# COURSE DATA

## DATA SUBJECT

**Code:** 34751
**Name:** Informatics
**Cycle:** Undergraduate Studies
**ECTS Credits:** 6
**Academic year:** 2026-27

## STUDY (S)

| Degree | Center | Acad. year | Period |
|---|---|---|---|
| 1401 - Degree in Chemical Engineering | Escola Tècnica Superior d'Enginyeria | 1 | First quarter |
| 1934 - Double Degree Program in Chemistry-Chemical Engineering | Facultat de Química | 1 | First quarter |

## SUBJECT-MATTER

| Degree | Subject-matter | Character |
|---|---|---|
| 1401 - Degree in Chemical Engineering<br>1934 - Double Degree Program in Chemistry-Chemical Engineering | Information technology | BASIC |
| | Primer curso | COMPULSORY |

## COORDINATION

CASAS YRURZUM SERGIO

# SUMMARY

This course tries to show the basics of what a computer is, what are its basic components, potential uses and limitations.

It will introduce students to the understanding and management of the operating system as well as the description and use of the network as a key part in communicating information between computers.

It will also provide a basic knowledge of different tools, as well as a brief introduction to the concept of database.

It will also try to provide a sufficient knowledge of algorithm design using structured programming, as well as fundamental data structures.

Regarding the practical part of this course, the student will try to consolidate the knowledge studied in the

theoretical part of the subject, both in the knowledge of the computer and the basic tools for its use, and acquire software development skills in a language of structured programming of general purpose.

The general objectives of the course are:

- Introduce students to a basic knowledge about the structure of a computer, both at the hardware (processor, memory, etc.) and the software level (operating system, application programs, etc.).
- Introduce students to the use of the network as a tool for information sharing, access and use of remote computers.
- Provide a generic view of what databases are and what are they used for, as well as some practical examples of use.
- Introduce students to the procedural programming methodology by introducing the concept of programming language and the concept of algorithm, as well as different methods of problem solving (refinement, top-down resolution, etc.).
- Introduce students to data types, variables, constants, control structures and the data need to develop programs.

Observations: The classes will be taught in the language as stated in the course sheet available on the website of the degree.

## PREVIOUS KNOWLEDGE

### RELATIONSHIP TO OTHER SUBJECTS OF THE SAME DEGREE

There are no specified enrollment restrictions with other subjects of the curriculum.

### OTHER REQUIREMENTS

## COMPETENCES / LEARNING OUTCOMES

### 1401 - Degree in Chemical Engineering

Acquire knowledge of basic and technological subjects to facilitate the learning of new methods and theories, and develop the versatility to adapt to new situations.

Act autonomously in learning, making decisions based on different contexts, making judgments based on experimentation and analysis and transferring knowledge to new situations.

Collaborate effectively in work teams, assume responsibilities and leadership roles, and contribute to collective improvement and development.

Know how to communicate effectively, both orally and in writing, adapting to the characteristics of the situation and the audience

Use and programme computers, operating systems, databases and software applications relevant to engineering.

Work in a multilingual and multidisciplinary environment.

## DESCRIPTION OF CONTENTS

### 1. INTRODUCTION

Concept of computer: Basic concepts. Internal structure of the computer.
Software: Operating system. Utilities.
Information management.
Computer programs for engineers applications.

### 2. COMPUTER NETWORKS

Introduction and basic concepts.
Utilities to share information.

### 3. PROGRAMMING IN HIGH-LEVEL LANGUAGES

Algorithm
Languages and programming paradigms.
Characteristics of high level programming: Variables and constants.
Simple Data Types.
Phases in conducting a program: Analysis of the problem. Algorithm design and programming.

### 4. STRUCTURED PROGRAMMING

Structured programming.
Design of structured programms.
Control Structures: Sequential structure. Conditional structure. Iterative structure.

### 5. MODULAR PROGRAMMING

Module definition.
Modular programming.
Definition subprogrammes: Functions.
Parameters of a subprogram.
Scope of identifiers.

Recursion.

## 6. STRUCTURED DATA TYPES

Vectors/arrays, matrices, strings and records.

## 7. FILES

Basics concepts of files.
Access types.
Logical and physical files.
Text files.
Processing files.
Data bases.

## 8. COMPUTER LAB

Laboratory practices will be held:
1 .- Network and internet and programming environment
2 .- Input, output and data types
3 .- Selective Structures
4 .- Repetitive structures
5 .- Modular programming
6 .- Vectors and matrices
7 .- Strings and Records
8 .- Files

# WORKLOAD

### PRESENCIAL ACTIVITIES

| Activity | Hours |
|---|---|
| Theory | 30,00 |
| Laboratory | 20,00 |
| Classroom practices | 10,00 |
| **Total hours** | **60,00** |

### NON PRESENCIAL ACTIVITIES

| Activity | Hours |
|---|---|
| Attendance at other activities | 0,00 |
| Individual or group project | 15,00 |
| Independent study and work | 0,00 |
| Preparation of lessons | 65,00 |
| Preparation for assessment activities | 10,00 |

| | |
|---|---|
| Resolution of case studies | 0,00 |
| **Total hours** | **90,00** |

## TEACHING METHODOLOGY

In the theoretical face-to-face activities, the topics of the course will be developed by providing a global and integrative perspective, analyzing in greater detail the key and more complex aspects, and constantly encouraging student participation. These activities are complemented by practical activities aimed at applying basic concepts and expanding them with the knowledge and experience gained during the completion of the proposed assignments. They include the following types of face-to-face activities:

- Problem and question-solving classes in the classroom.
- Discussion sessions and solving problems and exercises previously worked on by students.
- Laboratory practices.
- Completion of individual assessment questionnaires in the classroom in the presence of the teaching staff.

