University of Texas at El Paso
Course Syllabus

COURESE DESCRIPTION

Dept., Number	CS3350	Course Title	Automata, Computability, and Formal Languages
Approval Date		Course Coordinator	Vladik Kreinovich

CATALOG DESCRIPTION

Automata, Computability and Formal Languages (3-0) Theoretical computing models and the formal languages they characterize: finite state machines, regular expressions, pushdown automata, context-free grammars, Turing machines and computability. Capabilities and limitations of each model, and applications including lexical analysis and parsing.

TEXT BOOK

Sipser, Michael. Introduction to the Theory of Computation.

COURSE OUTCOMES

Level 1: Knowledge and Comprehension:

Level 1 outcomes are those in which the student has been exposed to the terms and concepts at a basic level and can supply basic definitions. Upon successful completion of this course, students will be able to:

a. Describe implications of Church-Turing thesis.
b. Describe problems for which an algorithm exists, and problems for which there are no algorithms (non-recursive, non-recursively enumerable languages) and understand the implications of such results.
c. Describe and explain the diagonalization process as used in proofs about computability.
d. Describe the difference between feasible and non-feasible algorithms, understand the limitations of the current formalization of feasibility as polynomial-time.
e. Describe the main ideas behind the concepts of NP and NP-hardness, know examples of NP-hard problems.

Level 2: Application and Analysis:

Level 2 outcomes are those in which the student can apply the material in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details. Upon successful completion of this course, students will be able to:

a. Convert a non-deterministic finite automaton into an equivalent deterministic finite automaton.
b. Convert a non-deterministic finite automaton into an equivalent regular expression.
c. Convert a regular expression into an equivalent finite automaton.
d. Construct a regular expression for a regular language.
e. Convert a context-free grammar into an equivalent pushdown automaton.
f. Construct a context-free grammar for a given context-free language.
g. Design an algorithm for a machine model to simulate another model.

h. Build simple Turing machines.
i. Prove formally properties of languages or computational models.

h. Apply a parsing algorithm.
i. Build a parse tree or a derivation from a context-free grammar.
j. Use the closure properties in arguments about languages

Level 3: Synthesis and Evaluation

Level 3 outcomes are those in which the student can apply the material in new situations. This is the highest level of mastery. Upon successful completion of this course, students will be able to:

a. Compare regular, context-free, recursive, and recursively enumerable languages.
b. Compare finite automata, pushdown automata, and Turing machines.

ABET STUDENT OUTCOMES MAPPING

Course outcomes	Student outcome
2a-c, 2f	1
1b, 3a, 3b	2 (ABET 1)
None	3 (ABET 2)
None	4 (ABET 5)

None	5 (ABET 4)
None	6 (ABET 3)
None	7
None	8
None	9
1b, 2a-c, 2i-j	10 (ABET 6)

PREREQUISITES BY TOPIC

C or higher in CS 2302 Data Structures and (MATH 2300 Discrete Mathematics or CS 2202 Discrete Structures II), OR B or better in CS 2401 Elementary Data Structures and Algorithms and MATH 2300