UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Algorithms and Data Structures

2324-1-E3101Q107

Aims

The aim of the course is to teach to design, evaluate and implement efficient algorithms, making use of the most appropriate progamming techniques and data structures.

We will introduce recursive programming and Divide-and-conquer technique, and main data structures. Students will learn to evaluate which programming technique and which data structures are more adequate to solve various types of computational problems

Contents

Basic techniques to develop algorithms and to analyse their efficiency. Introduction to the use of fundamental data structures.

Detailed program

- Introduction and basic definitions: algorithm, problem, instance.
- Computational complexity analysis of algorithms.
- Recursive programming and Divide-and-Conquer programming technique: Mergesort and Quicksort.
- Time complexity for recursive algorithms: recursive equations.
- Linear time sorting algorithms.
- Basic data structures: arrays, linked lists, stacks, queues.
- Binary trees and Search Binary Trees.
- Heap and priority queues. Heapsort.
- Graphs and graph representation.

- Traversing algorithms for graphs: BFS and DFS

Prerequisites

Basics of Computer Programming

Teaching form

Theoretical lectures, exercises, and practical implementation of proposed algorithms. Further exercises are available online, through an E-learning website.

The course is taught in Italian.

Textbook and teaching resource

T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, Mit press ed.

Further material available through the e-learning website.

Semester

Second semester

Assessment method

Written examination, maximum mark 30/30. The exam consists of

- Exercises to be solved by developing an algorithm to solve a given computing problem
- Simulations on specific inputs for algorithms illustrated during the lectures
- Open questions concerning theoretical aspects discussed during the lectures

Score for theoretical questions: 8 points (in total)

Up to 2 more points can be given in case of extremely good exercises.

The exam can be substituted by two intermediate exams, each evaluating some of the subjects covered during the course.

Split examination:

Written examination can be replaced by two split short exams, held during mid course break and at the end of the course. Such possibility is only allowed to first-year students. Each short exam consist of:

- An exercise to be solved by developing an algorithm to solve a given computing problem
- A simulation on specific inputs of algorithms illustrated during the lectures
- Open questions concerning theoretical aspects discussed during the lectures

The maximum score for each short exam is 15/15. Final mark is obtained by summing the marks of the two short exams. Students can repeat one of the two short exams (in case of non sufficient result, or to improve the result) during the exam in july.

Up to 2 more points can be given (considering both short exams) in case of extremely good exercises.

Office hours

By appointment

Sustainable Development Goals