Syllabus

Descrizione corso

Titolo insegnamento	Introduction to Programming
Codice insegnamento	76401
Titolo aggiuntivo	
Settore Scientifico-Disciplinare	INFO-01/A
Lingua	Inglese
Corso di Studio	Corso di laurea in Informatica e Management delle Aziende digitali
Altri Corsi di Studio (mutuati)	Bachelor in Computer Science
Docenti	prof. Chiara Ghidini, Chiara.Ghidini@unibz.it https://www.unibz.it/en/faculties/engineering/academic-staff/person/49601 dr. Xiaozhou Li, Xiaozhou.Li@unibz.it https://www.unibz.it/en/faculties/engineering/academic-staff/person/51393
Assistente	
Semestre	Primo semestre
Anno/i di corso	1
CFU	9
Ore didattica frontale	60
Ore di laboratorio	30
Ore di studio individuale	135
Ore di ricevimento previste	
Sintesi contenuti	• Data types and expressions • Basic data structures and generics • Functions and parameter passing • Conditionals and loops • Arrays and collections • Classes and objects

	• Basic Input/Output • Exception handling • Recursion
Argomenti dell'insegnamento	Data types and expressions • Basic data structures and generics • Functions and parameter passing • Conditionals and loops • Arrays and collections • Classes and objects • Basic Input/Output • Exception handling • Recursion
Parole chiave	Programming, Algorithms, Java, Object Oriented
Prerequisiti	There are no specific prerequisites. Basic notions of mathematics and set theory will be used.
Insegnamenti propedeutici	
Modalità di insegnamento	Frontal lectures interleaved with exercises, labs with exercises, individual programming projects.
Obbligo di frequenza	Not mandatory, but highly recommended.
Obiettivi formativi specifici e risultati di apprendimento attesi	Type of course: "di base" for L-31 Scientific area: "Formazione informatica di base" for L-31 The objective of the course is to teach the fundamental principles of programming. We will focus especially on imperative programing as the basic way to learn: (1) the basics of programming and programming elements; (2) the basics of algorithmic thinking; and (3) The basics of writing code. As programming language, we will use a subset of the Java language, mainly restricted to its imperative part. The student will learn how programs can be constructed, and also structured in more files/objects in order to solve a problem. Students will learn how to solve computational problems with well-designed programs that implement effective solutions. The learning will be based on examples, from very simple ones to more complex. We will use the Java programming language and the integrated development environment (IDE), so the goal is to train the student capability to develop java applications in this environment. The

	final objective for the student is to acquire the ability to solve basic algorithmic problems in a Java-based application. Knowledge and understanding: • D1.3 - Know the basic principles of programming. Applying knowledge and understanding: • D2.2 - Ability to solve algorithmic problems using programming methods. • D2.17 - Know how to manage small projects for the development of information systems and how coordinate small working groups. Communication skills • D4.5 - Ability to collaborate in interdisciplinary teams to achieve IT objectives. Learning skills • D5.1 - Learning ability to undertake further studies with a high degree of autonomy.
Obiettivi formativi specifici e risultati di apprendimento attesi (ulteriori info.)	
Modalità di esame	The assessment consists of • a programming project • a final written exam. The project is designed to evaluate learning outcomes related to the application of acquired knowledge, critical thinking, communication, and learning skills. Specifically, students are expected to design a computer application capable of effectively solving a given problem. The written exam assesses knowledge and understanding, the ability to apply that knowledge, and the student's learning skills. It includes verification questions, transfer-of-knowledge questions, and practical exercises.
Criteri di valutazione	The project accounts for 40% of the final grade (12 points), while the written exam represents 60% (18 points). If the project receives a positive evaluation, the result remains valid for all three regular exam sessions within the academic year.

	The project will be assessed based on the quality of the solution, including ease of use, the relevance and effectiveness of the implemented functions, and the quality of the code, in line with the principles discussed during the lectures. Written exam answers will be evaluated based on their correctness and clarity.
Bibliografia obbligatoria	John Lewis and William Loftus, Java Software Solutions, Pearson, 2018. Kathy Sierra, Bert Bates, Trisha Gee, Head First Java, 3rd Edition. O'Reilly Media, Inc. Cay S. Horstmann, Brief Java: Early Objects. Wiley Subject Librarian: David Gebhardi, David.Gebhardi@unibz.it
Bibliografia facoltativa	The Java Tutorials: https://docs.oracle.com/javase/tutorial/
Altre informazioni	Software used: IntelliJ
Obiettivi di Sviluppo Sostenibile (SDGs)	Innovazione e infrastrutture, Istruzione di qualità