Computer Structure

Academic Year: (2023 / 2024) **Review date: 09-02-2024**

Department assigned to the subject: Computer Science and Engineering Department

Coordinating teacher: GARCIA CARBALLEIRA, FELIX

Type: Basic Core ECTS Credits : 6.0

Year : 2 Semester : 1

Branch of knowledge: Engineering and Architecture

SKILLS AND LEARNING OUTCOMES

CB1. Students have demonstrated possession and understanding of knowledge in an area of study that builds on the foundation of general secondary education, and is usually at a level that, while relying on advanced textbooks, also includes some aspects that involve knowledge from the cutting edge of their field of study.
CB2. Students are able to apply their knowledge to their work or vocation in a professional manner and possess the competences usually demonstrated through the development and defence of arguments and problem solving within their field of study.
CB3. Students have the ability to gather and interpret relevant data (usually within their field of study) in order to make judgements which include reflection on relevant social, scientific or ethical issues.
CB4. Students should be able to communicate information, ideas, problems and solutions to both specialist and non-specialist audiences.
CB5. Students will have developed the learning skills necessary to undertake further study with a high degree of autonomy.
CG1. Students are able to demonstrate knowledge and understanding of concepts in mathematics, statistics and computation and to apply them to solve problems in science and engineering with an ability for analysis and synthesis.
CG3. Students can solve computationally with the help of the most advanced computing tools mathematical models coming from applications in science, engineering, economy and other social sciences.
CG4. Students are able to show that they can analyze and interpret, with help of computer science, the solutions obtained from problems associated to real world mathematical models, discriminating the most relevant behaviours for each application.
CG6. Students can search and use bibliographic resources, in physical or digital support, as they are needed to state and solve mathematically and computationally applied problems arising in new or unknown environments or with insufficient information.
CE10. Students have shown that they know and understand the algorithmic procedures to design and build programs that solve mathematical problems paying special attention to performance.
CE13. Students have shown that they understand how computers work, and the impact of their structure and operation on programs performance as well as their physical limitations.

RA2. Through sustained and well prepared argument and procedures, students will be able to apply their knowledge, their understanding and the capabilities to resolve problems in complex specialized professional and work areas requiring the use of creative and innovative ideas.
RA3. Students must have the capacity to gather and interpret data and information on which they base their conclusions, including where relevant and necessary, reflections on matters of a social, scientific, and ethical nature in their field of study.
RA4. Students must be able to perform in complex situations that require developing novel solutions in the academic as well as in the professional realm, within their field of study.
RA5. Students must know how to communication with all types of audiences (specialized or not) their knowledge, methodology, ideas, problems and solutions in the area of their field of study in a clear and precise way.
RA6. Students must be capable of identifying their own education and training needs in their field of study and the work or professional environment and organize their own learning with a high degree of autonomy in all types of contexts (structured or not).

OBJECTIVES

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

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
   - Virtual Memory

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

LEARNING ACTIVITIES AND METHODOLOGY

* Lectures: 1 ECTS. They aim to achieve the specific cognitive competences of the subject, as well as the transversal competences of analysis and abstraction.
* Practical classes: 1 ECTS. They aim to initiate the development of the specific instrumental competences, as well as the transversal competences problem solving and application of knowledge.
* Continuous evaluation exercises: 2 ECTS. Initiated during the practical classes and completed outside of them, they aim to complete the development of the specific instrumental competences and to initiate the development of the specific attitudinal competences, as well as the transversal competences problem solving and application of knowledge.
* Practical work: 1.5 ECTS. Developed without the presence of the teacher, they aim to complete and integrate the development of all the specific and transversal competences, in the resolution of two practical cases where the approach to the problem, the choice of the method of resolution, the results obtained and their interpretation are well documented.
* tutoring: TUTORIALS. Individualized assistance (individual tutorials) or in group (collective tutorials) to the students by the professor.

* Final exam: 0.5 ECTS. It aims to influence and complement the development of specific cognitive and procedural skills. It reflects especially the use of the master classes.

ASSESSMENT SYSTEM

The evaluation includes the following parts:

The continuous assessment (60 %) includes:
- Programming and laboratory projects: 30%
- Exercises and small exams to perform in the small groups: 30 %

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%. 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, 2009

ADDITIONAL BIBLIOGRAPHY

- D. A. Patterson, J. L. Hennessy Computer organization and Design RISC-V Edition, Morgan Kaufmann , 2020

BASIC ELECTRONIC RESOURCES

- Félix García Carballeira, Alejandro Calderón Mateos . The Web Elemental Processor SIMulator: https://wepsim.github.io
- Félix García Carballeira, Alejandro Calderón Mateos . CREATOR Simulator (RISC-V programming): https://creatorsim.github.io