School of Computing
UNIVERSITY OF GEORGIA

Course Information Sheet
CSCI 2670
Introduction to Theory of Computing

Brief Course Description
(50-words or less)

Fundamental Topics include finite automata, regular expressions and languages, context-free grammars and languages, push-down automata, pumping lemmas for regular languages and for context-free grammars, the Chomsky hierarchy of language classes, Turing machines and computability, undecidability of the halting problem, reducibility among decision problems and languages, time complexity, and NP-completeness and tractability.

Extended Course Description / Comments

A first course on the theory of computing. It is open to any students interested in learning the underlying mathematical models of computation.

Pre-Requisites and/or Co-Requisites

CSCI/MATH 2610: Discrete Mathematics or CSCI 2611 Discrete Mathematics for Engineers

Required, Elective or Selected Elective

Required

Approved Textbooks
(if more than one listed, the textbook used is up to the instructor's discretion)

Michael Sipser
Introduction to the Theory of Computation 3rd Edition
ISBN-13: 978-1133187813

Specific Learning Outcomes (Performance Indicators)

1. Define regular languages using various methods and be able to prove if a language is or is not regular.
2. Define context-free languages using various methods and be able to prove if a language is or is not context-free.
3. Define Turing recognizable languages using various types of Turing machines and be able to prove the different types of machines are equivalent with regard to the languages they accept.
4. Define decidability and be able to prove if a language is decidable or not.
5. Distinguish between P and NP and prove a language is in P or in NP.

Relationship Between Student Outcomes and Learning Outcomes

	Student Outcomes						
Learning Outcomes		a	b	c	d	e	f
	1	●	●				●
	2	●	●				●
	3	●	●				●
	4	●					●
	5	●					

Major Topics Covered for each Learning Outcome

1. Regular Languages (Knowledge level: Assessment)
    a. Given an NFA M, create a DFA or regular expression that accepts L(M).
    b. Given a regular language L, create an NFA that accepts L.
    c. Use the pumping lemma for regular languages to prove a language is not regular.
2. Context-free Languages (Knowledge level: Assessment)
    a. Given a description of a context free language L, develop a context free grammar G such that L(G) = L.
    b. Convert a context free grammar to an equivalent push-down automata and vice-versa.
    c. Convert a CFG into Chomsky normal form (CNF).
    d. Given a context-free grammar G in CNF and a string w, use the CYK algorithm to determine if G generates w
    e. Use the pumping lemma for context-free languages to prove a language is not context-free.
    f. Identify if a given language is regular, context-free, or neither.
3. Turing-recognizable Languages (Knowledge level: Assessment)
    a. Given a language L, create a Turing machine that accepts L.
    b. Convert between the different variations of Turing machines (e.g., multi-tape to single-tape).
    c. Create a Turing machine that computes a function.
4. Decidability (Knowledge level: Assessment)
    a. Define decidability and determine if a language is decidable.
    b. Prove that the Halting problem is undecidable.
    c. Reduce one problem to another.
    d. Use reductions to prove a problem is undecidable.
5. Computational Complexity (Knowledge level: varies by topic)
    a. Define P, NP, and NP-complete (Familiarity)
    b. Show a problem is in P. (Usage)
    c. Write pseudo-code describing a non-deterministic Turing machine's steps to solve a problem. (Usage)
    d. Describe the Cook-Levin theorem. (Familiarity)
    e. Describe a verifier for an NP-complete language. (Usage)

Knowledge Levels

The following is the ACM's categorization of different levels of mastery: Assessment, Usage, and Familiarity. Note that Assessment encompasses both Usage and Familiarity, and Usage encompasses Familiarity.

Familiarity: The student understands what a concept is or what it means. This level of mastery concerns a basic awareness of a concept as

opposed to expecting real facility with its application. It provides an answer to the question “What do you know about this?”

Usage: The student is able to use or apply a concept in a concrete way. Using a concept may include, for example, appropriately using a specific concept in a program, using a particular proof technique, or performing a particular analysis. It provides an answer to the question “What do you know how to do?”

Assessment: The student is able to consider a concept from multiple viewpoints and/or justify the selection of a particular approach to solve a problem. This level of mastery implies more than using a concept; it involves the ability to select an appropriate approach from understood alternatives. It provides an answer to the question “Why would you do that?”

Course Master	Dr. Michael Cotterell
Modified	6/5/2019 by Dr. Funk, Dr. Cotterell and Dr. Hollingsworth
Approved	Yes