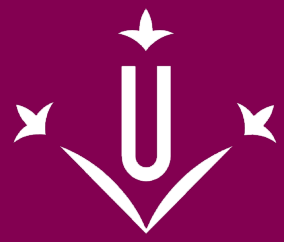

Universitat de Lleida

DEGREE CURRICULUM

COMPUTER SCIENCE BASICS

Coordination: LERIDA MONSO, JOSEP LLUIS

Academic year 2022-23

Subject's general information

Subject name	COMPUTER SCIENCE BASICS
Code	102109
Semester	2nd Q(SEMESTER) CONTINUED EVALUATION
Typology	

Degree	Course	Character	Modality
Bachelor's Degree in Automation and Industrial Electronic Engineering	1	COMMON/CORE	Attendance-based
Bachelor's Degree in Energy and Sustainability Engineering	1	COMMON/CORE	Attendance-based
Bachelor's Degree in Mechanical Engineering	1	COMMON/CORE	Attendance-based
Common branch in industrial engineering programs - Lleida	1	COMMON/CORE	Attendance-based
Double bachelor's degree: Degree in Mechanical Engineering and Degree in Energy and Sustainability Engineering	1	COMMON/CORE	Attendance-based

Course number of credits (ECTS)	6
Type of activity, credits, and groups	

Activity type	PRALAB	TEORIA
Number of credits	3	3
Number of groups	5	3

Coordination	LERIDA MONSO, JOSEP LLUIS
Department	COMPUTER SCIENCE AND INDUSTRIAL ENGINEERING
Teaching load distribution between lectures and independent student work	40% Classroom 60% Autonomous Work
Important information on data processing	Consult this link for more information.
Language	Catalan

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
FLIX ROVIRA, JOSE MARIA	josepmaria.flix@udl.cat	3	
LERIDA MONSO, JOSEP LLUIS	joseplluis.lerida@udl.cat	12	
PARDO CASANOVAS, VALENTI	valenti.pardo@udl.cat	3	
SOLA GIMENO, JOSEP MARIA	josepmaria.sola@udl.cat	6	

Subject's extra information

This is a mainly practical course in which the study is based on the resolution of recommended exercises. Individual work is essential for acquisition of the skills and the correctly use computer tools that will be used during the course. No previous programming knowledge required.

You can find the following collections of teaching materials in the Virtual Campus: http://cv.udl.cat

- Introduction to the Computer Basics
- MATLAB Programing
- Collections of problems, Examples, Solutions and extra materials

The use of the Virtual Campus is essential to access the resources of the course, notifications, deadlines information, teaching plan and assessment.

Learning objectives

The instrumental nature of the computer programming course make it very useful in many areas and professional fields. The computer programming provides a variety of specific solutions for the professional development in the field of Industrial Engineering, but also plays an essential role in the area of process engineering, control systems and automation. The main objective for this course is that students learn to design and implement on a programmable device, efficient and quality solutions to different types of problems. In addition to manage the programming environment and the solving techniques, this course introduce the students to the basic components of a programmable device as well as to acquire the conception and practice of how to write a program, compile and run knowledge.

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

1. Identify the basic components of a computer or 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 correctly the basic structures of high-level programming language: conditionals and iterators.
4. Use properly a top-down design to address complex problems.
5. Evaluate and validate the quality of the solution according to the results.

6. Acquire practical skills in solving real problems in interpreted environments.

Competences

University of Lleida strategic competences

- UdL3. Mastering ICT's.

Degree-specific competences

- EPS5. Capacity of abstraction and of critical, logical and mathematical thinking.

Degree-transversal competences

- GEM3 i GEEIA3. Basic knowledge of the use and programming of computers, operating systems, databases and computer programs with applications in engineering.

Subject contents

1 Preliminaries

- 1.1 Von Neumann Architecture
- 1.2 Evolution and Current status of programmable systems
- 1.3 Operating Systems: Functions, current OSs, Virtualization
- 1.4 Information representation

2 Introduction to Problem Solving for Engineering

- 2.1 Computational Thinking
- 2.2 Algorithmic Thinking
- 2.3 Algorithmic complexity

3 Programming and Algorithms

- 3.1 Getting started. Interface and basic syntax in MATLAB.
- 3.2 Introduction to programming in MATLAB
  - 3.2.1 M-File Scripts
  - 3.2.2 Control Flow and operators
  - 3.2.3 User-defined Functions. Top-down algorithm design
  - 3.2.4 Input/Ouput Functions
- 3.3 Vectors and Matrices
- 3.4 Access to Files and Databases in Matlab
- 3.5 Toolboxes and Advanced tools in MATLAB

4 Solving cases of Study

Methodology

During a 100% face-to-face modality the sessions will consist in:

- Lectures: theoretical concepts are introduced and activities and raise issues that require the application of theoretical concepts for its resolution. We discuss different solutions, analyzing the results, the efficiency and the effectiveness.
- Laboratory: Laboratory classes are taught in small groups of students, thus promoting dialogue and participation from them. A number of problems or activites that require a solution programmed with gradual difficulties are posed. Students seek and implement an algorithmic solution to the real problems posed. It is discussed individually and / or in groups the best techniques to solve the problems. Each practice session gradually add the concepts seen in the lectures.

