Course Title: Computation Theory

Course Description:
This course sets out to the fundamentals of formal language theory and the theory of computation. Key topics include models of computation, the Church-Turing thesis, Turing machines, decidability and undecidability, computational complexity, NP-completeness, and diagonalization.

Prerequisite:
- **CS 3000:** Discrete Structures

Instructor Information:

- **Instructor:** Dr. Majid Mirzanezhad
- **Office:** 376 Stocker Center
- **Email:** miirza@ohio.edu
- **Office Hours:** To Be Determined

Meeting Times and Location:
- **Lectures:** Monday, Wednesday, Friday | 12:55 PM – 1:50 PM
- **Location:** Irvine Hall, Room 194

Textbook:
- *Introduction to the Theory of Computation* by Michael Sipser, MIT, 3rd Edition

---

Course Objectives:
By the end of this course, students will:
1. Apply foundational knowledge of automata, grammars, and formal languages.
2. Understand the Church-Turing thesis and basic Turing machine models.
3. Apply techniques to prove undecidability of certain languages.
4. Utilize the Recursion Theorem and Rice's Theorem in problem-solving.
5. Comprehend computable functions and address precision in computations.
6. Grasp concepts of computational complexity, including P and NP classes.
7. Prove NP-completeness of problems.

Course Topics:
- **Introduction and Mathematical Preliminaries:**
  Sets, relations, functions, strings, languages, and proof techniques.
- **Automata and Formal Languages:**
  Finite automata, regular languages, context-free grammars, pushdown automata, etc.
- **Turing Machines and the Church-Turing Thesis:**
  Turing machine models, variants, the Church-Turing thesis, and universal Turing machines.
- **Decidability and Undecidability:**
  Decidable languages, undecidable problems, reducibility, and the Halting Problem.
- **Recursion Theorem and Rice's Theorem:**
  Understanding and applications of these theorems.
- **Computable Functions and Numbers:**
  Computable functions, computable real numbers (e.g., $\sqrt{2}$), and precision issues.
- **Complexity Theory:**
  Time complexity classes (P, NP), space complexity, and nondeterministic computation.
- **NP-Completeness:**
  Polynomial-time reducibility, NP-completeness, and methods for proving NP-completeness.
- **Diagonalization and Hierarchy Theorems:**
  Diagonalization techniques and time/space hierarchy theorems.