



UNIVERSIDADE LUSÍADA
LISBOA

Course Unit Program
Introduction to Programming

Academic Year 2024/2025

1. Organic Unit

Faculty of Business and Economic Sciences

2. Course

Computer Engineering

3. Cycle

1st

4. Course Unit

Introduction to Programming (L9004)

5. Scientific Area

6. Curricular Year

1st

7. Type of Curricular Unit / Semester

1st Semester



UNIVERSIDADE LUSÍADA
LISBOA

8. Class type and workload (work time)

Theoretical classes:

Practical classes:

Theoretical-Practical Classes 30,00

Tutorial Guidance: 20,00

9. ECTS Credits

6,00

10. Scientific Area Coordinator

11. Regent

Prof. Doutora MARIA INÊS VASCONCELLOS FURTADO

12. Assistants

13. Teaching Language

Portuguese

14. General Objectives

1. Introduce the student to programming in the imperative model.
2. Learn to solve different problems with programming and to use iterative and recursive programming methods.
3. Begin practical use of the C language by practicing coding algorithms in this language.

15. Specific Objectives

- A. Characterize the imperative programming model and the control structures that are part of it.
- B. Study and practice iterative and recursive programming methods.
- C. Introduction to the C imperative programming language
 - C.1 Manipulation of a programming environment IDE (Integrated Development Environment)
 - C.2 Creation of a programming project (introductory level)

16. Skills to be acquired

1. Knowledge of the characteristics of the imperative programming model.
2. Practice solving imperative programming problems.
3. Knowledge of the fundamental instructions and types of the C language.
4. Students, by the end of the course, should be able to analyze and create systems and software (introductory level)



UNIVERSIDADE LUSÍADA
LISBOA

17. Teaching Methodology

1. The theoretical-practical classes are based on practical examples and supported with notes and references.
2. The tutorial classes are dedicated to the practice of the C language and are supported by tutorial lessons.

18. Course Unit Program



UNIVERSIDADE LUSÍADA
LISBOA

1. Introduction

1.1. Computers and programming

- Introduction to basic computer fundamentals and their relationship to programming.

1.2. Programming languages

- Overview of programming languages and their characteristics.

1.3. Algorithm

- Definition of an algorithm as a logical sequence of steps to solve problems.

1.4. Program

- Structure and composition of computer programs.

1.5. Imperative model

- Explanation of the imperative paradigm and its application in problem-solving.

2. Algorithmics and Programming

2.1. Algorithm and algorithmic notations

- Discussion on the importance of clear and efficient algorithms. Use of notations like pseudocode and flowcharts.

2.2. Data types

2.2.1. Concept of variable

- Definition of a variable as a storage location with a name, type, and value.

2.3. Fundamental instructions in imperative programming

2.3.1. Assignment

- Use of assignment operators to store values in variables.

2.3.2. Input/output

- Methods for user interaction and data manipulation.

2.3.3. Control structures: Selection and loops

- Use of conditional structures (if-else) and loops (for, while) to control program flow.

2.4. Routines

2.4.1. Main program

- Organization and structuring of the main program.

2.4.2. Procedures and their input/output parameters

- Definition and use of procedures to modularize code.

2.4.3. Functions and their parameters: pass by value

- Implementation of functions to perform specific tasks, with examples of passing parameters by value.

2.7. Study and manipulation of arrays

- Exploration of one-dimensional structures to store and manipulate data sets.

2.8. Study and manipulation of matrices (bi or n-dimensional)

- Applications of multidimensional structures to represent complex data.

2.9. Introduction to pointers

2.9.1. Concept of pointers

- Explanation of pointers as variables that store memory addresses.

2.9.2. Strings

- Manipulation of strings as arrays of characters in the programming language.



UNIVERSIDADE LUSÍADA
LISBOA

2.9.3. Records (structs)

- Use of structs to organize and manipulate heterogeneous data.

2.9.4. Functions and their parameters: pass by reference

- Use of pointers to pass parameters by reference in functions.

2.10. Text files

- Manipulation of text files for reading and writing persistent data.

19. Assessment Methods

Continuous evaluation is made according to the university's current regulations comprising assiduity, participation and engagement in class. The work requested is the object of evaluation, following the objective to be achieved and the defined schedule.

20. Didactic Resources

Classes will be given, whenever possible, using visual presentations.

Computers in practical classes with CLion, Visual Studio, Dev C++ or similar.

Theoretical-practical lectures and exercises.

21. Sustainability Objectives

22. Keywords

Algorithms

Programming

C Language



UNIVERSIDADE LUSÍADA
LISBOA

23. Main Bibliography

Author(s): Damas, Luís Manuel Dias
Title: Linguagem C
Edition: 1 Year: 1999
Local: Lisboa
Publisher: FCA

Author(s): Rocha, António Manuel Adrego da
Title: Introdução à programação usando C
Edition: 1 Year: 2006
Local: Lisboa
Publisher: FCA

Author(s): Pereira Alexandre
Title: C e algoritmos
Edition: 1 Year: 2017
Local: Lisboa
Publisher: Sílabo

24. Complementary Bibliography

Author(s): Dennis M. Ritchie, Brian W. Kernighan
Title: The C programming language
Edition: 2 Year: 1988
Local: New Jersey
Publisher: Prentice Hall

Author(s): Deitel, Harvey M.; Deitel, P. J.
Title: Como programar em C
Edition: Year: 1999
Local: Rio de Janeiro
Publisher: LTC - Livros Técnicos e Científicos Editora



UNIVERSIDADE LUSÍADA
LISBOA

Ficha técnica MIP / MIP Datasheet	
Título / Title	Course Unit Program 'INTRODUÇÃO À PROGRAMAÇÃO': Academic Year 2024/2025
Autor / Author	Prof. Doutora MARIA INÊS VASCONCELLOS FURTADO
Classificação / Classification	-
Descritores / Descriptors	-
Colaborador(es) / Colaborator(s)	-
Data Hora / Date Time	
Formato de dados / Data Format	Texto / Text, PDF
Estatuto de utilização / Usage	Acesso público / Public Access
Relação / Relation	Version 1,0
(c) Universidade Lusíada, 2024	