**MET CS 662 – Madani Naidjate - Spring 2020**

**Overview**

The goal of this course is to provide the student with a solid knowledge of the fundamental concepts and methods of the theory of computation as well as to outline modern research directions. Three different approaches for capturing the idea of computing in a formal mathematical way will be discussed finite state machines, grammars and recursive functions. At the end of the course students are expected to be able to interpret relate and apply the basic concepts of the theory of computation to problems from different areas of computer science.

**Course Objectives**

* Apply the algebra of theoretical machines to computing problems
* Possess knowledge of non-determinism in computational models.
* Remove nondeterminism from simple models when it is possible.
* Understand the basic problems of computability, decidability and the halting problem as well as the relationships among them.
* Apply the concept of a Turing machine to the decidability problem where possible.
* Understand the Church-Turing Thesis and its significance in computer science.
* Have a working knowledge of the Chomsky hierarchy of languages.
* Relate theoretical computer science topics to programming languages and other recursively enumerable sets.

**Prerequisites**

* Discrete Mathematics course (MET CS248 or equivalent.)
* Introductory computer programming class

**Textbook**

* P. Linz "An Introduction to Formal Languages and Automata" ***any*** edition, D.C. Heath and Co. (2001 - 2016)

**Grading**

* Midterm 35% (Date will be announced)
* Final Exam 35% (Date will be announced)
* Assignments 30%

**References**

* 1- H. R. Lewis, C. H. Papadimitriou "Elements of the Theory of Computation" Prentice Hall, 1981.
* *2-* J.E. Hopcroft, "Introduction to Automata Theory, Languages and Computations" Addison Wesley, 1979.

All assignments and announcements pertinent to the course can be found on Blackboard.

**Course outline**

|  |  |  |
| --- | --- | --- |
| **Lecture** | **Topic** | **Description** |
| 1 | 1.1 | Mathematical preliminaries |
| 2 | 1.2, 2.1, 2.2, 2.3 | Basic concepts of languages, grammars and automata - DFAs - NDFAs and equivalence |
| 3 | 3.1, 3.2, 3.3 | Regulars expressions, regular grammars |
| 4 | 4.1, 4.2, 4.3 | Basic properties of regular languages |
| 5 | 5.1, 5.2, 5.3 | Context-Free (CTF) languages. Parsing and Ambiguity. Programming languages |
| 6 | 6.1, 6.2, 7.1 | Simplification of CTF grammars. Normal forms. |
| 7 | 7.2, 7.3, 7.4 | Pushdown automata (PA). Nondeterministic & deterministic PA. PA and CTF languages. |
|  | Midterm Examination |
| 8 | 8.1, 8.2 | Discuss exam. Properties of CTF languages. Pumping lemmas. Properties. |
| 9 | 9.1, 9.2, 9.3 | Turing Machines (TM). Standard TMs, Turing Thesis. |
| 10 | 10.1, 10.2, 10.4, 10.5 | Models of TMs (option stay, Semi-infinite tape, off-line, Multitape, Multidimensional, Nondeterministic, universal). Linear bounded Automata. |
| 11 | 11.1, 11.2, 11.3, 11.4 | Hierarchy of formal languages and Automata. Recursive and Recursively Enumerable Languages. |
| 12 | 12.1, 12.2, 13.1 | Limits of Algorithmic Computation. Problems that cannot be solved by TMs. Undecidable Problems for Recursively Enumerable Languages. Other Models of computation |
|  | Final Examination |