scheduled for 24 October (4th Tuesday) and 14 November (7th Tuesday), and will be administered in-class. In-class exams will be closed-book, closed notes.

The third examination will be a take-home exam. The exam wil be distributed on 8 December (10th Friday) and will be due at NOON on 15 December (11th Friday).

Exams will focus primarily on material covered during the corresponding third of the course; however, since material in this course is cumulative, exams will have a cumulative nature as well. Examinations will cover material presented in lecture and announced sections of the textbook. You are responsible for all material presented in class.

Examinations should be entirely your own work; unauthorized use of outside materials will be ***severely penalized***, up to and including course failure.

Except for emergencies, makeup exams must be arranged *in advance* of the announced examination date.

Homework

There will be weekly written homework assignments given; the cumulative homework score will be worth 25% of the final course grade. Homework will be submitted digitally.

Homework assignments will be due on the following schedule:

- Homework 1: 13 October (2nd Friday)
- Homework 2: 20 October (3rd Friday)
- Homework 3: 3 November (5th Friday)
- Homework 4: 10 November (6th Friday)
- Homework 5: 1 December (9th Friday)
- Homework 6: 8 December (10th Friday)

For written homework assignments, you will be assigned to work in teams of 3-4 students to complete the assignment. A separate handout describes group homework requirements.

Late homework assignments will be only be accepted under extraordinary circumstances. Materials other than the course textbook and course notes may not be used in completing homework assignments.

Course Grades and Grading Policies

Midterm course grades will be based on a weighted average of the first exam (75%) and the first two homework assignments (25%). With two-thirds of the course grade determined after the deadline for midterm grade submission, midterm grades will not be a good predictor of final course performance. Estimated course grades will be updated as the term progresses.

Final course grades will initially be computed using the relative weights described above. A class curve will be applied. Individual grades may be adjusted to reward improvement over the course of the semester. The official interpetations of the Kettering University Grading Scheme in the Kettering University Catalog will be used to guide the above adjustments.

Questions regarding individual scores on assignments and exams should be addressed to the instructor within one week of receiving the score.

All suspected cases of academic dishonesty will be handled in *strict* accordance with Kettering University policy. Any questions regarding appropriate behavior should be cleared with the instructor in advance.