In addition to classroom-based activities, students will be required to carry out personal tasks (outside the classroom) related to questions and problems, as well as the preparation of classes and exams (study). These tasks will be carried out mainly on an individual basis, in order to promote autonomous work. However, there will also be assignments that require the participation of small groups of students (2-4) to foster the ability to integrate into work groups.
The Universitat de València's e-learning platform (Aula Virtual) will be used as a communication support tool with students. Through this platform, students will have access to the teaching materials used in class, as well as the problems and exercises to be solved.

## EVALUATION

The assessment of the course will be carried out through:

- Continuous assessment, based on participation and level of involvement in the teaching-learning process, taking into account regular attendance at in-person activities and the resolution of proposed questions and problems. This part is not recoverable.(N_Continuous).
- Assessment of practical activities based on the achievement of objectives during laboratory sessions and the completion of practical exercises. Written exams and/or oral presentations (individually and/or in groups) may be conducted to evaluate the assimilation of knowledge. For this section, a final programming project may also be required. If such a project is carried out, an individual oral defense may be required after submission. This part is not recoverable, except by explicit and individualized indication from the instructor. (N_Practicals).
- Individual tests, consisting of exams or knowledge tests, which will include both theoretical-practical questions and problems. This section will be divided into a final exam and a midterm. The final exam will account for 70% of this section, while the midterm will account for 30%. If the midterm cannot be held due to scheduling or other reasons, the grade for this section will be that of the final exam. (N_Exams).

First call (ordinary session):
In this session, the final grade will be calculated as follows:
Final Grade = 15% N_Continuous + 35% N_Practicals + 50% N_Exams
It will be necessary to obtain at least a 4 out of 10 in each of the last two parts (N_Practicals and N_Exams) in order to average the grade.

Second call (resit session):
Final Grade = 10% N_Continuous + 20% N_Practicals + 70% N_Exams
In this case, the N_Exams component will only include the final exam and will account for 70% of the final grade, while the other components (N_Continuous and N_Practicals) will still be considered but with a lower weight. Thus, a student who only takes the final exam can obtain a maximum of 70% of the final grade. In the second call, the grades obtained in the first call for N_Continuous and N_Practicals will be retained.

**Early call:**
To request an early call, students must have previously taken the course and obtained a minimum grade of 5 in the assessment of laboratory practical activities (N_Practicals). In any case, the assessment system will be governed by the provisions of the Regulations on Assessment and Grading of the University of Valencia for undergraduate and master's degrees (ACGUV 108/2017).

**Plagiarism:**
Intellectual honesty is vital in academic communities and for the fair assessment of students' work. Therefore, copying or plagiarism (total or partial) of any activity that is part of the assessment will result in the impossibility of passing the course for all those involved in the copying or plagiarism, regardless of the origin or destination of the copy. This is without prejudice to all those involved being subsequently subjected to the appropriate disciplinary procedures indicated in the PROTOCOL FOR ACTION AGAINST FRAUDULENT PRACTICES AT THE UNIVERSITY OF VALENCIA (ACGUV 123/2020).

**Fraudulent collaboration and use of generative Artificial Intelligence (AI):**
All assignments and activities submitted and subject to assessment in this course must be original works by the student. Any work produced through fraudulent collaboration or with the help of Artificial Intelligence (ChatGPT or any other similar generative AI tool) will be considered plagiarism, except when its use is

expressly authorized in writing by the course instructor and is part of the assigned task.

## REFERENCES


- Apuntes de la asignatura
- [G. Beekman (2005)]. Introducción a la informática [Recurs electrònic]. Pearson. Prentice Hall 2005. Ingebook Ebooks.
- [H.M. Deitel, P.J. Deitel (2014)]. C++ How to Program (Prentice-Hall). Disponible en catálogo electrónico de la UV.
- [L. Joyanes (2006)]. Programación en C++: Algoritmos, estructuras de datos y objetos (MacGraw Hill). Disponible en catálogo electrónico de la UV.
- [L. Joyanes, I. Zahonero (2005)]. Programación en C: Metodología, algoritmos y estructuras de datos (MacGraw Hill). Disponible en catálogo electrónico de la UV.
- [H.M. Deitel, P.J. Deitel (2009)]. C++ Cómo programar. [Recurs electrònic]. Pearson. Prentice Hall. 2009. Ingebook Ebooks
- [L. Joyanes (2006)]. Programación en C++: Algoritmos, estructuras de datos y objetos [Recurs electrònic]. MacGraw Hill. 2006. Ingebook Ebooks