[CS 3350](#) Automata, Computability, and Formal Languages
Fall 2015 Syllabus

Class Meetings: Monday and Wednesday 12-1:20 pm, Room CRBL 305

Final Exam: Tuesday December 11, 1-3:45 pm

Instructor: [Vladik Kreinovich](#), email vladik@utep.edu, office CCSB 3.0404,
office phone (915) 747-6951
office hours: Monday and Wednesday, 1:30-3 pm, Monday 4:30-5:30 pm, or by appointment.

Teaching Assistant (TA): TBA

Course Objectives: 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. Prerequisite: CS 2302 with a grade of C or better.

Major Topics Covered in the Course

- Regular languages, finite automata, non deterministic FA
- Context-free languages, pushdown automata
- Parsing, normal forms, ambiguity
- Pumping lemmas and closure properties
- Turing machines and other equivalent models
- Decidable languages, non-decidable languages, recognizable languages, Chomsky hierarchy

Learning 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. The material has been presented only at a superficial level.
Upon successful completion of this course, students will:

1a. Be familiar with the implications of Church-Turing thesis.

1b. Understand that there are 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.

1c. Understand and explain the diagonalization process as used in proofs about computability.

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:

2a. Convert a non-deterministic FA (respectively transition graph) into an equivalent deterministic FA, convert a transition graph or NFA into an equivalent regular expression, and convert a regular expression into an equivalent FA.

2b. Construct a regular expression (respectively a context-free grammar) for a regular language (respectively context-free language).

2c. Convert a context-free grammar into an equivalent pushdown automaton.

2d. Construct a context-free grammar for a given context-free language.

2e. Design an algorithm for a machine model to simulate another model.

2f. Build simple Turing machines.

2g. Prove formally properties of languages or computational models.

2h. Apply a parsing algorithm.

2i. Build a parse tree or a derivation from a context-free grammar.

2j. 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:

3a. Compare regular, context-free, recursive, and recursively enumerable languages.

3b. Compare FA, PDA, and Turing machines.

Textbook: Reading and laboratory assignments will be drawn from *Introduction to the Theory of Computation*, by Michael Sipser (both 2nd and 3rd editions are OK). If necessary, this book is available at the bookstore and through major online book retailers, and you are expected to acquire a copy for your use in this course. Photocopied textbooks are illegal and their use will not be tolerated.

Assignments: Reading and homework assignments will be handed out or announced in class and in labs. If you miss a class, it is your responsibility to find out what you missed. You should expect to spend at least 10 hours/week outside of class on reading and homework.

Grading: There will be two tests and one final exam. Each topic means home assignments (mainly on the sheets of paper, but some on the real computer). Maximum number of points:

- first test: 20
- second test: 20
- home assignments and quizzes: 20
- final exam: 40

The nominal percentage-score-to-letter-grade conversion is as follows:

- 90% or higher is an A
- 80-89% is a B
- 70-79% is a C
- 60-69% is a D
- below 60% is an F

We reserve the right to adjust these criteria downward, e.g., so that 88% or higher represents an A, based on overall class performance. The criteria will not be adjusted upward, however.

Homework Assignments: Homework and lab assignments are designed to allow you to practice using the concepts presented in lecture and in your reading. Homework and lab assignments may include written problems, tutorial exercises, and programming problems. Assignments usually will be due at the start of the next class. Late homework will be accepted only in unusual circumstances, by prior arrangement if at all possible.

Homework must be done individually. While you may discuss the problem in general terms with other people, your answers and your code should be written and tested by you alone. If you need help, consult a TA or a professors

Quizzes: The purpose of a quiz is to ensure that you have read the weekly reading assignment and to verify that you have mastered the major concepts of recent lectures. Quizzes typically will be about 5-10 minutes in length and will cover the material assigned to be read for the upcoming lecture plus selected concepts from previous lectures. There will be no make-up on missed quizzes.

Exams: The purpose of the exams is to allow you to demonstrate mastery of course concepts. Make-up exams will be given only in extremely unusual circumstances. If you must miss an exam, please meet with an instructor, BEFORE the exam if at all possible.

Standards of Conduct: You are expected to conduct yourself in a professional and courteous manner, as prescribed by the UTEP Standards of Conduct.

Graded work, e.g., homework and tests, is to be completed independently and should be unmistakably your own work (or, in the case of group work, your team's work), although you may discuss your project with other students in a general way. You may not represent as your own work material that is transcribed or copied from another person, book, or any other source, e.g., a web page.

Academic dishonesty includes but is not limited to cheating, plagiarism and collusion.

- Cheating may involve copying from or providing information to another student, possessing unauthorized materials during a test, or falsifying data (for example program outputs) in laboratory reports.
- Plagiarism occurs when someone represents the work or ideas of another person as his/her own.
- Collusion involves collaborating with another person to commit an academically dishonest act.

Professors are required to -- and will -- report academic dishonesty and any other violation of the Standards of Conduct to the Dean of Students.

Disabilities: If you feel you may have a disability that requires accommodation, contact the The Center for Accommodations and Support Services (CASS) at 747-5148, go to Room 106 E. Union, or e-mail to cass@utep.edu.