UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

SYLLABUS DEL CORSO

Laboratorio di Informatica

2627-1-E3005Q005

Aims

Knowledge and understanding
Students will learn the conceptual basis of computer programming, and fundamental notions about the typical structure of digital computers, and about operating systems.

Applied knowledge and understanding
Students will acquire the ability to design and implement, in the C language, data structures and algorithms for solving problems in scientific fields, and in a more general setting.

Independent judgment
Lectures and laboratory activities will allow students to develop the ability to analyse algorithms and software.

Contents

The notions of algorithm, and of computational cost of an algorithm.

Basics of imperative programming: data structures and flow control.

Basic notions on C programming, the functional structure of computers and on operating systems.

The representation of data; errors and approximations in numerical computations.

Examples of algorithms and strategies for algorithm design.

Detailed program

The notions of algorithm, and of computational cost of an algorithm.

The functional structure of digital computers. Nature and functions of operating systems.

Representation of data. The binary notation. Representation of integers and of real numbers. Floating-point representation. Approximation and errors.

Principles of imperative programming. Variables and types, primitive instructions, control structures (sequence, choice, iteration). Functions and parameter passing. Examples in *C*.Structured types: arrays, matrices*,* records. Dynamic data types: linked lists, trees, hash tables. Examples in *C*.

Pseudorandom sequences and application to simulation problems.

Strategies for the design of data structures and algorithms. Examples of relevant algorithms.

Basic notions on programming paradigms (imperative, functional, logic, object-oriented), interpreted languages, and the logical and mathematical foundations of computer science. Hints on the use of LLMs in software development.

Prerequisites

Basic notions of propositional logic. Basic notions of mathematics (functions and relations, simple combinatorics).

Teaching form

Instructional teaching: Lectures (2 CFU, 16h)
Interactive teaching: Laboratory activity (2 CFU, 24h).

Lectures are given in Italian.

Textbook and teaching resource

Notes prepared by the teacher, exercises with commented solutions, published on the course site.

1. JG Brookshear, SG Kochan, *Fondamenti di informatica e programmazione in C*, Pearson, 2014.

2. BW Kernighan, DM Ritchie, *Il linguaggio C*, seconda edizione, Pearson, 2004.

Semester

Second semester.

Assessment method

The exam includes a written part and an oral part, both mandatory.

Written exam: the exam consists in one or more C programming tasks.

Oral exam: discussion about the written exam, and questions on the content of the course.

The evaluation is comprehensive.

There are no in-progress tests.

Office hours

Luca Bernardinello: on appointment.
Federico Nati: on appointment.

Sustainable Development Goals