DEGREE CURRICULUM

INFORMATICS BASICS

Coordination: DIAZ LLOBET, MANEL

Academic year 2023-24

INFORMATICS BASICS 2023-24

Subject's general information

<table>
<tr><td>Subject name</td><td>INFORMATICS BASICS</td></tr>
<tr><td>Code</td><td>102329</td></tr>
<tr><td>Semester</td><td>2nd Q(SEMESTER) CONTINUED EVALUATION</td></tr>
<tr><td>Typology</td><td>
<table>
<tr><th>Degree</th><th>Course</th><th>Character</th><th>Modality</th></tr>
<tr><td>Bachelor's degree in Industrial Organization and Logistics Engineering</td><td>1</td><td>COMMON/CORE</td><td>Attendance-based</td></tr>
<tr><td>Common branch in industrial engineering programs - Igualada</td><td>1</td><td>COMMON/CORE</td><td>Attendance-based</td></tr>
<tr><td>Doble titulació: Grau en Enginyeria en Organització Industrial i Logística i Grau en Administració i Direcció d'Empreses</td><td>1</td><td>COMMON/CORE</td><td>Attendance-based</td></tr>
<tr><td>Not informed</td><td>1</td><td>COMMON/CORE</td><td>Attendance-based</td></tr>
</table>
</td></tr>
<tr><td>Course number of credits (ECTS)</td><td>6</td></tr>
<tr><td>Type of activity, credits, and groups</td><td>
<table>
<tr><th>Activity type</th><th>PRALAB</th><th>TEORIA</th></tr>
<tr><td>Number of credits</td><td>3</td><td>3</td></tr>
<tr><td>Number of groups</td><td>2</td><td>1</td></tr>
</table>
</td></tr>
<tr><td>Coordination</td><td>DIAZ LLOBET, MANEL</td></tr>
<tr><td>Department</td><td>COMPUTER ENGINEERING AND DIGITAL DESIGN</td></tr>
<tr><td>Teaching load distribution between lectures and independent student work</td><td>6 ECTS = 60 hours of class in person/virtual + 90 hours of autonomous work.</td></tr>
<tr><td>Important information on data processing</td><td>Consult this link for more information.</td></tr>
<tr><td>Language</td><td>Catalan.</td></tr>
<tr><td>Distribution of credits</td><td>Theoretical credits: 3
Practical credits: 3</td></tr>
</table>

INFORMATICS BASICS 2023-24

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
DIAZ LLOBET, MANEL	manel.diazllobet@udl.cat	9	

Subject's extra information

Mostly practical subject in which the study is based on the resolution of programming exercises. The individual work is fundamental to obtain the established competences and acquire the necessary skills to use the computer tool with which we will work during the course correctly. It is not necessary to have previous programming knowledge.

You can find collections of the following didactic materials on the Virtual Campus:

- Notes of Fundamentals of computer science.
- Programming in a numerical computing environment.
- Collections of problems, examples, solutions and complementary materials.

The use of the Virtual Campus is fundamental to access the resources of the subject, notifications about the dates of delivery of exercises, agenda of sessions and finally the delivery of practices and tests of evaluation.

Learning objectives

The instrumental nature of the computer science subject makes it very useful in a large number of professional areas and areas. Computing provides a wide range of specific solutions for professional development in the field of Industrial Engineering, but also plays an essential role in the area of Process Engineering and the programming of control systems and automation. This last field of study is where this subject intends to introduce the student. The main objective is that the student learns to design and implement in a programmable device efficient and quality solutions to different types of problems raised. In addition to the programming environment and the resolution techniques, it is intended to introduce students into the basic components of a programmable device with which the student will have to interact. Therefore, the student will acquire a conceptual knowledge as practical about how to write a program, treat and process the data.

This general objective can be divided into the following more specific objectives:

1. Identify the basic components of a programmable device, its functionality and the process of interaction with the other components.
2. Learn and understand the syntax and semantics of a high level language.
3. Use a high-level language to solve complex problems efficiently.
4. Use the basic programming structures of a high level programming language correctly: conditionals and iterators.
5. Properly use the downward design to solve complex problems.
6. Evaluate and validate the quality of the solution based on the results obtained.
7. Acquire practice and dexterity in the resolution of real calculation problems in interpreted programming environments.

Competences

Basic Competences

- **B01.** That students have demonstrated to possess and understand knowledge in an area of study that starts from the base of general secondary education, and is usually found at a level that, although supported by advanced textbooks, also includes some aspects that imply knowledge coming from the vanguard of his/her field of study.
- **B02.** That students know how to apply their knowledge to their work or vocation in a professional manner and possess the skills that are usually demonstrated through the elaboration and defense of arguments and the resolution of problems within their area of study.
- **B03.** That students have the ability to gather and interpret relevant data (usually within their area of study) to make judgments that include a reflection on relevant social, scientific or ethical issues.

Transversal Competences

- **CT2.** To develop meaningful command of a foreign language, especially English.
- **CT3.** To implement new technologies and technologies of information and communication.
- **CT5.** To apply essential notions of scientific thinking.
- **CT6.** Apply the gender perspective to the tasks of the professional field.

