Bocconi University

Ph.D. in Statistics and Computer Science 2022-2023

TITLE: Computer Science II

LECTURER: Alon Rosen, Bocconi University

PREREQUISITE: Algorithms, data structures, probability, and discrete math.

TEACHING MATERIAL: The course will closely (but not completely) follow Michael Sipser's *Introduction to the Theory of Computation*. Other, sources that you may find useful are:

- Computational Complexity: A Conceptual Perspective. O. Goldreich.
- Computational Complexity: A Modern Approach. S. Arora and B. Barak.

HOMEWORK: Every two weeks, 4-5 questions.

DESCRIPTION: In this course we will learn how to reason precisely about computation and prove mathematical theorems about its capabilities and limitations.

We will start by studying the theory of Computability, which is concerned with the rigorous definition of computational tasks and the analysis of automated procedures that may solve them. This will set the stage for the theory of Computational Complexity, whose goal is to examine what are the resources that are necessary for any algorithm to solve a given task. We will end by discussing cryptography and how it makes use of computational hardness.

Topics covered in the course include Turing machines, universality, non-determinism, the halting problem, recursive and recursively enumerable functions, space and time complexity, the classes P and NP, reducibility between decision problems, the Cook-Levin theorem, NP-completeness, encryption and authentication.

OBJECTIVES AND LEARNING OUTCOMES: The most important skill that the students are expected to pick up during this course is the ability to recognize and interpret computational intractability in case it is encountered. The course aims to develop a solid conceptual understanding of notions related to computation:

- The concept of universal models of computation (such as Turing machines), that capture our intuitive notion of computation and allow us to reason about the capabilities of computers in a technology-independent manner.
- The existence of intrinsic limits to computation. Computational problems that cannot be solved by any algorithm whatsoever (undecidability), and problems that are solvable but require unreasonable computational resources (computational complexity).
- The notion of nondeterminism and in particular the conceptual difference between finding a solution and verifying that a given solution is correct.
- The representation of computational problems, and the distinction/relationships between decision and search problems.
- The notion of a reduction between computational problems and its implications on the relative complexity of the problems.

PROGRAM: 12 lectures, for a total of 24h. Each lecture is 2 “sessions”, 45 mins each.

Lecture 1 (Computability - introduction)

- Course overview
- Introduction
- Turing Machine (TM)

Lecture 2 (Computability – more on TM)

- More on the definition of TM
- Decidable and Recognizable languages
- Variants of TM
- Simulation

Lecture 3 (Computability – undecidability)

- The Church-Turing Thesis
- Examples of decidable languages
- The Halting problem

Lecture 4 (Computability – undecidability contd.)

- More non-decidable problems
- Reductions

Lecture 5 (Computability – Rice’s theorem)

- Rice’s Theorem
- Post Correspondence Problem
- Wrap up computability

Lecture 6 (Complexity - Introduction)

- Definition of time complexity
- Complexity of single vs Multiple Tape TM’s
- PTIME, PATH

Lecture 7 (Complexity – The class NP)

- Non-deterministic TM
- Poly-time verifiability
- The classes NP and coNP

Lecture 8 (Complexity – NP completeness)

- Poly-time reducibility
- NP completeness
- Existence of NP-complete problems

Lecture 9 (Complexity – Cook-Levin)

- Cook-Levin Theorem
- More NP-complete problems
- Decision vs. Search

Lecture 10 (Complexity – The class PSPACE)

- Cook/Karp/Levin reductions
- Decision vs. Search
- Coping with NP-hardness
- Space complexity

Lecture 11 (Cryptography - Encryption)

- Perfect secrecy
- Computational secrecy

Lecture 12 (Cryptography - Authentication)

- Message authentication
- Digital signatures