Universidad Zaragoza

Academic Year/course: 2021/22

29902 - Fundamentals of computing

Syllabus Information

Academic Year: 2021/22
Subject: 29902 - Fundamentals of computing
Faculty / School: 110 - Escuela de Ingeniería y Arquitectura
Degree: 435 - Bachelor's Degree in Chemical Engineering
ECTS: 6.0
Year: 1
Semester: 435-First semester o Second semester
107-Second semester
Subject Type: Basic Education
Module:

1. General information

1.1. Aims of the course

The course and its expected results respond to the following approaches and objectives:
The objectives of the course are fundamentally of two types:

1. Train the student so that they can propose solutions to a problem by creating simple programs. Therefore, its basic and core content is programming and, in particular, the specification of problems, the proposal of a range of solutions as possible alternative algorithms, the choice of the best solution based on experimentation or previous experiences, and the translation of these solutions into computer-executable programs in a general-purpose programming language.
2. That the student knows the constituent elements of a computer, understands its basic operation, and is able to search for information and apply programming and problem-solving knowledge using the available software tools and applications.

These approaches and objectives are aligned with some of the Sustainable Development Goals, SDGs, of the 2030 Agenda (https://www.un.org/sustainabledevelopment/en/) and certain specific goals, in such a way that the acquisition of the Learning outcomes of the subject provide training and competence to the student to contribute to a certain extent to their achievement:

- Goal 1: End poverty in all its forms everywhere.
  Target 1.4: By 2030, ensure that all men and women, in particular the poor and the vulnerable, have equal rights to economic resources, as well as access to basic services, ownership and control over land and other forms of property, inheritance, natural resources, appropriate new technology and financial services, including microfinance.

- Goal 8: Promote inclusive and sustainable economic growth, employment and decent work for all.
  Target 8.2: Achieve higher levels of economic productivity through diversification, technological upgrading and innovation, including through a focus on high-value added and labour-intensive sectors

- Goal 16: Promote just, peaceful and inclusive societies.
  Target 16.5: Substantially reduce corruption and bribery in all their forms.

1.2. Context and importance of this course in the degree

Informatics is a basic course taught in the first year of the degree. This particular temporary location allows students to apply in all the subjects of the degree the knowledge acquired in this course, most of which, to a greater or lesser extent, need to rely on computer tools to solve problems.

1.3. Recommendations to take this course

This course introduces the Engineering student in solving information treatment problems using a computer as a tool. The tool (the computer) is introduced from the beginning, both from a general perspective of use, and in particular aspects aimed

for solving specific problems. To take this course, the student must be willing to develop skills for solving problems using a computer, through **a continuous practical work of solving specific problems**, which it cannot be replaced by any other learning technique.

2. Learning goals

2.1. Competences

After passing the course, the student will be more competent to ...

Generic competences

C04 - Ability to solve problems and make decisions with initiative, creativity and critical reasoning.

C05 - Ability to apply information and communication technologies in Engineering.

Specific competences

C14 - Use adequately and efficiently computers, operating systems, programming environments, databases and computer programs with application in Engineering.

2.2. Learning goals

The student, to pass this course, must demonstrate the following results ...

- Ability to retrieve information (including browsers, search engines, and catalogs).
- Know the basic operation of computers, operating systems and databases and make simple programs on them.
- Ability to operate with computer equipment effectively, taking into account its logical and physical properties.
- Know and use with ease the tools and software applications available in the laboratories of the basic courses.
- Correct formulation of the problem from the proposed statement and identify the options for its resolution. Apply the appropriate resolution method and give evidences about the correction of the proposed solution.
- Ability to specify, design and build simple computer systems.

2.3. Importance of learning goals

This course deals with the concepts and skills that constitute the "way of thinking of the engineer", and that are experienced into practice with real problems from the beginning. With the focus in the resolution of problems, Informatics concerns the knowledge, design and exploitation of computing and computer technology, giving rise to a discipline that:

1. Develops the ability to express solutions to problems as algorithms, in areas of engineering as system design, problem solving, simulation, or modeling.
2. Deploys a disciplined approach to problem solving, from which quality solutions are expected.
3. Controls the complexity of the problems, first through abstraction and simplification processes, and then design solutions by integrating components.
4. Facilitates the understanding of the opportunities offered by the automation of processes, and how people interact with computers.
5. Facilitates the learning through experimentation, of basic principles such as conciseness and elegance, as well as to recognize bad practices.

