Computer Structure

Academic Year: (2019 / 2020) **Review date: 29-04-2019**

Department assigned to the subject: Computer Science and Engineering Department

Coordinating teacher: GARCIA CARBALLEIRA, FELIX

Type: Compulsory ECTS Credits : 6.0

Year : 2 Semester : 1

REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

Programming
Computer technology

OBJECTIVES

The main objective of the course is to describe the main components of a computer and the basic behaviour of a computer.

1. Generic competences:
- Capacity of analysis and synthesis (PO a, c, e)
- Capacity of organization and scheduling (PO c, d)
- Problem solving (PO c, g)
- Capacity to apply theoretical concepts (PO a, c)
- Capacity to apply their knowledge to their work in a professional manner and have competences to solve problems within their field of study (CB2)
- Knowledge of the structure, organization, operation and interconnection of computer systems, the basics of programming and its application for solving engineering problems (CGB5)

2. Specific competences:
a. Cognitive (PO a, c, e)
- Understanding the behaviour of a computer and the internal structure of a computer (CECRI9)
- Data representation in a computer and arithmetic basic concepts
- Assembly programming and the relation to high level programming languages
- Levels of the memory hierarchy
- Input/output techniques

b. Procedimental/Instrumental (PO a, c, g, k)
- Ability to design and build digital systems, including computers, microprocessor-based systems and communications systems (CEIC1)
- Programming simple applications in assembly programming
- Evaluating memory cache systems
- Evaluating the performance of an assembly program

c. Attitude (PO: c, e)
- Creativity
- Critical vision of the computer structures
- Motivation
- Interest for acquiring new knowledge and information

DESCRIPTION OF CONTENTS: PROGRAMME

The basic concepts of this course are: organization and structure of a computer; data representation; basic arithmetic; execution of instructions; assembly programming; main memory; cache memory; virtual memory; input/output systems.

1. Introduction to computers
- Von Neumann architecture
- Computer programming
- Characteristic parameters of a computer
- Computer performance

2. Data representation and basic arithmetic
   - Number representation
   - Floating point
   - Basic arithmetic

3. Assembly programming
   - Machine instructions representation
   - Programming model of a computer
   - Data, instructions, and control flow structures
   - Addressing modes
   - Instructions format
   - Procedures and stack usage

4. Processor
   - Processor components
   - Control unit
   - Execution of instructions
   - Execution modes
   - Interrupts
   - Control unit design
   - Starting of a computer
   - Program execution time
   - Microcontrollers

5. Memory Hierarchy
   - Technology of memories
   - Memory Hierarchy concept
   - Cache memory

6. input/output systems
   - Input/output devices
   - Storage based on disks
   - Input/output modules
   - Input/output techniques

LEARNING ACTIVITIES AND METHODOLOGY

The teaching methodology includes:

1. Theoretical lectures 1 ECTS (PO a, c, e)
   - Computer structure theoretical concepts
   - Basic text books for both theory and problems will be also recommended.

2. Projects 1.5 ECTS (PO a, c, d, e, g, k)
   - Several projects are made along the course applying the concepts shown in theoretical lectures. Partial teacher support in computer labs.
   - Students have to analyze the requirements and provide a working solution
   - The projects are developed in team of 2 students, in order to promote the collaborative work

3. Problem solving with the teacher 1 ECTS (PO a, c, e)
   - By solving exercises and case of studies in a participatory way.

4. Student work 2 ECTS. (PO a, c, e, k)
   - Self-study to understand the theoretical concepts
   - Homework for solving proposed exercices

5. Exams 0.5 ECTS (PO a, c, e, g)
   - Midterm and small exams made along the course
   - Final exam

ASSESSMENT SYSTEM

The evaluation includes the following parts:

The continuous assessment (60 %) includes:
- Programming and laboratory projects: 30% (PO: a, c, d, e, g, k) (CB2, CEIC1)
- Exercises and small exams to perform in the smalls groups: 30 % (PO: a, e, g) (CGB5, CECRI9)

All labs are mandatory. A student follows the continuous assessment when the student makes all lab projects.
The percentage of the final exam is: 40% (PO: a, e, g). The final exam will include theoretical and practical concepts.
The minimum value for this exam will be 4.
The minimum value for the lab pojects will be 4.
The minimun value for each lab will be 2.

The final grade will be increased by 1 point to those students who complete all parts of the continuous assessment, obtain more than 7 in the continuous evaluation, and at least 4 in the final exam.

The final exam in the extraordinary period will include the theoretical and practical concepts of the course.

% end-of-term-examination:	40
% of continuous assessment (assigments, laboratory, practicals…):	60

BASIC BIBLIOGRAPHY

- Félix García, David Expósito, José Daniel García, Jesús Carretero Problemas resueltos de Estructura de Computadores, 2ª edición, Paraninfo, 2015

ADDITIONAL BIBLIOGRAPHY

- D. A. Patterson, J. L. Hennessy Computer organization and Design, Morgan Kaufmann , 2014

- J. Waldron Introduction to RISC Assembly Programming, Addison-Wesley, 1999

BASIC ELECTRONIC RESOURCES

- Félix García Carballeira, Alejandro Calderón Mateos . WepSIM: https://wepsim.github.io