



UNIVERSITÀ DI PAVIA

Anno Accademico 2022/2023

COMPUTER PROGRAMMING, ALGORITHMS AND DATA STRUCTURES

Anno immatricolazione	2022/2023
Anno offerta	2022/2023
Normativa	DM270
SSD	INF/01 (INFORMATICA)
Dipartimento	DIPARTIMENTO DI MATEMATICA 'FELICE CASORATI'
Corso di studio	ARTIFICIAL INTELLIGENCE
Curriculum	PERCORSO COMUNE
Anno di corso	1°
Periodo didattico	Annualità Singola (03/10/2022 - 19/06/2023)
Crediti	12
Ore	110 ore di attività frontale
Lingua insegnamento	INGLESE
Tipo esame	SCRITTO E ORALE CONGIUNTI
Docente	FERRARI STEFANO (titolare) - 6 CFU DONDI PIERCARLO - 6 CFU
Prerequisiti	None
Obiettivi formativi	<p>The course introduces the student to programming in Python and solving computational problems using algorithms. The main notions of imperative programming (variables, expressions, loops, functions, recursion, input / output) and the fundamental elements of object-oriented programming are provided during the first module. The second module illustrates the most important data structures (linear, trees and graphs) and the main algorithms that work on them. The students will also learn how to analyze algorithms and how to use them to solve problems of medium complexity.</p>
Programma e contenuti	<p>Module 1: Computer Programming</p> <p>Imperative programming</p>

- top-down / bottom-up programming
- values, variables, expressions
- I/O instructions
- constructs, selection, loop
- functions, recursion
- I/O file
- libraries

Object-oriented programming

- fields and methods
- encapsulation, abstraction, inheritance, and polymorphism
- data types (arrays, lists)

+++++

Module 2: Algorithms and Data Structures

Introduction

- Definitions of algorithm and structured data
- Methods for algorithms analysis (Big-O Notation, worst case, average, best case)

Main data Structures

- Linear (stack, queues, linked lists)
- Trees
- Graphs

Search and Sort

- Search (binary search, binary search trees, hash table)
- Sort (SelectionSort, InsertionSort, BubbleSort, HeapSort, MergeSort, QuickSort)

Algorithms on trees and graphs

- visit/traversal
- connected components
- topological sorting
- minimum spanning tree
- shortest path

Main algorithm approaches:

- Divide and Conquer
- Greedy algorithm
- Dynamic Programming

Metodi didattici

For both modules:
Frontal lessons (theory) and laboratories (programming in Python)

Testi di riferimento

For Module 1:

Think Python: How to Think Like a Computer Scientist by Allen B. Downey
Beijing: O'reilly Media

For Module 2:

Problem Solving With Algorithms and Data Structures Using Python,
2nd edition, By Brad Miller and David Ranum
Franklin Beedle & Assoc

(Optional) Introduction to Algorithms, 3rd edition By Thomas H. Cormen,
Charles E. Leiserson, Ronald L. Rivest and Clifford Stein
The MIT Press

**Modalità verifica
apprendimento**

Module 1:

Project ...

Module 2:

Written test (open questions and exercises about theory)

Programming (solve a problem applying the concepts learned during the
theoretical and laboratory lectures)

Altre informazioni

**Obiettivi Agenda 2030 per lo
sviluppo sostenibile**

[\\$lbl_legenda_sviluppo_sostenibile](#)