UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Theory of Computation

2526-1-F1802Q151-F1802Q15102

Aims

Knowledge and Understanding

The course in Theory of Computation aims to provide students with the theoretical tools necessary to understand the computational complexity of problems, how they are classified based on their complexity, and the methodologies for solving them—either exactly or with provable approximation performance. It also introduces algorithmic methodologies and advanced data structures to tackle fundamental problems involving large-scale text data.

Specifically, the student will be able to:

- Understand the fundamental concepts of the main computazional model (Turing Machine) and its variants
- Understand which problems can be solved by a computer (decidability) and which cannot be solved in reasonable time (intractability)
- Understand approximation algorithms, i.e., algorithms that provide, in reasonable time, a provably near-optimal solution for problems that are intractable to solve exactly
- Understand parameterized algorithms, i.e., algorithms that allow exact solutions to certain intractable problems by restricting the exponential growth of computation time to a specific instance parameter that is limited in practical cases
- Understand the theoretical foundations and practical applications of exact and approximate string matching, as well as data structures for string indexing

Applying Knowledge and Understanding

The student will be able to:

- Understand how one problem can be reduced to another to demonstrate the relative complexity of problems
- Classify problems based on their computational complexity (e.g., P, NP, NP-complete)

Making Judgements

The student will be able to:

- Assess whether a problem is computable or not
- Evaluate which search algorithm to use in a specific application context
- Evaluate which indexing structure to use in a specific application context

Communication Skills

- Acquisition of the formal language specific to the discipline of Theory of Computation.

Learning Skills

- Development of the ability to abstract from computational problems addressed during the course.

Contents

Basic concepts of theory of computation: Turing machines, decidability, and intractability of computational problems. Reductions between computational problems and classification of problems based on computational complexity. Approximation and parametric complexity. String matching algorithms. Indexing structures.

Detailed program

1. Basic concepts of theory of computation:

- The Turing Machine model of computation and its equivalence with its main variants (multitape machines, non-binary alphabet, nondeterministic machines)
- Relationships between formal languages and computational problems
- Limits of computability: undecidability of the Halting Problem

2. Tractability and intractability, i.e., classification of problems based on computational complexity:

- Decidability in polynomial time on deterministic machines (P, coP) and on nondeterministic machines (NP, coNP)
- Equivalence of nondeterministic polynomial-time decidability and deterministic polynomial-time verifiability
- Polynomial reductions between decision problems
- NP-completeness of the Satisfiability Problem (SAT)
- Proofs of NP-hardness and NP-completeness

3. Approximation complexity
4. Parameterized algorithms
5. Exact string matching algorithms

- Finite state automaton
- Knuth-Morris-Pratt algorithm
- Algorithm based on the shift-and paradigm

6. Approximate string matching algorithms

- Algorithm based on the shift-and paradigm

7. Text indexing structures

- Suffix Array
- Burrows-Wheeler Transform
- FM-index

Prerequisites

Basic concepts of formal languages. Basic concepts of algorithms and data structures.

Teaching form

Lectures and classroom exercises.

All activities are conducted in person and are not recorded or streamed.
Teaching is delivered in Italian.

There are 28 lectures of 2 hours each, conducted in a lecturing format initially and then in an interactive format.

Textbook and teaching resource

Slides and written notes.
Book: Sipser, Michael. Introduction to theory of computation.

Semester

First semester.

Assessment method

The assessment of learning consists of a written exam.

The written exam is based on open-ended questions related to the concepts and techniques presented in the course, as well as exercises requiring the application of the learnt concepts and techniques.

The written exam is evaluated based on the correctness and completeness of the answers.

Two mid-term written exams are scheduled.

Office hours

Office hours are by appointment to be arranged with the instructors.

Sustainable Development Goals