SYLLABUS

COURSE NUMBER: CSC 553

COURSE TITLE: Theory of Computation

CREDITS: 3:3:0

PREREQUISITES/COREQUISITES: Pr. grade of at least C in CSC 350 or consent of instructor.

FOR WHOM PLANNED: This course is required for Computer Science graduate and upper-level undergraduate students. It is also intended as an elective for Mathematics graduate and upper-level undergraduate students.

INSTRUCTOR INFORMATION:
Dr. Francine Blanchet-Sadri
Office number: 157 Petty Building
Office hours: MWF 7:20–8:00am or by appointment
Phone number: 336–256–1125
E-mail address: blanchet@uncg.edu

CATALOG DESCRIPTION: Finite state automata and regular expressions, context-free grammars, push-down automata and their use in parsing, overview of language translation systems, models for programming language semantics, computability and undecidability.

STUDENT LEARNING OUTCOMES: On completion of the course, the student will be able to understand the basic theoretical models of computability.

Upon successful completion of this course, a student should be able to:

1. Understand the basic theoretical models of computability: deterministic and nondeterministic finite automata, pushdown automata, and variants of Turing machines.

2. Design finite automata corresponding to given regular sets, and describe the regular set associated to a given finite automaton. Do the

same with pushdown automata and context-free languages, and with Turing machines and recursively enumerable sets.

3. Comprehend and apply a number of algorithms such as: the subset construction to transform a nondeterministic finite automaton into a deterministic one; the DFA state minimization to minimize the number of states in a deterministic finite automaton; and the conversion algorithms from regular expressions to finite automata and vice versa.

4. Understand limitations of finite automata (respectively, pushdown automata) and prove that some sets are not regular (respectively, context-free) by using the pumping lemma for regular languages (respectively, context-free languages).

5. Simulate CFGs by NPDAs and vice versa, that is, convert a given context-free grammar to an equivalent nondeterministic pushdown automaton, and convert a nondeterministic pushdown automaton to an equivalent context-free grammar.

6. Apply algorithms to transform context-free grammars into normal forms such as the Chomsky and Greibach normal forms.

7. Prove that some problems are decidable or undecidable using techniques such as diagonalization and reduction.

TEACHING STRATEGIES: Group lecture is the primary mode of instruction.

EVALUATION METHODS AND GUIDELINES FOR ASSIGNMENTS:

5 homework assignments, worth 7 points each

3 tests, worth 20 points each

class participation, worth 5 points

Graduate students: In addition to the above, a graduate student taking CSC 553 is required to write a term paper on a topic chosen by me and the student. Suitable topics can be chosen from journals held by the library (for instance *Theoretical Computer Science*). The weight of the paper is 20 points, and the total (120 points) will be prorated to 100 points.

Assignment 1 *Due September 11*

Assignment 2 *Due September 27*

Test 1 *October 2*

Assignment 3 *Due October 13*

Assignment 4 *Due October 25*

Test 2 *October 30*

Assignment 5 *Due November 20*

Term paper for graduate students *Due November 20*

Test 3 *November 29*

REQUIRED TEXT/READINGS/REFERENCES:

Dexter C. Kozen, *Automata and Computability*, Springer-Verlag, 1997.

TOPICAL OUTLINE:

Introduction

Lecture 1 Course Roadmap and Historical Perspective

Lecture 2 Strings and Sets

Finite Automata and Regular Sets

Lecture 3 Finite Automata and Regular Sets

Lecture 4 More on Regular Sets

Lecture 27 The Cocke-Kasami-Younger Algorithm

Turing Machines and Effective Computability

Lecture 28 Turing Machines and Effective Computability

Lecture 29 More on Turing Machines

Lecture 30 Equivalent Models

Lecture 31 Universal Machines and Diagonalization

Lecture 32 Decidable and Undecidable Problems

Lecture 33 Reduction

Lecture 34 Rice's Theorem

Lecture 35 Undecidable Problems about CFLs

Lecture 36 Other Formalisms

ACADEMIC HONOR CODE: Each student is required to sign the Academic Integrity Policy on all major work submitted for the course. Refer to UNCG *Undergraduate Bulletin*.

ATTENDANCE POLICY: Students are expected to attend all the lectures. If a student misses any of them, it is his/her responsibility to find out what went on during the lecture and to collect any material that was handed out. Any student missing more than three lectures will fail the course (see ADDITIONAL REQUIREMENTS below).

ADDITIONAL REQUIREMENTS: Students will be allowed to drop the course after the drop deadline if there is supporting evidence of problems interfering with adequate course performance.

No make-up exam is given unless extenuating circumstances are proved.

I may withdraw a student from the course for behavior that is deemed by me to be disruptive to the class. The grade assigned will be "W" if the behavior occurs before the deadline for dropping the course without academic penalty, and I have the option of giving a "W" or a "WF" if the behavior occurs after the deadline. (Refer to POLICIES FOR STUDENTS)

Note that late arrivals are considered disruptive to the class. Any two late arrivals will be counted towards the missing of a lecture (see ATTENDANCE POLICY above).

No late assignment will be accepted. Solutions will be distributed the day an assignment is due.