3. Assessment (1st and 2nd call)

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

The student must demonstrate that they have achieved the expected learning outcomes through the following assessment activities

Laboratory practical work (30%). The ability to identify the information needs to solve the problems proposed and their use in their resolution (RA 1) will be evaluated. The critical capacity when selecting alternatives and the degree of justification of the solution reached (RA 5) will also be assessed. The fluency in the use of the computer to solve problems will be evaluated (RA 2 and RA 4). The solutions implemented for each of the exercises proposed for the practical sessions will also be evaluated, taking into account the quality of the procedures and efficient resolution strategies in the computer (RA 3), as well as the quality of the program that implements the strategy (RA 3).

Written exam (70%) in which questions and/or problems in the field of engineering to be solved by means of a computer will be asked. The typology and level of complexity will be similar to that used during the course. The quality and clarity of the resolution strategy (RA 5) will be assessed, as well as its efficiency (RA 2). The quality of the program, written in the general-purpose programming language used during the course (RA 6), which carries out the proposed strategy (RA 3) will also be assessed. Serious semantic errors - ignorance of the basic rules for the construction and coding of algorithms - could mean the total penalty of the exercise. To pass the course, a minimum grade of 4 out of 10 is required in this activity.

Organization of assessment activities

The student will pass the subject by completing the activities listed in the previous section.

The overall assessment will be divided into two parts. The first corresponds to the previous assessment activity numbered as 2 and with the weighting indicated there. The second corresponds to the previous assessment activity numbered as 1. The weight will be equal to the sum of the weights indicated there. The date of completion will be specified sufficiently in advance by the center in the periods for the examinations. The schedule will be defined by the teaching staff of the course well in advance.

Students will be able to pass Activity 1 during the course by taking and passing the tests that will be indicated in advance. Students who have not passed Activity 1 during the course must take the second part of the global assessment. Students who have passed it during the course may appear, if they wish, to upload a grade on the dates of the global assessment.

4. Methodology, learning tasks, syllabus and resources

4.1. Methodological overview

The learning process that is designed for this course is based on:

- The presentation of the contents of the course in lectures
- Analyzing and solving case studies in class.
- Personal study of the subject by students.
- The design and implementation of lab exercises by students, guided by teachers, in the computer laboratory.
- The development of simple programs of increasing difficulty proposed by the teachers as homeworks

Keep in mind that the course has both theoretical and practical orientation. Therefore, the learning process emphasizes both student attendance at lectures, as in the experiments in the laboratory, performing simple programs of increasing difficulty, and individualized study.

4.2. Learning tasks

The program is offered to the student in order to help him/her to achieve the intended learning outcomes and includes the following activities:

- In classes taught in the classroom, the program of the course will be developed.
- In classes of case studies, problems will be solved illustrating the concepts and techniques presented in the lectures
- In the laboratory sessions, problems of information processing will be solved designing and implementing programs running in a computer

4.3. Syllabus

The course program is organized into the following three blocks:

1. **Computer: A machine for the execution of algorithms.**
   The notion of Algorithm.
   Structure of the computer: Digital nature, coding, hardware, software.
   Operating systems.
   Databases.
   Programming: Programming Styles, Hierarchy of languages, Programming elements.
   Computer networks

2. **Abstraction with procedures.**
   Data types and algorithmic composition schemes: Data type concept.
   Constants and variables.
   Basic data types: Boolean, character, integer, real.
   Control structures, procedures and functions.
   Algorithm Design Techniques. Treatment of Sequences (sequential files and search). Recursion.

3. **Data abstraction.**
   Tables.
   Indexed access.
   Sorting algorithms as an example.
   Abstract Data Types: Modularity, objects and state.

Introduction to Object-Oriented Programming.
Introduction to techniques of object-oriented design.

The concepts, methods and tools of the above paragraphs are illustrated through examples, as realistic as possible, within the fields of chemical engineering, covering aspects such as: performing mathematical calculations, treatment of non-numerical information, simulation, etc.

4.4. Course planning and calendar

Scheduling of the sessions and presentation of works

The schedule of the course will be defined by the School in the academic calendar of the corresponding academic year.

4.5. Bibliography and recommended resources

http://biblos.unizar.es/br/br_citas.php?codigo=29902&year=2019