

Course Unit	Programming Laboratory	Field of study	Computer Science
Bachelor in	Informatics Engineering	School	School of Technology and Management
Academic Year	2023/2024	Year of study	1
Type	Semestral	Semester	1
Workload (hours)	162	Contact hours	T - TP - PL 60 TC - S - E - OT - O -
		Level	1-1
		ECTS credits	6.0
		Code	9119-706-1103-00-23

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) Luís Manuel Alves, Maria João Tinoco Varanda Pereira, Davide Emanuel da Silva Dias, Nelson Alexandre Perdigo Figueiredo, Pedro Gaspar Padrão Antunes Vilares

Learning outcomes and competences

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

1. Implement algorithms and data structures in an imperative programming language;
2. Design, implement, test and debug programs written in C, using imperative programming techniques;
3. Analyze and explain the behavior of programs written in C;
4. Modify and optimize programs written in C;
5. Develop the skills to work as a team when solving small scale computational problems.

Prerequisites

Before the course unit the learner is expected to be able to:
Not applicable.

Course contents

Installation and exploration of the Integrated Development Environment (IDE); declaration of variables of the different elementary data types; implementation of control structures; implementation of functions with and without arguments; use of the math.h and string.h library functions; declaration and use of vectors and multidimensional arrays; use of functions for manipulation of data files; declaration and implementation of structures; dynamic memory allocation.

Course contents (extended version)

1. Installation, presentation and exploration of the Integrated Development Environment (IDE).
2. Declaration of variables of the different elementary data types.
3. implementation of control structures.
4. implementation of functions with and without arguments.
5. use of the math. h and string. h library functions.
6. declaration and use of vectors and multidimensional arrays.
7. use of functions for manipulation of data files.
8. declaration and implementation of structures.
9. dynamic memory allocation.
10. Coding of a project based on modular programming.

Recommended reading

1. Luís Damas, "Linguagem C", Tecnologias de Informação, FCA, 1999.
2. Pedro Guerreiro, "Elementos de Programação com C", Tecnologias de Informação, FCA, 2006.
3. António Rocha, "Introdução à Programação Usando C", Tecnologias de Informação, FCA, 2006.
4. Brian W. Kernighan e Dennis M. Ritchie, "The C Programming Language", Prentice-Hall, 1988.
5. R. Johnsonbaugh, and M. Kalin, "C for Scientists and Engineers", Prentice-Hall, 1997.

Teaching and learning methods

Laboratory classes where students develop a team project throughout the semester. In line with this project, students solve small programming problems on the computer or perform longer programmer assignments with the help of a formal guideline. Students complete their training through individual or group work, done outside the classroom.

Assessment methods

1. Alternative 1 - (Regular, Student Worker) (Final, Supplementary)
 - Projects - 40% (Final project)
 - Final Written Exam - 60% (component with a minimum score of 7 out of 20)
2. Alternative 2 - (Regular, Student Worker) (Supplementary, Special)
 - Final Written Exam - 100% (includes supplementary exercises designed to replace the final project classification)

Language of instruction

1. Portuguese
2. English

Electronic validation

Lúis Manuel Alves, Maria João Tinoco Varanda Pereira	Tiago Miguel Ferreira Guimaraes Pedrosa	Lúisa Maria Garcia Jorge	José Carlos Rufino Amaro
10-10-2023	25-10-2023	25-10-2023	31-10-2023