THE CITY COLLEGE OF NEW YORK

Csc 486 – Introduction to Computational Complexity

Tuesday and **Thursday** from **11am** to 1**2:15pm** in Room **NA-6311**

Instructor: Prof. **Rosario Gennaro** (rosario@ccny.cuny.edu)
Office Hours: Tuesday 3:30-5pm.
Class URL: http://www-cs.ccny.cuny.edu/~rosario/csc486/

Topics

This class will be an introduction to ***hard computational problems***, those for which we do not know how to compute a solution with feasible amount of ***resources***.

In order to do so the first thing we will have to do is to define formal computational models (Turing Machines, RAM, circuits, etc.) and corresponding notions of cost (time, space, interaction, etc.) This will bring us to the definition of the classes P (the problems which are "easy" to solve) and NP (the problems for which the solution can be easily verified, but we do not know if it can be found efficiently). We will talk about the notion of NP-completeness (identifying crucial problems that if solved efficiently would guarantee that all of NP is in P) and discuss the question of P vs NP. We will also cover complexity classes beyond NP, such as PSPACE

In the second part of the class we will discuss "coping mechanisms" for hard problems. We will explore the use of randomized algorithms, and of algorithms that find an approximate solution. We will also talk about the ability to rely on "help" from more powerful machines (Interactive Proofs).

We will also discuss how complexity theory has informed the development of modern cryptography. The idea is that if a problem is computationally hard to solve, then we can use it as the basis of a way to encrypt information. An adversary, in order to "break" the code would have to solve such a hard problem. If time permits we will also cover the notion of zero-knowledge proofs, and its applications.

Grading

Homework will be assigned every Tuesday and due the following Tuesday. You are allowed to skip one homework during the semester. We will have an in-class midterm and an in-class final exam on the last day of class. Homework assignments, Midterm and Final exam will each count towards 30% of the grade. The remaining 10% will be given to class participation and interaction.

Syllabus (Tentative)

August 29	Introduction. Models of Computation.
Sept 3	Turing Machines.
Sept 10-12	Computability Theory: Decidable and Undecidable Problems.
Sept 17-19	Reducibility. More Undecidable Problems.
Sept 24-26	Time Complexity: The classes P and NP.
Oct 1-3	Cook's Theorem and NP-Complete Problems.
Oct 8-10	More NP-complete Problems.
Oct 17	Midterm Exam (in class)
Oct 22-24	Space Complexity: PSPACE
Oct 29	Space Complexity: L and NL
Oct 31	Approximation Algorithms
Nov 5-7	Randomized Algorithms: BPP, Primality Testing.
Nov 12-14	Interactive Proofs: The IP class and the proof that IP=PSPACE
Nov 19-21	Introduction to Cryptography.
Nov 26	Factoring and RSA Encryption
Dec 3	Computational Hardness and Pseudo-Randomness
Dec 5-10	Zero-Knowledge Proofs
Dec 12	Final Exam (in class)