

| | | | | | |
|------------------|--------------------------------------|---------------|----------------|-------------------------------------|---------------------|
| Course Unit | Programming II | | Field of study | Informatics | |
| Bachelor in | Electrical and Computers Engineering | | School | School of Technology and Management | |
| Academic Year | 2023/2024 | Year of study | 1 | Level | 1-1 |
| Type | Semestral | Semester | 2 | ECTS credits | 6.0 |
| | | | | Code | 9112-742-1205-00-23 |
| Workload (hours) | 162 | Contact hours | T 30 | TP - | PL 30 |
| | | | TC - | S - | E - |
| | | | OT - | O - | |

T - Lectures; TP - Lectures and problem-solving; PL - Problem-solving, project or laboratory; TC - Fieldwork; S - Seminar; E - Placement; OT - Tutorial; O - Other

Name(s) of lecturer(s) Adília Isabel Domingues Cruz Alves, Pedro João Soares Rodrigues

Learning outcomes and competences

At the end of the course unit the learner is expected to be able to:

1. Structure a rationale that allows you to outline a solution and build complex programs in Python
2. Apply knowledge in the Python language, with object-oriented programming (PÓO), database and graphical interfaces on desktop and WEB.

Prerequisites

Before the course unit the learner is expected to be able to:

To know the algorithmic bases and programming bases in Python.

Course contents

Object-oriented programming in Python. Classes, objects, members, constructors, and inheritance. Introduction to databases. Database modeling and normalization. ER diagrams. Python database. Introduction to graphical interfaces in Python. Use of Tkinter to create application interfaces. Web interfaces. Use of flask, HTML and CSS to create web interfaces. The NumPy library in support of engineering problems. Matplotlib.

Course contents (extended version)

1. Object-oriented programming.
 - Classes, objects and constructors
 - Capsulation
 - Access to class members
 - Inheritance
 - Abstraction
 - Polymorphism
 - Class diagram
2. Databases
 - ER diagrams
 - Modelation
 - Table normalization
 - Usage of Python with databases
3. Graphical interfaces
 - Window components
 - Layout
 - Use of Tkinter to create application interfaces
4. WEB interfaces
 - HTML
 - CSS
 - Flask - backend
 - Jinja templates
5. The Numpy library in support of engineering problems
6. Matplotlib

Recommended reading

1. Charles Severance, Sue Blumenberg, et al., " Python for Everybody: Exploring Data in Python 3", Independently published, 2020
2. Adelaide Carvalho, "PRÁTICAS DE PYTHON - ALGORITMIA E PROGRAMAÇÃO", FCA, 2021
3. Ernesto Costa, "PROGRAMAÇÃO EM PYTHON - FUNDAMENTOS E RESOLUÇÃO DE PROBLEMAS", FCA, 2015

Teaching and learning methods

The teaching method is semi-expository, which enables the transmission of the knowledge with continuity and with a minimum expenditure of time. Thus, the classes are more practical, and the most used method is the active one, thus provoking the students' activity through the resolution of practical exercises. The student is also expected to perform tasks during non-contact hours.

Assessment methods

- Alternative 1 - (Regular, Student Worker) (Final, Supplementary, Special)
- Final Written Exam - 100%

Language of instruction

Portuguese

Electronic validation

| | | | |
|--|--|-----------------------------------|--------------------------|
| Adília Isabel Domingues Cruz Alves, Pedro João Soares Rodrigues | Tiago Miguel Ferreira Guimaraes Pedrosa | José Luís Sousa de Magalhaes Lima | José Carlos Rufino Amaro |
| 15-02-2024 | 14-03-2024 | 15-03-2024 | 16-03-2024 |