

PROGRAMAÇÃO IMPERATIVA

Degree in Electrical and Computer Engineering

Code: 10120

Main Scientific Area: Science and Technology Program

Lecturer: Duarte Filipe Oliveira Duque

Language of Instruction: Portuguese

Regime: S1

Contact Hours: 60h Total Workload: 100h

ECTS: 6,0

Objectives

This curricular unit consists on the fundamental concepts of the imperative programming paradigm, covering algorithms and data structures, as well as structured programming. The students should be able to understand and analyze problems, and to plan and develop structured solutions using an algorithmic language, and perform their implementation in the C programming language.

Learning Outcomes

Students should be able to analyze problems and implement solutions in an imperative programming language (C programming language) supported by flowcharts and algorithms. They should be able to understand the codification, compilation and execution process.

Also, they should be able to use conditional and cyclic structures, arrays, strings and pointers.

Course Contents

- 1 - Programming Logic: Generic aspects.
- 2 - Programming fundamental elements and structures
- 3 - Simple sequence algorithms
- 4 - Control structures: Conditional and cyclic structures.
- 5 - Functions, procedures and recursion
- 6 - Complex data types
- 7 - Search and sort algorithms
- 8 - Pointers and memory management (Heap and Stack)

Recommended Bibliography

- António Rocha. "Estruturas de Dados e Algoritmos em C", 3ª Edição, FCA, 2014. ISBN 978-9727227693.

- João Neto. "Programação – Algoritmos e Estruturas de Dados", 3ª Edição, Escolar Editora, 2014. ISBN 978-9725924242.

- Stephen G. Kochan. "Programming in C", 4th ed., Addison-Wesley Professional, 2014. ISBN 978-0321776419.

- Jeffrey McConnell. "Analysis of Algorithms", 2nd ed., Jones Bartlett Learning, 2007. ISBN 978-0763707828.

- Brian Kernighan e Dennis Ritchie. "C Programming Language", 2nd Edition, 1988. ISBN 978-0131103627.

Learning and Teaching Methods

The syllabus was defined with the aim to give to the students the ability of learning models for algorithms representation and, subsequently, learning an imperative programming language (language C).

The presentation, exploration and implementation of algorithm representation techniques are addressed in section 1 of the program syllabus. The remaining points are dedicated to learning the programming language (language C).

Assessment Methods

With this curricular unity it is intended to encourage students to use Computational Thinking methods to solve real-world problems. Students will be encouraged to actively participate in the resolution of programming exercises. The evaluation consists of a theoretical component (CT), composed of two written tests (T1 and T2) and a practical component (CP), which is the development of a computer application, coded in C language, in order to solve a particular problem.

Final Evaluation:

The final evaluation (NF) is given by the following formula: $NF = 50\% \cdot CT + 50\% \cdot CP$

Where, $CT = 50\% \cdot T1 + 50\% \cdot T2$

The minimum score on any of the evaluations (theoretical CT and practical CP) is 9.0.

It is not allowed the evaluation of the practical component in any of the exam periods.

The exam grade only replaces the theoretical component.

Is required a 9.0 minimum score in the exam.