General Competences

- **CG3.** To synthesize basic and technological subjects, which enable them to learn new methods and theories, and provide them with versatility to adapt to new situations.
- **CG4.** To solve problems with initiative, make decisions, creativity, critical reasoning and to communicate and transmit knowledge, skills and abilities in the field of Industrial Chemical Engineering / Industrial Organization Engineering.
- **CG10.** To work in a multilingual and multidisciplinary environment.

Specific Competences

- **CE3.** To acquire fundamental knowledge of the use and programming of computers, operating systems, databases and computer programs with applications in engineering.

Subject contents

1. Preliminary Concepts.
    1. Von Neumann architecture.
    2. Current evolution and state of computational systems.
    3. Operating Systems: Functions, current OS, Virtual Machines.
    4. Representation of the information.
2. Introduction to the resolution of engineering problems.
    1. Computational thinking
    2. Algorithmic thinking
3. Programming and algorithms.
    1. First steps in numerical computing environments.
    2. Programming in numerical computing environments.
        1. Scripts
        2. Operators and flow control.
        3. User functions and descending design.
        4. Input / Output functions.
    3. Vectors and arrays.
    4. Access to files and databases.
    5. Toolboxes and other advanced tools in numerical computing environments.

Methodology

The classroom activities are divided into two parts that are complemented: master classes and laboratory.

- Master classes: Theoretical concepts are introduced and activities and problems that require the application of the theoretical concepts for their resolution are presented. Different solutions are discussed analyzing the results and their effectiveness.
- Laboratory classes: Laboratory classes are taught in small groups of students, thus favoring the dialogue and participation of them. A series of problems or activities of gradual difficulty are posed. Students seek and implement an algorithmic solution to the problems encountered. The best techniques to solve the problems are discussed individually and / or group. Each practical session gradually includes the concepts seen in the keynote sessions.

Development plan

Week	Description	Activity Theory	Activity PRALAB
1	Introduction + T1	Introduction + T1	Introduction
2	T1	Theory	LAB01
3	T2	Theory and problems	LAB02
4	T2	Theory and problems	LAB03
5	T3	Theory and problems	LAB04
6	T3	Theory and problems	LAB05
7	T3	Theory and problems	LAB06
8	Doubts and Review	Non School Day	Doubts and Review
9	PA1		
10	Octave	Theory and problems	Laboratory: Practice 1
11	Octave	Theory and problems	Laboratory: Practice 1
12	Octave	Theory and problems	Non School Day

Week	Description	Activity Theory	Activity PRALAB
13	Octave	Theory and problems	Laboratory: Practice 2
14	Octave	Theory and problems	Laboratory: Practice 2
15	Octave	Theory and problems	Laboratory: Practice 2
16	PA2		
17	PA2		
18	PA2		
19	Evaluation and tutoring		
20	Recovery exam		

Evaluation

Evaluation block	% Final grade	Evaluation activities	Grade %	Minimum note	In group	Compulsory	Recoverable
Lab test (LAB)	**40%**	Lab test 1 (LAB1)	20%	4	YES(<=2)	NO	YES
		Lab test 2 (LAB2)	20%	4	YES(<=2)	NO	
Theory (THE)	**35%**	Partial Exam 1 (PE1)	20%	-	NO	NO	YES
		Partial Exam 2 (PE2)	15%	-	NO	NO	
Participation (AP)	**25%**	Autonomous work and participation (AP)	25%	-	NO	NO	NO

Evaluation grade = Weighted average of the evaluation blocks.
Final grade = Evaluation grade
To pass the subject, the Final Grade must be >= 5
LAB activities score 100% in the case of delivering the activity during the session
LAB activities score 80% if the activity is delivered until 11:55 p.m. on the same day.
LAB activities do not score if they are delivered after the same day of the activity.
Recovery:

- Students will be able to retrieve the grade of the subject corresponding to PE1 and PE2 following the guidelines of the EPS Academic Framework.
- Recovery will be done through a written exam of the total content of the subject, with a weight of 40% of the final mark.
- Additionally, LAB1 and LAB2 activities can be recovered through a practical exam, with a weight of 35% of the final mark.
- Only students that have failed the subject can apply for the recovery exam.

Final Grade = 0,40 * LAB + 0,35 * THE + 0,25 AP

ALTERNATE EVALUATION:

Students who have consent to be evaluated through the alternative evaluation (see requirements and procedure in the evaluation regulations) will have to carry out the following activity:

- Practical test at the end of the subject, which will include the contents of LAB1 and LAB2. The test will be face-to-face.

Bibliography

- John W. Eaton, David Bateman, Søren Hauberg, Rik Wehbring. GNU Octave: A high-level interactive language for numerical computations (https://octave.org/octave.pdf)
- Knut-Andreas Lie. An Introduction to Reservoir Simulation Using MATLAB/GNU Octave: User Guide for the MATLAB Reservoir Simulation Toolbox (MRST)
- Alfio Quarteroni, Fausto Saleri. Calculo cientifico con Matlab y Octave. Ediz. italiana e spagnola (Unitext: La Matematica Per il 3+2)