

30105 - Fundamentals of computer studies

Syllabus Information

Academic year: 2023/24

Subject: 30105 - Fundamentals of computer studies

Faculty / School: 175 - Escuela Universitaria Politécnica de La Almunia
179 - Centro Universitario de la Defensa - Zaragoza

Degree: 425 - Bachelor's Degree in Industrial Organisational Engineering
563 - Bachelor's Degree in Industrial Organisational Engineering

ECTS: 6.0

Year: 563 - Bachelor's Degree in Industrial Organisational Engineering: 2
425 - Bachelor's Degree in Industrial Organisational Engineering: 1

Semester: Second semester

Subject type: Basic Education

Module:

1. General information

Goals:

- To enable to solve a problem by creating simple programs. Therefore, its basic content is programming and, in particular, the specification of problems, the approach of a range of solutions as possible alternative algorithms, the choice of the best solution based on experimentation or previous experience, and the translation of these solutions into programs executable by a computer in a general-purpose programming language.
- To know the constituent elements of a computer, understand its basic operation, search for information and apply programming and troubleshooting skills to the tools and available software applications and tools.

These approaches and objectives are aligned with the next Sustainable Development Goal (SDG) of the United Nations' Agenda 2030 (<https://www.un.org/sustainabledevelopment/es/>), in such a way that the acquisition of the learning results of the subject provides training and knowledge, skills and competencies to contribute to some extent to their achievement:

Company profile

Goal 7: Affordable and Clean Energy

Target 7.3 By 2030, double the global rate of improvement in energy efficiency.

Defense profile

Goal 9: Industry, innovation and infrastructure.

2. Learning results

1. Acquire the ability to retrieve information from digital sources (including browsers, search engines and catalogs).
2. Know the basic operation of computers, operating systems and databases, and create programs about them.
3. Operate computer equipment effectively, taking into account its logical and physical properties.
4. Use ranges for program development.
5. Understand, analyze and propose solutions to information processing problems in the engineering world, engineering world, of low-medium complexity.
6. Specify, design and implement correct programs for problem solving.

3. Syllabus

Company profile

The subject program is structured around two complementary content components:

- Theoretical.
- Practical.

Theoretical contents

Block 1

- Computer: Machine that executes Algorithms. Notion of Algorithm. Computer structure: Digital Nature, coding, hardware, software.
- Operating systems.

- Databases.
- Programming: Programming styles, language hierarchy, programming elements.
- Computer networks.

Block 2

- Function design.
- Text and input/output.
- Conditional.
- Notions of classes and objects.
- - Lists.
- Iteration.

Block 3

- Other collections: sets, tuples, dictionaries.
- Design algorithms.
- Searching and sorting.
- Files.

Block 4

- Classes, objects and methods.

Practical contents

Each topic discussed in the previous section has practices associated with it. As the topics are developed, these practices will be presented, either in class or through the Moodle platform.

Defense profile

1. Computer architecture, hardware and software
2. Programming languages and environment
3. Predefined data types
4. Modular Programming I
5. Control structures
6. Modular Programming II
7. Handling of exceptions and files
8. Structured Data Types I
9. Structured data types II
10. Algorithms

4. Academic activities

Company profile

The time distribution of the activities will be as follows:

- **Lectures (30 hours)**, combining theoretical exposition with the resolution of examples that help to better understand the theory.
- **Practical classes (30 hours)**. Problems and exercises solved by the students during the lessons **Work and personal study (84 hours)**
- **Assessment tests (6 hours)**

Defense profile

In each of the topics of the program, use is made of different activities from among those programmed in the subject to achieve the specific objectives for Fundamentals of Computer Science and some of the general objectives for the degree. A series of activities such as exercises, practices, projects, controls, etc. will be addressed. The sessions will be of a theoretical nature for the presentation of concepts and practical for the resolution of exercises and implementation of programs. Projects/practices may be proposed to the students for their development in the hours of study or in those planned in the laboratory, which may involve a telematic delivery (or other type of assessment) with a deadline announced at the time of publication.

5. Assessment system

Company profile

The assessment process includes two types of actions:

A system of continuous assessment, which will consist of the following group of gradable activities

- practical work done and corrected in class (10%).
- two written tests. The percentage of the overall grade for each written test will be one percent 45%

An overall assessment test, reflecting the achievement of the learning outcomes, at the end of the teaching period. The exam will consist of two parts, corresponding to the written tests of the continuous assessment, which weight 45% and 45%. The grade for the internship is added to the grade for the subject with a 10% of the grade.

The two written tests must be passed separately in order to contribute to the average of the final grade.

The grade for each written exam must be greater than 4 in order to contribute to the final grade average. If at the end of the assessment any of the written tests has a grade lower than 4, the final grade will be a maximum of 4.9, even if the average is higher.

Defense profile

FIRST CALL

Continuous assessment

The student will be able to pass the total of the subject by the continuous assessment procedure. The student must demonstrate that they has achieved the expected learning results through the assessment activities that will be distributed throughout the term.

- Practical Works (20%): a set of exercises supervised by the teacher will be presented and the effectiveness, efficiency, legibility, etc. of the results will be evaluated. . The weight of this practical work will be 20% of the continuous aassessment grade.
- First theoretical-practical test (20%): a series of problems and/or questions will be posed in order to evaluate the students' knowledge. The weight of this practical work will be 20% of the continuous assessment grade.
- First theoretical-practical test (60%): a series of problems and/or questions will be posed in order to evaluate the students' knowledge. The weight of this practical work will be 60% of the continuous assessment grade.

In order to pass the continuous assessment it will be necessary to obtain a grade higher than 3 in the second theoretical-practical test.

Global test:

Students who do not pass the subject by continuous assessment or who would like to improve their grade, will have the right to take the global test set in the academic calendar, prevailing, in any case, the best of the grades obtained. This global test will have a 100% weight in the final grade.

It will consist of two parts: 1) a questionnaire, which will evaluate the knowledge and know-how of everything learned (in lectures, problems, practices, projects...) and 2) some problems, in which the programming skills acquired during the term must be demonstrated.

SECOND CALL

Global test:

Students who do not pass the course in the first exam may sit for an overall exam set in the academic calendar for the second exam. This global test will consist of the same parts as the global test of the first call and will have a weight of 100%.

INSTRUMENTS vs. LEARNING RESULTS (RA)

Assessment instruments:	Weighting	RA-1	RA-2	RA-3	RA-4	RA-5	RA-6
Practical Work	20%	x	x	x	x	x	x
First theoretical-practical test	20%		x		x	x	x
Second theoretical-practical test	60%		x		x	x	x