COURSE DATA

Data Subject	
Code	34679
Name	Operating Systems
Cycle	Grade
ECTS Credits	6.0
Academic year	2020 - 2021

Study (s)

Degree	Center	Acad. year	Period
1400 - Degree in Computer Engineering	School of Engineering	2	Second term

Subject-matter

Degree	Subject-matter	Character
1400 - Degree in Computer Engineering	14 - Operating systems, distributed systems and networks	Obligatory

Coordination

Name	Department
PEREZ CONDE, CARLOS	240 - Computer Science

SUMMARY

The course "Operating Systems" is a compulsory 6 ECTS, taught in the second quarter of second-degree course in Informatics Engineering and Telematics Engineering. In the degree in Informatics is part of the matter "Operating Systems, Distributed Systems and Networks."

The course covers the operating systems from three complementary viewpoints:

- The operating system as an interface for developing and running applications. From this point of view we consider the basic abstractions provided by the operating system (processes, memory, files and input / output) and the services related to them.

- The operating system as a control system that manages the use of computer resources and relies on the hardware (hardware) to ensure the proper functioning of the system.

• The OS as a program. Therefore it also takes into account aspects such as its internal structure, and the data structures and algorithms used to perform their functions.

Overall Objectives

• Show what an operating system is and what services it offers, providing an overview of the functioning of today's computers and, specifically, the roles played by the operating system.

• Show basic abstractions provided by the operating system and what operations can be done with them, emphasizing the role of the operating system as a platform for developing and running applications.

• Show the correspondence between these basic abstractions and the physical components of a computer, illustrating how the operating system requires hardware support to provide these abstractions. And how operating systems manage the physical resources available, with special emphasis on the efficiency and cost of the different solutions.

• Analyze current concepts and relate them to older ones, highlighting the benefits of new solutions and why they were introduced.

• To enable the student as a user and as a programmer in the operating system environment.

• To initiate the student in the administration of operating systems and its security.

Contents

• Introduction

• Processes and threads

• Processor scheduling

• Memory Management

• Process communication and synchronization

• Input/ouput management

• Filesystems

• Security

• Virtualization

PREVIOUS KNOWLEDGE

Relationship to other subjects of the same degree

There are no specified enrollment restrictions with other subjects of the curriculum.

Other requirements

It is recommended to have completed the following courses: Computer Science, Computer Technology, Computer Fundamentals, Computer Structure, Programming, Data structures and algorithms, and User environments.

OUTCOMES

1400 - Degree in Computer Engineering

- G4 - Ability to define, evaluate and select hardware and software platforms for the development and implementation of computer systems, services and applications, in accordance with both the knowledge and the specific skills acquired in the degree.
- R1 - Ability to design, develop, select and evaluate computer applications and systems while ensuring their reliability, safety and quality, according to ethical principles and current legislation and regulations.
- R10 - Knowledge of the features, functionalities and structure of operating systems and ability to design and implement applications based on the services that they provide.
- R14 - Knowledge and application of the fundamental principles and basic techniques of parallel, concurrent, distributed and real-time programming.
- TI2 - Ability to select, design, implement, integrate, evaluate, build, manage, exploit and maintain hardware, software and network technologies, within adequate cost and quality thresholds.

LEARNING OUTCOMES

This course allows for the following learning outcomes:

1. Describe what an operating system (OS) is and what its role is, being able to compare major operating systems.

2. Explain what are processes and threads and how they are managed by the operating system, and write simple programs which use the services to manage processes and threads.

3. Explain the advantages and disadvantages of several scheduling algorithms and evaluate their suitability based on certain objectives.

4. Explain the advantages and disadvantages of the different mechanisms of memory management including virtual memory.

5. Describe the different communication and synchronization mechanisms and select which one to use in a particular case, being able to design and implement concurrent algorithms that use them.

6. Explain the differences among different I/O devices based on how they are managed by the operating system and what is the structure of the I/O.

