

COURSE DESCRIPTION

Department and Course Number CMPS 440 **Course Coordinator** Mark G. Radle

Course Title Introduction to the Theory of Computation **Total Credits** 3

Current Bulletin Description:

Abstract basis of machines and programming. Automata, context free grammars and Turing machines. Equivalence and nonequivalence of classes of devices. Chomsky hierarchy. Incomputability. Computational complexity. Prereq: CMPS 341, CMPS 351 both with a grade of C or better, or CMPS 405, 406 both with a grade of B or better.

Textbooks

1. Peter Linz, *An Introduction to Formal Languages and Automata*, Jones and Bartlett, 2006.
2. H. Barnes, "A Programmer's View of Automata," *Computing Surveys*, 4, 4 (December, 1972).
3. C.A.R. Hoare and D.C.S. Allison, "Incomputability", *Computing Surveys*, 4, 3 (September, 1972).

References

1. J. Glenn Brookshear, *Theory of Computation: Formal Languages, Automata, and Complexity*, Benjamin/Cummings, 1989.
2. Daniel I. A. Cohen, *Introduction to Computer Theory*, Wiley, 1991.
3. H. R. Lewis and C. H. Papadimitriou, *Elements of the Theory of Computation*, Prentice Hall, 1995.
4. Dean Kelley, *Automata and Formal Languages: an Introduction*, Prentice-Hall, 1995.
5. Michael Sipser, *Introduction to the Theory of Computation*, PWS, 1997.

Course Goals

1. Basic knowledge of the abstract foundations of computation – formal models and essential limits.
2. General knowledge of the fundamental models of language structure and their application to design and
3. parsing of programming languages – automata, regular expressions, grammars, pushdown automata, Turing machines.
4. Understanding of mathematical techniques of representing and inferring characteristics of complex systems from simple models – inclusion, exclusion, and equivalence.
5. Introduction to hierarchies of difficulty and impossibility of computation.

Prerequisites by Topic

1. Formal proof techniques – induction, contradiction, counterexample.
2. Basic logic and set theory.
3. Graph theory – directed and undirected graphs, trees, paths and connectedness.
4. Combinatorics – counting problems, permutations and combinations
5. Programming and data structures – sets, lists, stacks, trees, strings, recursion, BNF representation of languages.

Major Topics Covered in the Course

- | | |
|--|-------------|
| 1. Regular Languages | (4-5 weeks) |
| 2. Context Free Languages | (4-5 weeks) |
| 3. Computability and Decidability | (3-4 weeks) |
| 4. Complexity of computations and algorithms | (1-2 weeks) |

Written Communications:

Homework assignments and examinations include proofs and formal definitions, graded in part on clarity and quality of mathematical exposition.

Social and Ethical Issues

Not directly addressed in course content or examination questions. Ethical standards expected in the course are stated in course materials and discussed in class.

Theoretical Content

The entire course is theoretical, contents as stated above.

Problem Analysis

The techniques of applying an abstract model (automaton, regular expression, grammar, etc.) to a given problem are discussed, practiced in many exercises, and are critiqued.