

COURSE OF STUDY	THREE-YEAR BACHELOR PROGRAMME IN MATHEMATICS
ACADEMIC YEAR	2023-2024
ACADEMIC SUBJECT	COMPUTER SCIENCE

General information	
Programme year	First
Term	First semester (October 2, 2023 – January 15, 2024)
European Credit Transfer and Accumulation System credits (ECTS)	6
SSD	INF/01
Language	Italian
Mode of attendance	Not mandatory

Lecturer	
Name and surname	Marco Polignano
E-mail	marco.polignano@uniba.it
Telephone	+39 080 544 2312
Department and office	Department of Computer Science, room 759, 7 th floor
Virtual meeting room	Microsoft Teams: 22zgorv
Web page	
Office hours	Monday 3:30 p.m.-5:30 p.m. (for optimal organization, please make appointment by email)

Work schedule				
	Total	Lectures	Hands-on learning (recitations/laboratories)	Self-study
Hours	150	32	16	102
ECTS credits	6	4	2	

Learning objectives	
	Acquire basic concepts regarding computer architectures. Acquire basic concepts regarding programming methods and techniques. Apply these concepts to solve problems by using the Python language. Basic notions and concepts regarding algorithms and computational complexity.

Course prerequisites	
	Elementary notions of logic provided by upper secondary schools.

Syllabus	
Course contents	<ul style="list-style-type: none"> - Introduction to the course - Introduction to computer architecture and networks - Introduction to programming <ul style="list-style-type: none"> o Flowcharts o Programming languages o Structured programming o Variables and functions o Data structures o Programming constructs, conditions and loops

	<ul style="list-style-type: none"> ○ Sorting and searching algorithms ○ AlgoBuild - Introduction to Computability <ul style="list-style-type: none"> ○ Compiler and Interpreter ○ Turing Machine - Introduction to the Python Language <ul style="list-style-type: none"> ○ Conditions, Loops and Variables ○ Functions ○ Numpy and Pandas Libraries
Reference books	Informatica. Algoritmi, architetture, linguaggi, applicazioni. Autori: Schneider Michael , Gersting Judith. ISBN: 9788891644817. 2020, Apogeo, Maggioli Editore.
Additional course materials	The lecturer will provide students with supplementary material, such as scientific articles, guides or tutorials.
Repository	-

Expected learning outcomes	
Knowledge and understanding	<ul style="list-style-type: none"> - Recognize components of the architecture of an electronic computer. - Formally describe a programming language. - Be familiar with the basics of programming. - Be familiar with the basics of computer science.
Applying knowledge and understanding	<ul style="list-style-type: none"> - Recognize and use data structures and primitive data types. - Use basic programming constructs. - Correctly use environments and tools to solve complex problems using a software program.
Soft skills	<i>Making judgements:</i> <ul style="list-style-type: none"> - Decide which data structures and programming constructs to use in solving complex problems. - Identify errors in software program code and correct them independently. - Choose appropriate software and programming languages to solve complex problems.
	<i>Communication skills:</i> <ul style="list-style-type: none"> - Use Computer Science terminology correctly.
	<i>Learning skills:</i> <ul style="list-style-type: none"> - Independently solve complex problems that require the aid of a software program. - Understand and resolve errors in software code.

Teaching methods	
	Frontal classroom teaching and guided exercises.

Assessment	
Assessment methods	Exam sessions (upon completion of teaching). Examination with written exercises and theoretical open-ended questions. The test will last 3:00h.
Evaluation criteria	<ul style="list-style-type: none"> • <i>Knowledge and understanding:</i>

	<p>Ability to learn fundamental concepts regarding programming methods and techniques and general concepts of Computer Science.</p> <ul style="list-style-type: none"> • <i>Applying knowledge and understanding:</i> Ability to apply learned concepts to solve and implement solutions to problems of various kinds. • <i>Making judgement:</i> Ability to judge the consistency of the logical structure used to construct algorithms. Ability to identify appropriate tools for solving programming problems. • <i>Communication skills:</i> Learn the methods and languages of computer science to understand texts, communicate learned knowledge, describe, analyze and solve algorithmic problems. • <i>Learning skills:</i> Acquire lasting learning methods, supported by contextual research and solving exercises and questions asked periodically throughout the course.
Grading policy	The final grade is given in thirtieths. The exam is considered passed when the grade is greater than or equal to 18. A score will be given for each individual question until a total of 32 points is reached. Honors will be awarded with scores above 30.

Further information	
	Attendance is recommended, especially participation in the practical exercises, which also include classroom performance of exam papers from previous years. For non-attending students, it is suggested that they consult the lecturer during office hours to verify the correctness of the proposed solutions.