7. Explain the basic abstractions provided by filesystems and the services related to them, and compare different file systems.

8. Explain the objectives of operating systems security, compare different security policies and choose the most suitable for each case.

9. Explain the concept of virtual machine and the differences among different types of virtualization, identify situations where it is beneficial to use virtualization and select the most appropriate type for each case.

To complement the above results, this subject also allows to acquire the following dexterities and social skills:

Dexterities:

- Understanding what an operating system (OS) is, being able to compare among major operating systems.
- Using operating system services for sequential and concurrent application development.
- Understanding the relationship between OS services and hardware, as well as the relationship among the abstractions, getting an overall knowldedge of how an operating system works.
- Compare and select the most appropriate algorithms for the management of processes and threads, memory, I/O and filesystems.
- Install, configure and perform basic system administration taking into account operating system security.
- Solve problems that span different concepts of the subject.
- Analyze the reasons for low performance or malfunctioning of operating systems.
- Compare and select different virtualization solutions and use some of them to create and maintain virtual machines.

Social skills:

- Being able to justify in writing the work done, including the analysis of different options and why one of them was selected.
- Being able to discuss issues orally on the subject.
- Being able to collaborate with others in problem solving and implementation of programs, participating in the organization and review of group work.

DESCRIPTION OF CONTENTS

1. Introduction

Theory and problems (3T)
- Definition and purpose of operating systems
- Milestones in the development of operating systems
- Operating system performance
- Virtualization

Laboratory
- Creating a virtual machine and installing a Linux operating system (2.5 hours)
- Shell scripting(2.5 hours)
- C Language (2.5 hours)

2. Processes and threads

Theory and problems (3T +1 P)
- Concept of process Concept
- Creating and destroying
- Context change
- Multithreaded processes
- Creating and destroying threads
- Advantages and disadvantages of using multiple threads

3. Processor scheduling

Theory and problems (4T+2P)
- Short, medium and long-term scheduling
- Scheduling algorithms for single processsor systems
- Multiprocessor and real-time scheduling

Laboratory
- Process and thread creation (2,5 hours)

4. IPC and synchronization

Theory and problems (4T +3P)
- Concept of concurrency
- Communication and synchronization models
- Mutex and condition variables
- Message Passing
- Other mechanisms of communication and synchronization
- Deadlocks

Laboratory
- Concurrent Programming (2.5 hours)

5. Memory

Theory and problems (5T +2P)
- contiguous allocation
- segmented model
- paged model
- virtual memory

6. Filesystems

Theory and problems (4T +1 P)
- Filesystem concept
- Logical description: files, folders, aliases, indirect files
- Physical Description: filesystem structure, free space management, space allocation
- Example cases

Laboratory
- Filesystems (2.5 hours)

7. Security

Theory and problems (2T +1 P)
- authentication
- access control

Laboratory
- Introduction to System Administration (2.5 hours)
- Security (2.5 hours)

8. Input/output

Theory and problems (3T)
- Requirements and general structure
- Device Drivers
- Device-independent I/O software, access control, synchronous and asynchronous I/O
- User-mode code, system and I/O libraries, queue management

WORKLOAD

ACTIVITY	Hours	% To be attended
Theory classes	30,00	100
Laboratory practices	20,00	100
Classroom practices	10,00	100
Development of group work	10,00	0
Development of individual work	5,00	0
Preparation of evaluation activities	10,00	0
Preparing lectures	35,00	0
Preparation of practical classes and problem	30,00	0
TOTAL	**150,00**	

TEACHING METHODOLOGY

Theoretical classroom activities will be used to introduce the main points of the subject, providing a global and inclusive vision, analyzing in detail key issues, encouraging student participation. These activities are complemented by practical activities in order to apply the basics and expand the knowledge and experience. They include the following types of classroom activities:

- Problem-solving sessions. (Individually and in groups).
- Labs. (In couples).
- Evaluation quizs.

