Universitat de Lleida

DEGREE CURRICULUM

COMPUTER ARCHITECTURE

Coordination: VILAPLANA MAYORAL, JORDI

Academic year 2018-19

Subject's general information

<table>
<tr><td>Subject name</td><td>COMPUTER ARCHITECTURE</td></tr>
<tr><td>Code</td><td>102369</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 Digital Interaction and Computing Techniques</td><td>1</td><td>COMMON</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>PRAULA</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>VILAPLANA MAYORAL, JORDI</td></tr>
<tr><td>Department</td><td>COMPUTER SCIENCE AND INDUSTRIAL ENGINEERING</td></tr>
<tr><td>Teaching load distribution between lectures and independent student work</td><td>Overall, the subject has 60 class hours and 120 hours of individual student 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>The distribution of credits counting on the group's deployment that is done in the subject is the following:
Concepció Roig: 1.5
Jordi Vilaplana: 7.5</td></tr>
</table>

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
TOMÀS GLEYAL, MARC	mtomas@diei.udl.cat	3	
VILAPLANA MAYORAL, JORDI	jordi@diei.udl.cat	6	

Subject's extra information

Subject taught during the second term of the first year of the degree.
It is a compulsory subject.
In order to attend the subject, the knowledge of the functional units of the computer system that are given to the previous Computer Structure course is required.

Learning objectives

- Study the global operation of the computer memory hierarchy and the different levels that make it up.
- Learn the organization of information in the memory system so that access is efficient.
- Be able to propose a basic structure for the main memory of a computer.
- Identify and understand the input and output system within the structure of a computer.
- Study the processes and algorithms that must be carried out to perform the basic and complex arithmetic operations within the arithmetic unit.
- Analyze the different solutions that are presented from the point of view of efficiency and cost. Know how to assess which design solutions achieve a better cost-performance commitment.

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.

Transversal Competences

- **CT3.** Acquire training in the use of new technologies and information and communication technologies.
- **CT5.** Acquire essential notions of scientific thought.

General Competences

- **CG2.** Design, develop, evaluate and guarantee the accessibility, ergonomics, usability and security of computer systems.
- **CG3.** Use adequate hardware and software platforms to develop and execute interactive digital applications.
- **CG5.** Know the basic subject areas and technologies needed to learn and develop new methods and technologies, and those that help to adapt to new situations.

- **CG7.** Solve problems through initiative, determination, independence and creativity.
- **CG8.** Capacity for abstraction and critical, logical and mathematical reasoning.

Specific Competences

- **CE2.** Capacity to understand and master the basic concepts of discrete mathematics, logics, algorithmic and computational complexity, and its application to solve computational problems.
- **CE3.** Basic knowledge of the use and programming of computers, operating systems and databases, and their use in the development of interactive applications.
- **CE4.** Capacity to know, understand and evaluate the structure and architecture of computers, as well as the basic components that conform them.
- **CE16.** Capacity to design and evaluate person-computer interfaces that guarantee the usability of systems, services and computer applications.
- **CE17.** Capacity to apply knowledge on design to propose and defend a design concept for an interactive system and use proper creative technologies to develop each project.
- **CE24.** Capacity to understand the human factors involved in any interactive process between humans and technology, as well as being able to adequately apply them in the design of interactive products and services, and their interfaces.

Subject contents

1. Memory hierarchy
2. Input / output unit
3. Arithmetic-logic unit
4. Advanced computer architecture concepts

Methodology

Big Groups: Theory Classes (3 ECTS)

- These classes will explain the theoretical contents of the subject, accompanied by illustrative examples. As material of support of the class the transparencies of the subject will be followed.

Small Groups: Problems / Laboratory Classes (3 ECTS)

- In these classes the resolution of the collection of problems associated with the theoretical explanations of the subject will be alternated, along with the realization of directed practices. The material of the practices will be uploaded to the Virtual Campus. The student must attend the class of practices with the previously read statements.

Autonomous Work (non-presential)

- It is recommended that the student solves the problems not resolved in the class of the problem collection in order to practice and obtain feedback from the teacher.

Development plan

Week	Description	BG Activity	SG Activity
1	Presentation + T1: Memory hierarchy	Presentation + T1	Theory
2	T1: Memory hierarchy	Theory	Theory
3	T1: Memory hierarchy	Theory	Problems
4	T1: Memory hierarchy	Theory	Problems

Week	Description	BG Activity	SG Activity
5	T2: Input / output unit	Theory	Laboratory: Practice 1
6	T2: Input / output unit	Theory	Problems
7	T2: Input / output unit	Theory	Laboratori: Practice 1
8	T2: Input / output unit	Theory and problems	Problems
9	Partial exam		
10	T3: Arithmetic-logic unit	Theory	Problems
11	T3: Arithmetic-logic unit	Theory	Laboratory: Practice 2
12	T3: Arithmetic-logic unit	Theory and problems	Laboratory: Practice 2
13	T3: Arithmetic-logic unit	Theory	Problems
14	T4: Advanced concepts	Theory	Problems
15	T4: Advanced concepts	Theory	Problems
16	Partial Exam 2		
17	Partial Exam 2		
18			
19	Recovery Exam		

Evaluation

Acr.	Evaluation activities	Weighting	Minimum grade	In group	Compulsory	Recoverable
E1	1st partial exam	35%	No	No	Sí	Sí
E2	2nd partial exam	45%	No	No	Sí	Sí
PRA	Practices	20%	No	Sí (<= 2)	Sí	No

Final grade = 35% * E1 + 45% * E2 + 20% * PRA
To pass the subject, the final grade must be greater than or equal to 5.

If you have not passed the subject, you can go to the recovery exam. In this case the note will be calculated as follows:
Final grade = 80% * recovery exam + 20% * PRA

Bibliography

- Stallings W., Organización y arquitectura de computadores. (10th edition) Prentice-Hall.
- Hamacher C., Vranesic Z., Zaky S. Organización de computadores (5th edition). McGraw-Hill.
- Ortega J., Anguita M., Prieto A. Arquitectura de computadores. Thomson.
- Hennessy J. L., Patterson D. A. Computer Architecture. A Quantitative Approach. Morgan Kaufmann.
- Subject Notes. J. Vilaplana. Resources section of the Virtual Campus.