

Operating Systems

Academic Year: (2023 / 2024)

Review date: 09-05-2023

Department assigned to the subject: Computer Science and Engineering Department

Coordinating teacher: CARRETERO PEREZ, JESUS

Type: Compulsory ECTS Credits : 6.0

Year : 2 Semester : 2

REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

Programming (First year, first semester)

Computer Structure (Second year, first semester)

SKILLS AND LEARNING OUTCOMES

¿ Understand the concept of the operating system, knowing its structure and operation, as well as being able to make use of the services of the operating system from a programme.

¿ Understand the difference between processes and threads, knowing the main scheduling policies used.

¿ Be able to develop applications with multiple processes or threads using communication and synchronization mechanisms between them, as well as concurrent programming mechanisms.

OBJECTIVES

The goal of the course is to allow the student knowing the functioning of the operating system as a expanded machine, its services for the system and its components, the major entities (processes, memory, files, etc.), concurrency, and the relations of the operating systems with the sw and hw of the computer.

DESCRIPTION OF CONTENTS: PROGRAMME

Program:

- History and evolution of operating systems
- Operating system services.
- File systems and directories
- Processes and threads
- Communication and synchronization between processes
- Fundamentals of memory management

LEARNING ACTIVITIES AND METHODOLOGY

- Theoretical classes and exercises: 42 hours (1,6 ECTS). Presential. Their objective is to achieve the specific cognitive skills of the subject. Students will receive class notes and will have basic reference texts. Activities:

* Theoretical concepts of operating systems indicated in the program, importance of the subject, critical vision of operating systems and importance of quality aspects in them. Awareness of the need to learn new concepts throughout professional life.

* Formulation and resolution of Operating Systems problems. Analysis and synthesis and application of technical concepts to solve Operating Systems problems.

* Examples during the classes that make the student aware of the possible professional and legal responsibilities due to operating system failures and their economic repercussions. As well as the impact on the company of choosing an OS solution.

* Solving exercises, practical cases and experiments in a participatory way in class (monitoring, installation, etc.). Students must study data and draw conclusions from it using professional tools, both individually and collaboratively.

- Workshops and laboratory practices. 6 hours (0,25 ECTS). Presential. Their objective is to initiate the development of specific instrumental skills, as well as transversal skills, problem solving and application

of knowledge.

- Group work and projects: 32 hours (1,3 ECTS). They develop specific instrumental competencies and most of the transversal ones, such as teamwork, the ability to apply knowledge to practice, to plan and organize, and to analyze and synthesize. They also aim to develop specific attitudinal capacities. To do this, it is:

- * They develop in groups several practices throughout the course applying principles of computer systems to the field of computer engineering and with partial support from the teacher.
- * Problem design is made from initial specifications, students must study the specifications and propose and implement a solution.
- * Students should use professional Linux and Windows OS tools to solve real-world problems.
- * They are developed in working groups in a collaborative way, thus expanding the ability to expand theoretical concepts and must demonstrate that the group is capable of developing an experiment meeting requirements and time constraints.
- * Improve communication skills through written reports of practices.

- Individual work. 56 hours (2,3 ECTS).

* Self-study of concepts and their application. Acquisition of extra information, importance of SO in the profession and need for learning.

- Tutorials: 0,6 ECTS. 25% Presential. Individualized assistance (individual tutorials) or in groups (collective tutorials) to students by the teacher.

- Final exam: 4 hours (0,13 ECTS). Its objective is to influence, complement and evaluate the development of specific cognitive and procedural capacities. It reflects especially the use of the master classes.

ASSESSMENT SYSTEM

The evaluation allows to know the degree of satisfaction of the knowledge goal, thus all work of the students will be valued by using continuous evaluation of their activities by using exercises, exams, projects, and other activities.

The following scoring will be used for continuous evaluation:

a) Ordinary Exam: 40%.

- * Activities to assess theory concepts and OS problem solving abilities.
- * It covers all the program.

b) Partial activities: 20%.

- * Partial assessments of theory concepts and OS problem solving abilities. It covers 50% of the program.
- * Extra projects or exercises requested in class.
- * Other activities requested along the course. Must be delivered on time.

c) Projects and exercises: 40%.

- * Activities must be delivered on time. They are mandatory.
- * Each project is evaluated separately, including solution adopted, functionality completeness, and design.
- * Evaluation of the project written memory. Project memory organization and correctness, written exam correctness.
- * Evaluation of tools usage.
- * Evaluation of the collaborative work of the members distinguishing roles. Responsibility of the result is shared by all members.
- * Total score for project is computed by given weight to each activity.

To pass the projects, it is mandatory to deliver of all them, to get a minim score of 2 per project, and a minimum average score of 4 for all the projects. If those criteria are not covered, the student will lose continuous evaluation.

In the case of copy detection in any project or partial exam, those students implicated will lose continuous evaluation. Copy could be among students or by taking the projects from Internet.

You will also lose continuous evaluation, if you not deliver all the projects, or do not get minimum a score of 2 in every project.

For those students not following the continuous evaluation, the ordinary exam will cover all the program (including projects). It will have a maximum value of 60% over 10.

A minimum score of 35% is required to follow the continuous evaluation.

If the student does not get the minimum, but the average of continuous evaluation and the exam is higher than 50%, the final student score will be 45%.

To pass the ordinary evaluation, the student must pass 50% considering the scores of the trajectory chosen.

To compute the final score for the extraordinary exam, the following situations are possible:

A.- Students following continuous evaluation that did not pass:

- a- Extraordinary exam will weight 35%
- b- Other 65% will come from the score of continuous evaluation.
- c- A minimum score of 40% is mandatory to pass the exam and compute the average.

B- Students not following continuous evaluation partially or totally:

- a.- Extraordinary exam will weight 100%
- b.- It may include all the topics related to the course contents, including theory and projects.
- c.- A minimum of 5 is required to pass the exam.

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

BASIC BIBLIOGRAPHY

- Abraham Silberschatz Operating System Concepts, 10th edition, Wiley & Sons, 2018
- J. Carretero, F. Garcia, F. Pérez. Problemas de Sistemas Operativos: de la base al diseño. 2ª Edición, Amazon, 2015

ADDITIONAL BIBLIOGRAPHY

- A.Silberschatz, P.B. Galvin, G. Gagner Operating Systems Concepts, Ninth Edition, John Wiley & Sons, Inc..
- F. García, J. Carretero, A. Calderón, J. Fernández, J. M. Pérez. Problemas resueltos de programación en C, Thomson, 2003. ISBN: 84-9732-102-2..