In addition to classroom activities, students must perform personal homework (directed bibliographic research, questions, problems, preparation of classroom activities, study). These tasks will be primarily on an individual basis, in order to promote autonomous work, but they will also include work requiring the participation of small groups of students (2-4) to build team work skills.

The e-learning platform of the University of Valencia will be used to support communication with students. This platform will provide access to course materials.

EVALUATION

The subject may be evaluated in two ways, one giving greater weight to classroom activities and a another one giving greater weight to the final exam. All students will have as final score the higher of the two.

The evaluation of the course will take place in the first call by:

- Evaluation of theory and problems (TP).

This part will count 75% of the final grade and it will be necessary to obtain 4,5 out of 10.

- Continuous assessment (CA), based on the participation and degree of involvement in the teaching-learning process, taking into account regular attendance and participation. This part can not be recovered.
- Tests, which consist of both theoretical and practical questions. The tests will be carried out in the first half of the semester (T1), during the second half of the semester (T2) and in the testing period (T3).

Each of these tests will cover all course content covered until then.

The TP grade will be computed as follows:

TP = 0.15 * CA + 0.15 * T1 + 0.25 * T2 + 0.45 * T3.

- Laboratory activities (L) will be assessed based on the achievement of objectives in the laboratory sessions.

Activities will be performed in pairs, their weight being 25% of the final mark. All laboratory sessions will have the same weight on the final grade.

It will be necessary to get 4,5 out of 10 in this part. (Both in the first and second calls).

If unable to attend a session, the student will have to show its work to the professor in person during attendance hours. Students must be prepared to answer questions about the conduct of the practice and to perform parts of it at the moment (with minor changes). This type of delivery has to be done before any laboratory session has taken place and will be penalized with 20%.

The mark will be formed in the case of following the continuous assessment as the sum of the previous parts as follows:

- If TP <4,5 or L <4,5

Final_grade = Minimum (TP, TL)

- Otherwise:

Final_grade = 0.75 * TP + 0.25 * L

In case of not passing the subject by following the continuous assessment model (or if the grade calculated in this second way is more favorable to the student), the T3 assessment test will be the final course exam and TP will be calculated as follows:

TP = 0.15 * CA + 0.85 * T3

The final grade is computed in the same way as with continuous assessment.

In the second call the course will be assessed in the same way as in the first call, with the following exceptions:

- A new delivery period for lab assignment will be opened (with the same conditions as in the 1st call), except for the penalty, which will be 30%. The deadline for submission is the day before the second call test.
- The second call test replaces the T3 test.
- The EC mark will be the same than in the first call.

REFERENCES

Basic

- Sistemas Operativos. William Stallings. Prentice Hall.
- Fundamentos de Sistemas Operativos. Abraham Silberschatz, Peter Baer Galvin y Greg Gagne. John Wiley & Sons.
- Sistemas Operativos. Una visión aplicada. Jesús Carretero, Félix García, Pedro de Miguel y F. Pérez. McGraw-Hill.

Additional

- "C for Programmers with an Introduction to C11". Harvey Deitel; Paul Deitel. Prentice Hall.
- "Programming in C". Stephen G. Kochan. Addison-Wesley Professional.
- Unix and Linux System Administration Handbook, Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley. Prentice Hall.

ADDENDUM COVID-19

This addendum will only be activated if the health situation requires so and with the prior agreement of the Governing Council

The teaching methodology for this subject will follow the model approved by the Academic Committee of the GII / GIM degrees (https://links.uv.es/catinfmult/modeloDocent). If the facilities are closed because of COVID-19 pandemics, the scheduled lectures will be replaced by synchronous online sessions within the assigned time slots of the course, using the tools provided by the university.

If the facilities need to be closed due to the pandemics causing any of the evaluation exercises to be held at ETSE-UV, these exercises will be substituted by equivalent exercieses held online using the tools provided by the university. The weights for each activity will remain the same as specified in the teaching guide.