UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Programming 1

2324-1-E3101Q105

Aims

To learn the basics of imperative programming in Java

Contents

The course will introduce the basis of high level programming languages, with a deeper focus on the imperative paradigm. The Java language will be used, and students will code simple algorithms.

Detailed program

1. Logical structure of a computer and data representations. Classes of programming languages, compilers vs. interpreters. Concepts of program correctness (syntax, semantics). The Java Virtual Machine. Algorithms and programs
2. Primitive data types. Variables, declarations and assignments. Expressions and their evaluation. Type checking
3. Strings. Input/Output via keyboard/screen (the 'Scanner' class)
4. Control structures: sequence, selection, iteration. Arrays with primitive types
5. Simplified application of logic development of a program
6. Class methods, definition and invocation. Primitive type parameters. Dot notation. Activation records. Class variables and constants
7. References: memory management in Java (heap, stack, and program execution). Arrays of references. Class methods with complex type parameters: definition and invocation. Passing parameters by value and by reference
8. Programming with recursion, recursive methods and stack handling

Prerequisites

Mathematical-logical skills at the high-school level

Teaching form

The course comprises usual lectures, exercises in the classroom, and assisted sessions in the laboratory.

Moreover, slides and exercises will be made available online.

The course is taught in Italian.

Textbook and teaching resource

Textbook (the English version is also available):

- Walter Savitch - Programmazione di base e avanzata con Java - Seconda Edizione, 2018, Pearson Education Italia, ISBN 9788871929613

Lecture notes written by the teachers.

Semester

First semester

Assessment method

Learning assessment includes a written exam and an oral exam.

The written exam consists of:

- multiple-choice and open questions, whose aim is to test student's knowledge of various parts of the program;
- some programming exercises developed on a computer using the Java programming language. In this case, the aim is to verify that the student is able to apply the programming techniques which have been presented during the course, and to implement such techniques in the Java programming language.

Each of the two parts will result in a score out of thirty and the final mark of the written part will be the average of the two partial scores.

The written exam could be replaced by two intermediate tests, each performed on some of the subjects covered during the course. They will be organised and evaluated in the same way as the written exam.

During the oral exam, besides discussing the contents of the written exam, some questions may be posed on the subjects of the course.

Office hours

On appointment

Sustainable Development Goals