UNIVERSITY OF NEBRASKA AT OMAHA
COURSE SYLLABUS/DESCRIPTION

Department and Course Number	CSCI 3660
Course Title	Theory of Computation
Course Coordinator	Hai-Feng Guo
Total Credits	3
Date of Last Revision	April 28, 2014

1.0 Course Description

1.1 Overview of content and purpose of the course

The course is intended for an introductory course on formal languages, automata, and computability. The topics covered in the course include finite automata, non-determinism, regular expressions and languages, context-free grammars, pushdown automata, context-free languages, Backus normal form, ambiguity, Turing machines, decidability, and the Chomsky Hierarchy.
The study of the theory of computation has several purposes, most importantly (1) to familiarize students with the foundations and principles of computer science, (2) to teach material that is useful in subsequent courses, and (3) to strengthen students' ability to carry out formal and rigorous mathematical arguments.

1.2 For whom course is intended

The course is intended for computer science students with junior standing.

1.3 Prerequisites

CSCI 2030, CSCI 3320

1.4 Unusual circumstances of the course

None

2.0 Objectives

2.1 Study Finite Automata
2.2 Study Regular Languages
2.3 Study Regular Grammars
2.4 Study Context-Free Languages
2.5 Study Pushdown Automata
2.6 Study Turing Machines
2.7 Study Finite Automata
2.8 Study Decidability
2.9 Study the Chomsky Hierarchy

3.0 Content and Organization

	Contact hours
3.1 Introduction	2.0
3.1.1 Mathematical Preliminaries and Notation	
3.1.2 Three Basic Concepts	
3.1.3 Some Applications	
3.2 Finite Automata	6.0

4.0 Teaching Methodology

4.1 Methods to be used

The primary teaching methods will be lecture, in-class demonstrations, and in-class exercises.

4.2 Student role in the course

The student will attend lectures and demonstration, participate in discussion on assigned readings, complete assigned homework, and complete required examinations.

4.3 Contact hours

Three hours per week

5.0 Evaluation

5.1 Type of student products

Students are asked to complete assignments intended to enhance theoretical analysis and proofs. This is in addition to exams given in the course.

5.2 Basis for determining the final grade

Component	Grading
Exams	70%
Assignments	25%
Participation	5%

5.3 Grading scale

Points	*Grade*
97-100%	A+
93-96%	A
90-92%	A–
87-89%	B+
83-86%	B
80-82%	B–
77-79%	C+
73-76%	C
70-72%	C–
67-69%	D+
63-66%	D
60-62%	D–
0-59%	F

6.0 Resource Material

6.1 Textbook(s) or other required readings used in the course, or equivalent examples

6.1.1 Peter Linz: An Introduction to Formal Languages and Automata, fourth edition, Jones and Bartlett publisher, 2006.

6.2 Other suggested reading material or equivalents

6.3 Other sources for gathering information or equivalent

6.4 Current bibliography or equivalent

6.4.1 *Languages and Machines: An Introduction to the Theory of Computer Science*, by Thomas A. Sudkamp, Addison Wesley, 1996.
6.4.2 *Theory of Computing: A Gentle Introduction*, by Efim Kimber & Carl A. Smith, Prentice Hall, 2000.
6.4.3 *Theory of Computation: Formal Languages, Automata, and Complexity*, by J. Glenn Brookshear, Benjamin Cummings, 1989.
6.4.4 *Elements of the Theory of Computation*, by Harry R. Lewis & Christos Papadimitriou, Prentice Hall, 1997.
6.4.5 *Introduction to the Theory of Computation*, by Michael Sipser, Brooks/Cole Publishing Company, 1996.
6.4.6 *Introduction to Computability*, by Frederick C. Hennie, Addison Wesley, 1977.
6.4.7 *Computability and Complexity: From a Programming Perspective*, by Neil D. Jones, MIT Press, 1997.

7.0 Computing Accreditation Commission Category Content (class time in hours):

CSAB Category	*Core*	*Advanced*
Data Structures		
Computer Organization and Architecture		
Algorithm and Software Design		
Concepts of Programming Languages		

8.0 Oral and Written Communications

Every student is required to submit at least __0___ written reports (not including exams, tests, quizzes, or commented programs) to typically _____ pages and to make ___0__ oral presentations of typically _____ minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

9.0 Social and Ethical Issues

No coverage.

10.0 Theoretical content

The course is very theoretical. Computational logic and formal language models are well founded in the theoretical aspects of computer science.

11.0 Problem analysis

The course is mathematical in nature and, as a result, requires considerable understanding of the methods used to solve mathematical problems in the course. This involves problem detailed problem reading to understand what is given and what is required. This is followed by analysis applied to special cases for better understanding of the problem requirements.

12.0 Solution design

The solution design includes verification and proof of correctness. In particular, proof by contradiction, and inductive proofs are used extensively.

CHANGE HISTORY

Date	***Change***	***By whom***	***Comments***
11/05/2002	Initial ABET version	Ali	
06/20/2003	Cleanup	Wileman	
03/01/2004	Updated course objective, content, textbook	Hassan Farhat, Hai-Feng Guo	
10/13/2008	Cleanup	Hai-Feng Guo	
10/13/2008	Filled mapping table (Outcomes vs. Objectives)	Hai-Feng Guo	
4/28/2014	Updated Textbook and contents	Hai-Feng Guo	