UNIVERSIDAD DE MURCIA

FACULTY OF MATHEMATICS

Programming Fundamentals

Syllabus

Course code:	6074
Number of ECTS credits:	6
Semester:	2nd (February-June)
Prerequisites:	None
Recommended components:	User-level management of a computer and the Internet.
Language of instruction:	Spanish (students are allowed to ask questions and write homeworks and exams in English)

Course description

The use of computers makes it easier for students to better understand complex math topics because it allows to know the processes that are carried out within a computer system to obtain the solution to these problems while facilitating repetitive and in some cases tedious calculations. This subject introduces the general aspects of the design of structured algorithms for solving mathematical problems by dealing with primitive and composite data types, handling of functions (operational abstraction), recursion and modularity, as well as the study of classical search and sort algorithms. For this, extensive use is made of the imperative programming paradigm as well as an introduction to the object-oriented programming paradigm.

Learning outcomes and competences

After completion of this course you will:

1. Create technical documents that deal with mathematical problems that contemplate solutions that can be implemented with mathematical software or a programming language.
2. Demonstrate the management of the fundamental elements of structured programming in solving different types of math and computer problems.
3. Design and build structured programs to solve math problems.

Course contents

I. Problem solving and algorithms.

Mathematics and Computing. Problem solving: resolution phase and implementation phase. Algorithms: designs.

II. Introduction to programming.

Program concept. Types of languages. Compilers vs interpreters. Programming paradigms. Numeric and alphanumeric expressions.

III. Variables.

Type of data. Variables and constants. Operators according to the type of data. Types of variables. Casting. Debugging programs.

IV. Structured programming.

Control structure concept. Sequential, selective and repetitive structures. Instructions continue and break. Nested structures.

V. Modular programming.

Functions: declaration and definition. Types of functions. Parameter passing: by value and by reference. Debugging programs with functions. Modularity. Libraries. Code reuse. Tree of calls.

VI. Structured data.

Data structures: types. Dynamic memory reserve. Stack vs heap. Aliasing. Assignment vs copy. String. Array. Records.

VII. Search and sort algorithms.

Sequential search. Non-recursive binary search. Sorting by direct insertion. Sort by direct exchange: bubble method. Sorting by selection. Sorting by lockers.

VIII. Recursion.

Concept. Structural recursion. Types of Recursion. Recursive call tree.

IX. Object Oriented Programming.

Introduction, need and basic concepts of Object Oriented Programming. Classes and objects. The life cycle of an object: definition, declaration, construction, use and destruction.

X. Members and methods.

Customer concept. Difference between functions and methods. Design of methods associated with a class. Use of objects.

References

Main texts

1. Luis Joyanes Aguilar. *Fundamentos de programación: Algoritmos, estructura de datos y objetos (4th edition)*; McGraw-Hill, 2008.

2. Daniel Shiffman. *Learning processing : a beginner's guide to programming images, animation, and interaction*; Morgan Kaufmann/Elsevier, 2008.

3. David M. Arnow. *Introducción a la programación con Java: un enfoque orientado a objetos*; Addison Wesley, 2000.

4. Ben Fry and Casey Reas. *Processing 3*; `https://processing.org/`, 2021.