During a semipresential modality the sessions will consist in:

- Lectures: theoretical concepts are introduced and activities and raise issues that require the application of theoretical concepts for its resolution. We discuss different solutions, analyzing the results and efficiency.
- Laboratory: Laboratory classes are taught in small groups of students, thus promoting dialogue and participation from them. A number of problems or activites that require a solution programmed with gradual difficulties are posed. Students seek and implement an algorithmic solution to the real problems posed. It is discussed individually and / or in groups the best techniques to solve the problems. Each practice session gradually add the concepts seen in the lectures.

During a non-attendance modality the sessions will consist in:

- 2 hours per week in videoconference format to explain content. The contents taught in these sessions are included in the transparencies that are made available to students through the CV prior to the session. These sessions are complemented with the presentation of practical examples using MATLAB tool and share the screen through the video conferencing tool. Tools such as the Blacboard or One Note will also be used to make written notes on or off transparencies.
- 2 hours per week of doubts and problems. Each week the student is asked to solve a set of exercises related to the contents taught in the last content session. In addition, each week the student is asked to solve a case study. During the doubts sessions the student shares the code of the problems that can not solve solve or the doubts she/he has. Based on the material provided by students the professor provide the solution to the doubts, errors in the code, clarify poorly acquired concepts, etc., the optimality of the solutions is also discussed and alternatives are proposed when necessary. In these sessions, the student can also provide doubts about the proposed case studies.

In parallel to the previous synchronous sessions, the student is encouraged to use different tools and carry out different actions to keep up to date with the subject.

- Use of the Forum tool. For each topic and case study, is created a topic on the CV's Forum tool. In this space, the student can raise doubts regarding the contents seen in the content sessions or about the problems available to students on the problems notebook from the beginning of the course. Students express their doubts and all students are encouraged to resolve the doubts of their peers. Teachers participate to clarify or resolve any doubt that have no answer from students.
- Student solutions folder. The students can upload the problems solved from the problem notebook. The solucitions can be uploaded on the 04_Problems_and_Solutions folder in the Resources section and all these solutions are available for the rest of the classroom. This action is very useful, because it provides students with a source of resources useful to resolve doubts, learn alternative solutions, error detection, self-assessment, review and comparison of different solutions, and so on.
- Solution to the case studies can be uploaded to the virtual campus with a deadline. The deadline for these activities marks the pace of the course. The realization of these activities favors the putting in practice of the knowledges and the self-evaluation of the learning process.

Student participation on the described activities will be taken into account on the student assessment.

Development plan

Dates (Weeks)	Description	Face-to-Face Activity	FH (2) (Hours)	Autonomous Activity	AH (3) (Hours)
Week 1 (07 Feb)	Course Presentation	Exhibition events and methodology	1	Academic guide review	1
	T1.1 Von Neumann Architecture, memory and processor	Lecture and participatory classes	1	Study	1
	T1.4 Information representation	Lecture and participatory classes	2	Study	2
	Resolution of doubts through the virtual forums			Forum review	1
Week 2 (14 Feb)	T1.2 Evolution and Current status of programmable systems	Lecture and participatory classes	2	Study	2
	T1.4 Representation of Integer numbers	Lecture and participatory classes	2	Study	1
	Resolution of doubts through the virtual forums			Forum review	1
Week 3 (21 Feb)	T1.3 Operating Systems	Lecture and participatory classes	2	Study	2
	T1.4 Representation of Alphanumeric data. Problems	Lecture and Problem Based Learning	2	Problems and study	3
	Resolution of doubts through the virtual forums			Forum review	1
Week 4 (28 Feb)	T2. Computational Thinking	Lecture and participatory classes	2	Study	2
	Lab. Getting started	Laboratory	2	Practical Activity, Software installation and Introduction	2 3
	Resolution of doubts through the virtual forums			Forum review	1
Week 5 (07 Mar)	T2. Algorithmic Thinking	Lecture and participatory classes	2	Study	2
	Lab. Built-in functions and graphical representation	Laboratory	2	Practical Activity	3
	Resolution of doubts through the virtual forums			Forum review	1
Week 6 (14 Mar)	T3. Control Flow and operators	Lecture, participatory classes and exercises	2	Exercises	4
	Lab. Scripts i Entrada/Sortida	Laboratory	2	Practical Activity (LAB03)	3
	Resolution of doubts through the virtual forums			Forum review	1
Week 7 (21 Mar)	T3. Control Flow and operators Problems	Lecture, participatory classes and exercises	2	Practical Activity (LAB04)	2
	Lab. (Lab04)			Practical Exercise	4
	Resolution of doubts through the virtual forums			Forum review	2
Week 8 (28 Mar)	T3. Iterative sentences and Problems	Lecture, participatory classes and exercises	2	Exercises	2
	T3. Problems and LAB04	Problem Based Learning	2	Problem Solving	4

