COURSE OF STUDY Physics (L-30)
ACADEMIC YEAR 2023-2024
ACADEMIC SUBJECT Informatics

General information	
Year of the course	First
Academic calendar (starting and ending date)	Second semester (March 4, 2024 – June 7, 2024)
Credits (CFU/ETCS):	8
SSD	Information Processing Systems (ING-INF/05)
Language	Italian
Mode of attendance	Not mandatory

Professor/ Lecturer	
Name and Surname	Teresa Maria Altomare Basile
E-mail	teresamaria.basile@uniba.it
Telephone	+39 080 544 2235
Department and address	Department of Physics (second floor, room 235)
Virtual room	/
Office Hours (and modalities: e.g., by appointment, on line, etc.)	Tuesday 10:30-12:30 (by appointment in person or remotely via Microsoft Teams) (to be scheduled by e-mail)

Work schedule			
Hours			
Total	Lectures	Hands-on (laboratory, workshops, working groups, seminars, field trips)	Out-of-class study hours/ Self-study hours
200	40	45	115
CFU/ETCS			
8	5	3	

Learning Objectives	Fundamentals of problem solving, computer architecture and programming.
Course prerequisites	According to didactic regulations

Teaching strategie	Lectures (with slides). Programming laboratory exercises.
Expected learning outcomes in terms of	
Knowledge and understanding on:	o Knowledge of computer architecture, information representation, basics principles of computer programming and programming language Python.
Applying knowledge and understanding on:	o Ability to autonomously perform software development life-cycle phases: planning, coding, testing. Ability to appropriately use the programming language Python.
Soft skills	• *Making informed judgments and choices* o Ability to identify a solution strategy after the problem analysis phase and evaluate the correctness, efficacy and efficiency of the proposed solution according to theoretical and practical acquired knowledge. o Ability to identify the appropriate tools in the development. • *Communicating knowledge and understanding* o Knowledge of appropriate language and formalism necessary to communicate the acquired knowledge and to describe, analyze and solve problems.

	• *Capacities to continue learning* o Ability to study independently and to consult and make use of manuals of different programming languages.
Syllabus	
Content knowledge	1. Basic notions. Computer architecture: the von Neumann machine. Machine Language. Data encoding and storage. Program execution: the fetch-decode-execute cycle. Arithmetic/Logic Instructions. 2. Problem Solving. Problems and algorithms. Stepwise refinement. Decomposition methods: sequence, selection, iteration, recursion. Structured programming. Algorithm representation: flow-chart and pseudocode. 3. Algorithms and programs. The procedural programming paradigm. Variables and data types. Control statements. Procedural Units. Variable scope and duration. Formal and actual parameters. Passing parameters methods: by value and by reference. Recursive function and the call stack. Translation process: interpreter and compiler. 4. Introducing Python. Statements and syntax. I/O statements. Data types and operations. Conditional control flow: if, if/else and elif statements. Iterative control flow: while and for loops. Defined vs undefined loops. Nested loops. Function basics and module packages. Recursive functions. Dynamic data types: Lists, Dictionaries, Set. Syntax, basics proprierties and operations. File stream: text and csv. Object Oriented Programming basics. Classes and OOP in Python. 5. Common algorithms: maximum/minimum/sum/average, swap, counting. Common array algorithms: computing statistics (maximum, minimum, sum, mean, standard deviation, mode, median), reversing, duplicate removing, computing and visualizing histograms, fusion, partition. Searching algorithms: sequential search and binary search. Sorting algorithms: insertion sort, selection sort, bubble sort, merge sort. Common 2D-array (Matrix) algorithms: sum, product, transpose. Comparison of Iterative and recursive algorithms: factorial, Fibonacci, Greatest commomn Divisor (GcD), binary search. 6. Programming laboratory exercise: design and implementation of algorithms to solve general and simple problems.
Texts and readings	- J. Glenn Brookshear, Dennis Brylow. Informatica. Una panoramica generale. Pearson - Cay S. Horstmann, Rance D. Necaise. Concetti di informatica e fondamenti di Python. Apogeo - Tony Gaddis. Introduzione a Python. Pearson
Notes, additional materials	*Slides of the lectures and programming laboratory exercises*
Repository	*Slides of the lectures and programming laboratory exercises posted on the course homepage*

Assessment	
Assessment methods	Laboratory test and oral exam; passing the laboratory test is a prerequisite for taking the oral exam. The laboratory test consists of defining, coding and testing a solution strategy for a set of simple problems. The oral exam starts with the discussion of the student's work on the laboratory test, followed by the discussion of theoretical and practical aspects of computer and computer programming.
Assessment criteria	• *Knowledge and understanding* o Knowledge of computer architecture, information representation, basics principles of computer programming and programming language Python. • *Applying knowledge and understanding* o Ability to propose a correct and efficient solution strategy and to implement an application tool to solve simple problems.

	• *Autonomy of judgment* o Ability to appropriately select and apply the theoretical and practical tools according to correctness and efficiency of the proposed solution for the given problems. o Accuracy and clarity in the oral exposition. Develop critical sense in discussing the most correct methodologies to solve problems • *Communicating knowledge and understanding* o Accuracy in solving problems and precision in exposing concepts. • *Communication skills* o Correct use of the specific vocabulary of the teaching and ability to discuss theoretical and practical aspects with precision. • *Capacities to continue learning* o Ability to identify the context of each concept and apply the learned programming skills in different application domains.
Final exam and grading criteria	The final grade is out of thirty. The exam is passed when the grade of both laboratory test and oral exam is greater than or equal to 18. The final grade is determined by both the laboratory test (50%) and the oral exam (50%).
Further information	