COURSE DETAILS

"FONDAMENTI DI INFORMATICA"

SSD ING-INF/05

DEGREE PROGRAMME: BACHELOR DEGREE IN COMPUTER ENGINEERING

ACADEMIC YEAR: 2023-2024

GENERAL INFORMATION – TEACHER REFERENCES

TEACHER: MULTIPLE STUDY COURSE
PHONE:
EMAIL:

SEE THE STUDY COURSE WEBSITE

GENERAL INFORMATION ABOUT THE COURSE

INTEGRATED COURSE (IF APPLICABLE): N.A.
MODULE (IF APPLICABLE): N.A.
CHANNEL (IF APPLICABLE): N.A.
YEAR OF THE DEGREE PROGRAMME (I, II, III): I
SEMESTER (I, II): I
CFU: 9

REQUIRED PRELIMINARY COURSES (IF MENTIONED IN THE COURSE STRUCTURE "REGOLAMENTO")

None.

PREREQUISITES (IF APPLICABLE)

None.

LEARNING GOALS

Provide the basics for computer science disciplines, introducing the student to the study of the theoretical foundations of computer science, computer architecture and high-level programming languages. Provide the necessary knowledge for the development of programs for solving problems of limited complexity.

EXPECTED LEARNING OUTCOMES (DUBLIN DESCRIPTORS)

Knowledge and understanding

The student must demonstrate knowledge, understanding and be able to describe the basic concepts related to theoretical computer science, computer architecture and high-level programming languages. In addition, the student must demonstrate knowledge and understanding of programs for solving problems of limited complexity.

Applying knowledge and understanding

The student must demonstrate to be able to apply the knowledge learned for the solution of simple programming problems, designing and developing programs for the solution of problems of limited complexity.

COURSE CONTENT/SYLLABUS

2 CFU (8 lessons)

The concept of processing and algorithm. **Models in Computer Science. Finite state automata: definition, graph and table. Turing machine. Computability.**
Boolean algebra: definitions and De Morgan's theorem. Boolean functions. The algebra of the logic of propositions. The encoding and representation of information. Representation of natural, relative, real numbers.
Fundamentals of architecture of computing systems: the Von Neumann model, processor operation. Memories, Input/Output.
The operating system. The life cycle of a program. Translators and interpreters. Programming languages: grammars; the Backus-Naur Form.

7 CFU (24-28 lessons)

Programming: simple structured data types; elementary instructions and control structures. Structured programming. Array. Standard subprograms and libraries.
Dynamic allocation and pointers. Algorithms on sequences and arrays. Structures and strings. Input/Output operations to mass storage.
Abstract data types: lists, stacks, queues. Search and sorting algorithms.
The C language. Use of a program development environment with examples of fundamental algorithms and abstract data type management. **Object-oriented programming elements.**

READINGS/BIBLIOGRAPHY

A. Chianese, V. Moscato, A. Picariello, C. Sansone, "*Le radici dell'Informatica: dai bit alla programmazione strutturata*", Maggioli Editore, 2017.
E. Burattini, A. Chianese, A. Picariello, V. Moscato, C. Sansone, "*Che C serve? Per iniziare a programmare*", Maggioli Editore, 2016.
MOOC "Fondamenti di Informatica" disponibile sulla piattaforma Federica.EU (www.federica.eu)

TEACHING METHODS

Teachers will use: a) lectures for about 60% of the total hours, b) and exercises on the development of programs in C++ language for about 40% of the total hours.
The exercises are carried out in the classroom and / or in the laboratory with the use of an integrated development environment and through platforms for virtual educational laboratories.

EXAMINATION/EVALUATION CRITERIA

a) Exam type:

Exam type	
written and oral	X
only written	
only oral	
project discussion	
other	X

In case of a written exam, questions refer to:	**Multiple choice answers**	
	Open answers	X
	Numerical exercises	

b) Evaluation pattern: L'esito della prova di programmazione è vincolante ai fini dell'accesso alla prova orale.