CMSC 303 (Fall 2017)
Introduction to the Theory of Computing
Syllabus

Catalog listing: CMSC 303
Course Level: Undergraduate
Prerequisites: CMSC 302 with a C or better
Instructor: Dr. Sevag Gharibian

Office: E4240
Phone: 804-828-0407
Email: sgharibian@vcu.edu
Classroom: Engineering East Hall 3229
Class website:
www.people.vcu.edu/~sgharibian/courses/CMSC303/2017_Fall/CMSC303.html

Office Hours: Tues: 3:15 pm – 4:15 pm
Thurs: 1:00 pm - 2:00 pm

1.0 – Overview (Catalog Course Description):

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 302 with a grade of C or better. Complexity classes, grammars, automata, formal languages, Turing machines, computability.

2.0 – Course Structure:

Lecture hours/week – 3

3.0 – Course Goals

Upon successful completion of this course, the student will be able to:

- Prove that a language is regular, context-free, or recursively enumerable
- Create DFAs, NFAs, regular expressions, and convert from one to the other
- Work with Turing machines, and explain the significance of the Church-Turing Thesis
- Prove that certain languages are not regular or not recursively enumerable
- Prove certain problems are undecidable via a diagonalization argument
- Define the classes P and NP, and prove NP-hardness of certain problems

4.0 – ABET Criteria Addressed:

a. An ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline.

b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.

c. An ability to analyze the local and global impact of computing on individuals, organizations, and society

d. Recognition of the need for and an ability to engage in continuing professional development

5.0 – Major Topics Covered:

- Introduction
  - Review of Discrete Mathematics (Logic, sets, functions, relations, etc.)
  - Alphabets, strings, formal languages, language classes
- Finite Automata
  - Deterministic Finite Automata
  - Nondeterministic Finite Automata
  - Regular expressions
  - The Pumping Lemma and non-regular languages
- Context-Free Languages
  - Context-Free Grammars
  - Pushdown Automata
- Turing machines and computability theory
  - Turing machines (deterministic, non-deterministic, multi-tape, enumerators)
  - Universal Turing machines and the Church-Turing Thesis
  - Detour: Countable versus uncountable sets, uncountability of the real numbers
  - Basic undecidability (halting problem)
  - More undecidability: Mapping and Turing reductions
  - Rice's Theorem (Problem 5.28 in the text)
- Introduction to complexity theory
  - P, NP, and the importance of the P vs NP question
  - Polynomial-time reductions
  - The Cook-Levin theorem (SAT is NP-complete) and its proof
  - More NP-complete problems: 3-SAT, CLIQUE, SUBSET-SUM, HAMPATH
- Independent learning via course project: Space complexity
  - PSPACE and NPSPACE
  - Savitch's theorem (PSPACE = NPSPACE)
  - TQBF is PSPACE-complete

6.0 – Textbook(s):

Michael Sipser, Introduction to the Theory of Computation, Third Edition (the Second Edition is also fine).

7.0 – Class Schedule:

- Lecture: TR 2:00 pm – 3:15 pm, Engineering East Hall, Room 3229
- Midterm Exam: Tuesday, Oct 24, 2017 in class from 2-3:15 pm.
- Final Exam: Tuesday, Dec 12, 2017 in class from 1-3:50 pm.

8.0 – Evaluation:

General Instructions:

Attendance policy:
I do not plan to associate grades with whether or not you attend class, unless a serious problem with attendance develops.

The grading scheme is below:

Grading:

Category	% weight
Problem sets	35%
Project	5%
Midterm	30%
Final exam	30%

Grading scheme:

A: >= 85%
B: >= 75% and < 85%
C: >= 65% and < 75%
D: >= 50% and < 65%
F: < 50%

Collaboration and Cheating Policy:

Collaboration:
Collaboration on homework assignments and projects is encouraged. However, do *not* simply copy one another's work. Rather, feel free to brainstorm and discuss general ideas for solutions together. Then, individually write up your own solution in your own private time.

Citations:
You must cite your collaborators at the start of your assignment or project. For example, if collaborating with Homer Simpson, write "Note: I completed this assignment in collaboration with Homer Simpson" at the start of your assignment.

Cheating:
You are NOT permitted to look up or copy solutions from any source, including the internet or collaborators. As discussed above, collaboration with fellow CMSC 303 students is encouraged, but each individual must independently write up their solutions from scratch without looking at his/her collaborators' final write-up.

Any cheating offense is taken seriously, and will be directly reported to the VCU Honor Council (https://students.vcu.edu/studentconduct/).
