

Academic Year/course: 2022/23

29500 - Fundamentals of Programming

Syllabus Information

Academic Year: 2022/23

Subject: 29500 - Fundamentals of Programming

Faculty / School: 175 - Escuela Universitaria Politécnica de La Almunia

Degree: 625 - Bachelor's Degree in Industrial Processes' Data Engineering

ECTS: 6.0

Year: 1

Semester: First semester

Subject Type: Basic Education

Module:

1. General information

1.1. Aims of the course

The course shows the basic concepts of algorithm, as well as the tools and techniques necessary for implementation using high-level languages.

These approaches and objectives are in line with the following Sustainable Development Goals (SDGs) of the United Nations 2030 Agenda (<https://www.un.org/sustainabledevelopment/es/>), in such a way that the acquisition of the course learning outcomes provides training and competence to contribute to their achievement to some degree:

Goal 7: Ensure access to affordable, reliable, sustainable and modern energy

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

1.2. Context and importance of this course in the degree

The subject has 6 ECTS credits, which represents 150 hours of student work during the second trimester, been part of the traversal module of data engineering degree.

It provides an additional formation to a data engineer in the computing area.

1.3. Recommendations to take this course

No previous knowledge is necessary.

Basic IT skills recommended.

2. Learning goals

2.1. Competences

Basic and general competences

- CG03 - Apply techniques for data acquisition, management and processing within an engineering context.
- CG04 - Process data and solve problems with ethical, social and professional responsibility.
- CB4 - Ability to transmit information, ideas, problems and solutions to both specialized and non-specialized audiences.
- CB5 - Develop those learning skills necessary to undertake further studies with a high degree of autonomy.

Transversal competences

- CT03 - Search, select and manage information and knowledge responsibly.
- CT07 - Analyze and solve problems autonomously, adapt to unforeseen situations and make decisions.

Specific competences

- CE01 - Design and implement algorithms using high-level languages and methodologies.

2.2. Learning goals

- Know the technical bases necessary for computer programming.
- Know the principles of computing, the design of algorithms to solve problems and their implementation in modern high-level programming languages.

- Have the necessary skills to be able to develop basic programs using the object orientation methodology.

2.3. Importance of learning goals

This subject has a strong engineering character, gives a formation with contents that can be applied immediately in jobs. Achieving the learning goals, the ability to understand the operation of computer systems is obtained, which will be essential in order to design and to put in operation applications and processes.

3. Assessment (1st and 2nd call)

3.1. Assessment tasks (description of tasks, marking system and assessment criteria)

The evaluation process will include two types of action:

- A continuous evaluation system, which will be carried out throughout the learning period.
- A global assessment test, reflecting the achievement of learning results, at the end of the teaching period.

Continuous evaluation system.

Following the spirit of Bologna, regarding the degree of involvement and continued work of the student throughout the course, the evaluation of the subject considers the continuous evaluation system as the most appropriate to be in line with the guidelines set by the new framework from the EHEA.

The continuous evaluation system will have the following group of qualifying activities:

- **Works:** The works will consist of practical exercises, solution to proposed problems, etc. The correctness and quality of the results will be assessed. The percentage with respect to the global mark of all these works will be 30%.
- **Assessment tests:** There are two throughout the course. The percentage with respect to the global mark of each evaluation test will be 35%

It is necessary to pass separately with a grade equal or higher than 4, the assignments and the written tests in order to contribute to the average of the final grade. If at the end of the evaluation any of the parts has a grade lower than 4, the final grade will be a maximum of 4.9, even if the average is higher.

To opt for the Continuous Assessment system, you must attend at least 80% of the face-to-face activities (practices, technical visits, classes, etc.)

Global final evaluation test.

The student must opt for this modality when, due to their personal situation, they cannot adapt to the rhythm of work required in the continuous evaluation system, have suspended or want to upload a grade having participated in said evaluation methodology.

The global final evaluation test will have the following group of qualifying activities:

- **Assessment Test:** It is carried out in the official calls. The exam will consist of three parts, corresponding to the tests of the continuous evaluation, with weights of 35% and 35% for the midterm and 30% for the assignments, which must be passed separately in order to contribute to the final grade. If at the end of the evaluation any of the parts has a grade lower than 4, the final grade will be a maximum of 4.9, even if the average is higher.

4. Methodology, learning tasks, syllabus and resources

4.1. Methodological overview

The course is mainly practical, with the presence of theory. Following methods will be applied.

- **Lectures:** Theoretical activities carried out mainly through exposition by the teacher, where the theoretical supports of the subject are displayed, highlighting the fundamental, structuring them in topics and or sections, interrelating them.
- **Practice Sessions:** The teacher resolves practical problems or cases for demonstrative purposes. This type of teaching complements the theory shown in the lectures with practical aspects.
- **Laboratory Workshop:** These classes are highly recommended for a better understanding of the concepts because those are shown in working mode with the use of specific tools.
- **Tutorials:** Those carried out giving individually, personalized attention with a teacher from the department. Said tutorials may be in person or online.

If classroom teaching were not possible due to health reasons, it would be carried out on-line.

4.2. Learning tasks

Face-to-face generic activities:

- Lectures: The theoretical concepts of the subject are explained.
- Practice Sessions: Problems and practical classes are carried out, complementary to the theoretical concepts studied.
- Laboratory Workshop: Reinforcement of practical classes are carried out with the help of specific tools.

Generic non-class activities:

- Study and understanding of the theory taught in the lectures.
- Understanding and assimilation of the problems and practical classes solved in the practical classes.
- Solving proposed problems, project, etc.

- Preparation of laboratory workshops, preparation of summaries and reports
- Preparation of the written tests for continuous assessment and final exams.

4.3. Syllabus

Theoretical contents

- General computer science concepts.
- Introduction to programming
- Fundamentals of algorithms.
- Data types and operators
- Flow control.
- Data collections
- Functions
- Libraries.
- Files
- Objects

Practical contents

- Each topic exposed in the previous section has associated practices. The programming language will be Python.

4.4. Course planning and calendar

The subject has 6 ECTS credits, which represents 150 hours of student work in the subject during the trimester, in other words, 10 hours per week for 15 weeks of class.

A summary of a weekly timetable guide can be seen in the following table.

- 1 hour of lectures
- 3 hour of laboratory workshops
- 6 hours of other activities

4.5. Bibliography and recommended resources

See next [link](#)