	Resolution of doubts through the virtual forums			Forum review	2
Week 9 (04 Abr)	**PA1. Evaluation Test - 1**	Individual written exam	2		
Week 10 (18 Abr)	T3. User Functions (S6)	Lecture, participatory classes and exercises	2	Exercises	2
	Lab. (Lab05)	Laboratory	2	Practical Exercise	4
	Resolution of doubts through the virtual forums			Forum review	2
Week 11 (25 Abr)	T3. **Vector** (S7)	Lecture, participatory classes and exercises	2	Problem Solving	2
	Lab. (Lab06)			Practical Exercise	2
	Resolution of doubts through the virtual forums			Forum review	2
Week 12 (02 Mai)	**PLAB 1. Avaluable Practice**	Case of study	2		2
	T3. Problems and LAB06	Probem solving	2		2
Week 13 (09 Mai)	T3. Vector and **Matrices** (S8)	Lecture, participatory classes and exercises	2	Problem Solving	2
	Lab. (LAB07)	Laboratory	2	Practical Exercise	2
	Resolution of doubts through the virtual forums			Forum review	2
Week 14 (16 Mai)	LAB 07 solutions	Laboratory	2	Problem Solving	3
	Problems with Matrices	Laboratory	2	Practical Exercise	6
	Resolution of doubts through the virtual forums			Forum review	2
Week 15 (23 Mai)	**PLAB 2. Evaluation Practice**	Case of study	2		
	Lab. (LAB08)	Laboratory	2	Practical Exercise	3
	Resolution of doubts through the virtual forums			Forum review	2
Week 16 Week 17	**PA2. Evaluation Test - 2**	Individual written exam	2		
Week 18	Qualifications and Tutoring session				90
Week 19	Recovery Test	Individual written exam	2		

Evaluation

Objectives	Evaluation activities	%	Dates	O/V (1)	I/G (2)	Remarks
Continous assesment Topics 2 & 3	Evaluable practice PLAB1	15	Week 9	O	I	
Continous assesment Topics 2 & 3	Evaluable practice PLAB2	15	Week 15	O	I	

Objectives	Evaluation activities	%	Dates	O/V (1)	I/G (2)	Remarks
Continous assesment Topics 1 & 2 & 3	Autonomous work and classroom participation. AP	10	During all the course	O	I	
Continous assesment Topics 1 & 2 & 3	PA1. Writting exam	25	Week 9	O	I	
Continous assesment Topics 2 & 3	PA2. Writting exam	35	Week 16 i 17	O	I	
Recovery assesment. Topics 1 & 2 & 3	Recovery assesment. Writting exam (*)	60	Week 19	V	I	Recoveries Week

(1) Mandatory / Voluntary
(2) Individual / Group

Final Marks:
During the weeks 9th and 16th / 17th will be performed the writting exams: **PA1** and **PA2**. Each of these tests has a weight of 25% and 35% over the final mark, respectively.

The final grade is calculated by the sum of the results of the writting test (**PA1, PA2**) plus the classroom practices (**PLAB1, PLAB2**) and autonomous work with classroom participation (**AP**): **Final Mark = PA1 + PA2 + PLAB1 + PLAB2 + AP**

The Participation is calculated considering the active participation of the student in ALL the following activities: Forum participation, Delivery of solutions through the Virtual campus or at classroom, delivery of LABS, results of the Tests, etc.. It is necessary an active participation in ALL these activities.

(*) On the 19th week the grade of the course can be recovered with a final writting exam, following the guidelines of the EPS Academic Degrees Framework. The recovery will be done through a written exam with all contents of the course with a weight of 60% of the final grade. The evaluable practices can not be recovered. This exam can also be done by all those students having passed the ongoing assessment and want to modify their grade, assuming that the obtained score in the recovery test is the only score considered to

Bibliography

Basic bibliography:

Matlab. A Practical Introduction to Programming and Problem Solving. Stormy Attaway. Butterworth-Heinemann; Fourth edition. 2016. (Electronic resource - accesible from UdL)

Essential MATLAB for Engineers and Scientists. Brian Hahn, Daniel Valentine. Academic Press; Sixth edition. 2017. (Electronic resource - accesible from UdL)

Breves apuntes de MATLAB. Una introducción rápida pero no trivial. Rosa Echevarria. Universidad de Sevilla. (http://personal.us.es/echevarria/documentos/IntroduccionMATLAB.pdf)