



## DEGREE CURRICULUM

# COMPUTER ARCHITECTURE

Coordination: ROIG MATEU, CONCEPCION

Academic year 2022-23

## Subject's general information

Subject name	COMPUTER ARCHITECTURE			
Code	102014			
Semester	2nd Q(SEMESTER) CONTINUED EVALUATION			
Typology	Degree	Course	Character	Modality
	Bachelor's Degree in Computer Engineering	2	COMPULSORY	Attendance-based
	Double bachelor's degree: Degree in Computer Engineering and Degree in Business Administration and Management	2	COMPULSORY	Attendance-based
	Master's Degree in Informatics Engineering		COMPLEMENTARY TRAINING	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	3		2
Coordination	ROIG MATEU, CONCEPCION			
Department	COMPUTER SCIENCE AND INDUSTRIAL ENGINEERING			
Teaching load distribution between lectures and independent student work	Globally, the subject has 60 hours of presential classes and 120 hours of independent student work.			
Important information on data processing	Consult <a href="#">this link</a> for more information.			
Language	Catalan			

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
BIGORDA SOLDEVILA, TOMAS	tomas.bigorda@udl.cat	6	
ROIG MATEU, CONCEPCION	concepcio.roig@udl.cat	6	
TOMAS CUÑAT, ROSA ANA	rosana.tomas@udl.cat	3	

## Subject's extra information

This subject is held during the second semester of the second course of the degree.

This is a compulsory subject.

To follow up the subject it is required to have the knowledge of functional units composing the computer system, that are studied in the previous subjects of Computer Organization I and II.

## Learning objectives

- Studying the global operation and the levels of the memory hierarchy in the computer.
- Learning the organization of the information in the memory system in order to have efficiency in the acces.
- Learning the pipeline mechanism to execute instructions inside the processor and evaluate its performance.
- Study the processes and the algorithms that are needed to carry out basic and complex arithmetic operations inside the arithmetic unit.
- Analyzing different solutions in terms of efficiency and cost. Being able to find which design solutions provide the best tradeoff between cost and performance.

## Competences

### Degree-specific competences

- GII-FB5: Knowledge of the structure, organization, workings and inter-connexion of computer systems, the basis of their programming, and their applications in the resolution of engineering problems.
- GII-CRI9: Ability to know, understand and evaluate computer structures and architecture, as well as the basic components which constitute them.

### Degree-transversal competences

- EPS5: Ability for abstraction and critical, logical and logical reasoning.

## Subject contents

### A. THEORETICAL CONTENTS

## 1. Memory hierarchy

### 1.1. Introduction

General concepts

Principle of locality

### 1.2. *Cache memory*

Cache memory configurations

Mapping and identification of blocs

Bloc replacing algorithms

Writing policies

Data consistency

Cache performance

### 1.3. Main memory

Organization for improving performace

Alternative technologies

### 1.4. Virtual memory

Elements of virtual memory

Page table

TLB (Transaction Look-aside Buffer).

## 2. Pipeline processing

### 2.1. Basic concepts

### 2.2. Hazard management

Estructural hazards

Data hazards

Control hazards

### 2.3. Influency of instruction set

### 2.4. Superscalar execution

## 3. Arithmetic processing

### 3.1. Adder circuits.

Half-adder, full-adder, parallel adder.

Carry-look-ahead.

adder/subtractor circuit.

### 3.2. Binary multiplication algorithms.

### 3.3. Binary division algorithms

### 3.4. Floating point arithmetic

Floating point format

Approximate representation: rank and precision

Add and subtract operations

Multiplication and division operations

## B. PRACTICAL CONTENTS

Simulation of the functional units of the system.

- Practice of memory hierarchy with SMPcaché simulator. (Theme 1)
- Practice of pipeline processing with WinMIPS64 simulator. (Theme 2)

## Methodology

Classes are taught in face-to-face mode and are divided in different groups: theoretical group (Teo group) and problems/practices group (PraLab group).

The contents of the different kind of groups are divided in the following way:

- Theoretical group: They are expositive lessons where they are shown the main contents on the subject.
- PraLab group: they are lessons to solve exercises and practices, related to the contents exposed in the Teo group classes, in a participative and interactive way.

## Development plan

Week	description	Activity Grup Teoria	Activity Grup PraLab
1	Memory hierarchy	Presentation of the subject. General concepts. Principle of locality.	Exercises of memory hierarchy
2	Memory hierarchy	Cache memory configurations. Mapping and identification of blocs	Exercises of memory hierarchy
3	Memory hierarchy	Bloc replacing algorithms. Writing policies	Exercises of memory hierarchy
4	Memory hierarchy	Data consistency	Exercises of memory hierarchy
5	Memory hierarchy	Cache performance	Practices of memory hierarchy
6	Memory hierarchy	Main memory. Virtual memory.	Practices of memory hierarchy
7	Pipeline processing	Basic concepts	Exercises of pipeline processing
8	Pipeline processing	Hazard management	Exercises of pipeline processing
9	Partial evaluation activities.	Realization of first partial exam	
10	Pipeline processing	Hazard management	Practices of pipeline processing
11	Pipeline processing	Influence of instruction set. Superscalar execution	Practices of pipeline processing
12	Arithmetic processing	Adder circuits	Exercises of arithmetic processing.
13	Arithmetic processing	Binary multiplication algorithms	Exercises of arithmetic processing.

14	Arithmetic processing	Binary division algorithms	Exercises of arithmetic processing.
15	Arithmetic processing	Floating point arithmetic	Exercises of arithmetic processing.
16 i 17	Partial evaluation activities	Realization of second partial exam	
18	Seminars		
19	Recuperation evaluation activities	Realization of the recuperation exam, if needed.	

## Evaluation

Acr.	Evaluation activity	Weighing	Minimum mark	In group	Compulsory	Recoverable
P1	Exam 1 <sup>er</sup> Partial	30%	NO	NO	NO	YES
P2	Exam 2 <sup>on</sup> Partial	50%	NO	NO	NO	YES
PRA	Practices	20%	NO	YES (<=2)	NO	NO
<b>FINAL_MARK=30% P1+50% P2+20% PRA</b> To pass the subject, it is necessary that FINAL_MARK is greater than or equal to 5.						
In the case of not passing the subject there is the option of recuperating it separately for each of the two parts. In this case, the mark is calculated as following: N_rec_P1: recuperation mark of first partial exam (or P1 if not presented to the recuperation exam) N_rec_P2: recuperation mark of second partial exam (or P2 if not presented to the recuperation exam) <b>FINAL_MARK = 30% N_rec_P1 + 50% N_rec_P2 + 20% PRA</b>						

## Bibliography

Stallings W., *Organización y arquitectura de computadores*. (7 edición) Prentice-Hall.

Hamacher C., Vranesic Z., Zaky S. *Organización de computadores* (5ªedición). 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.