

Academic Year: 2025/26

21220 - Programming I

Teaching Guide Information

Academic Course: 2025/26

Academic Center: 304 - Faculty of Law and Economics
332 - Faculty of Economic and Business Sciences

Study: 3327 - Bachelor's degree in International Business Economics

Subject: 21220 - Programming I

Credits: 5.0

Course: 3 and 4

Teaching languages:

Theory: Group 1: English

Group 2: English

Seminar: Group 101: Pending

Group 102: Pending

Group 201: Pending

Group 202: Pending

Teachers: Jesica de Armas Adrián, Jose Fernando Moreno Gutierrez

Teaching Period: First term o Third term

Presentation

The aim of this course is to provide an introduction to programming and the Python language. Students are introduced to computation and complexity, learn the core programming concepts, and start managing the most common libraries in the scientific and data analysis fields. The main idea of this course is to get students coding Python quickly. Since companies worldwide are using Python to harvest insights from their data and gain a competitive edge, this course also provides very initial concepts on Machine Learning through Python.

Learning outcomes | Specific learning outcomes

- Identify core aspects of programming and features of the Python language
- Model problems from a computational point of view through abstraction
- Understand and apply core programming concepts like data structures, conditionals, loops, variables, and functions
- Design and write fully-functional Python programs using commonly used data structures, custom functions, and reading and writing to files
- Understand the complexity of an algorithm and code efficient algorithms
- Manage some of the most common libraries in the scientific and data analysis fields
- Apply very basic data science techniques using Python

Contents

1. Computational thinking
 - Abstraction, decomposition
 - Algorithm, pattern recognition, representation
 - Computation
 - Flow diagrams, pseudocode, and high-level code
2. Programming languages
 - Interpreted vs Compiled
 - Why Python
3. First steps in programming with Python
 - Variables, data types and casting
 - Operators
 - Conditionals
 - Loops
 - Inputs/outputs
 - Functions
 - File handling
 - Advanced data structures: Tuples, Sets, Dictionaries
 - Dealing with errors and bugs
4. Algorithms
 - Complexity: O , Ω , Θ
 - Recursion
 - Searching mechanisms
 - Sorting mechanisms
5. Managing common libraries in Python
 - NumPy: Working with arrays
 - Pandas: Analyzing data
 - Seaborn: Statistical data visualization
 - Matplotlib: Graph Plotting
6. Introductory concepts on Machine Learning with Python
 - Mean, median and mode
 - Standard deviation
 - Data distribution
 - Linear and Non-Linear Regression
 - Decision Trees
 - Scale, train, and test

Evaluation and grading system

Seminar and homework exercises: Students will develop different exercises during the course. Some of them must be developed totally or partially as homework, while other exercises must be carried out during the seminar sessions. The average grade will be considered. This is the 50% of the final grade and cannot be retaken.

Participation in theory classes: There are 2 theory sessions per week, during the 10 weeks of the course. Students will actively participate in these sessions, since different challenges are proposed. This part is a continuous evaluation, so it cannot be retaken and it is the 15% of the final grade.

Final project: This project consists of 2 different parts. Each part counts the same, and globally it will be 35% of the final grade. **The development and delivery of both parts of the project are compulsory to pass the course. The minimum grade required is 5. Otherwise, students can retake this part once, developing a different project proposed by the teacher.**