

## Operating Systems

Code: 102747  
ECTS Credits: 6

Degree	Type	Year	Semester
2502441 Computer Engineering	OB	2	1

### Contact

Name: Antonio Espinosa Morales

Email: AntonioMiguel.Espinosa@uab.cat

### Use of Languages

Principal working language: spanish (spa)

Some groups entirely in English: No

Some groups entirely in Catalan: Yes

Some groups entirely in Spanish: No

### Teachers

Porfidio Hernández Bude

Xiaoyuan Yang

Carles Carrillo Jordan

Santiago Marco Sola

Montse Serra Vizern

### External teachers

Pere Pons

### Prerequisites

Previous content:

- Students should be familiar with Fundamentals of Computers, Fundamentals of Computing and Programming Methodology

### Objectives and Contextualisation

Students will become familiar with the services provided by the Operating System attending its functional description. We will cover management policies applied to system basic modules. Important concepts include: process and thread model, communication between processes and threads, Operating System design philosophies, file servers, Input/Output modules, security of input /output operations, security of Operating and Distributed systems

### Competences

- Acquire personal work habits.

- Capacity to design, develop, select and evaluate computer applications and systems, ensuring reliability, security and quality, in accordance with ethical principles, and applicable standards and legislation.
- Have the capacity to define, evaluate and select hardware and software platforms for the development and execution of computer systems, services and applications.
- Know and apply the basic and main techniques of parallel, concurrent, distributed and real time programming.
- Know and apply the functional and structural characteristics of distributed systems and computer and Internet networks, and design and implement applications based on these.
- Know the characteristics, functionalities and structure of operating systems and design and implement applications based on their services.

## Learning Outcomes

1. Create thread and socket based applications, including mutex primitives and condition-type variables.
2. Desenvolupar estratègies d'aprenentatge autònom.
3. Design, develop, select and evaluate computer systems, ensuring their reliability, security and quality.
4. Determine the influence of the operating system on the run performance of applications.
5. Evaluate an operating system on different work levels: user, advanced user, programmer, advanced programmer, etc.
6. Know and apply the basic and main techniques of concurrent and distributed programming.
7. Know and apply the characteristics, functionalities and structure of distributed hardware systems and computer networks to design and implement applications based on the same.
8. Know and understand the concepts related with computer networks, knowing how to situate them in a hierarchical system of protocols.
9. Prevent and solve problems.
10. Work with commands and command sequences (Shell Scripts), and with the programming interface (API) of the operating system.

## Content

### 1-Subject presentation

Description of contents and subject methodology

### 2-Introduction to Operating System

Operating system kernels and its meaning considering design philosophies. Virtualization. Tuning of of the operating system. System performance measures.

### 3. Process management

Processes, threads and Containers. Concurrency, Deadlock, CPU Scheduling in real time systems

### 4. Memory management.

Mono/multiprogramming models. MFT, MVT systems. Paging and segmentation. Virtual memory. Paging and Segmentation on demand

### 5. I/O Management

I/O Interface architecture. Device management. Device driver design. Device controllers

### 6. File Systems

Logical and physical visions of the module. Process of adress translation. Request prediction. Advanced storage systems

### 7. Advanced topics

## Methodology

Theory classes will be based on lectures, although students will be encouraged to actively participate in the resolution of examples and practical questions. Theory classes will be scheduled in the official subject planning published in the Engineering school degree web site. The syllabus is detailed in the session list that will be available the first day of class.

Problem sessions are defined in the timetable published in the Engineering school degree web site. Problem sessions are organized in small groups of students following alphabetical order of surnames. The list of problems to be solved during sessions will be published the first day of class. This list will be the guide of the practical work during the course.

Laboratory sessions will be distributed in different sessions during the course following the timetable published in the Virtual Campus of the subject. Lab teachers will generate the list of students and sessions to attend. All laboratory work will be done in groups of two students

Tranversal competences

In this subject we will evaluate transversal competences T02.02 - Develop strategies of autonomous learning and T02.04 - prevent and provide solutions to problems that will be evaluated mainly in problems and laboratory sessions.

## Activities

Title	Hours	ECTS	Learning Outcomes
Type: Directed			
Lab sessions	10	0.4	5, 6, 7, 1, 2, 10
Theory lectures	22	0.88	5, 6, 7, 8, 1, 4, 3, 9
problem classes	10.5	0.42	5, 6, 7, 1, 4, 9, 10
Type: Autonomous			
Autonomous	100	4	5, 6, 7, 8, 1, 2, 4, 3

## Assessment

a)Evaluation process and activities

The subject is composed of the following activities:

Theory: 60% of the final grades. It is divided into two written examinations where each of the tests corresponds to 30% of the final grades.

Problems: 10% of the final grades

Laboratory: 30% of the final grades. 20% belongs to the evaluation of the work done in the sessions and 10% of the reports of the practical labs

Students must have a minimum grade of 5 in theory and problems and a final global evaluation of 5.0 to pass the subject. In case of not obtaining this minimum, the numerical value in the student record will be the lower value between 4,5 and the average of the grades.

Theory evaluation will be done in the form of two written examinations. The test planning and the topics to be evaluated in each of the tests will be described in the subject planning sheet. Theory final grade will be the average of the theory tests when each of the tests grades is higher than 5. It is important to state here that second test will not cover the first part of the theory of the subject.

Problem evaluation will be done in three specific sessions of 30 minutes where it will be necessary to answer and deliver an exercise in the classroom. The description of the problem to solve will be available at the beginning of the session. These exercises will be solved individually and the grade obtained will be published in the virtual campus. Only those students registered in their problem session will be allowed to deliver the test and get an evaluation. If a student delivers the exercises in a class where he or she are not registered then those exercises will not be evaluated. Problem final grade will be obtained from the average of the three evaluated tests.

Laboratory work: Last year students must repeat the practical work again. Lab group teams will be managed using the virtual campus. Activation date to register in a specific lab will be public in the news part of the subject in the virtual campus. Remember that attendance is compulsory for all the sessions for all the group members.

Laboratory sessions are of compulsory attendance. Students must show the adequate work of lab deliveries that will be verified by the lab teachers. Each student in the lab also must answer correctly the questions of the teachers and must deliver in due time a written report of the lab sessions. Contents of the report will be:

- Objective of the lab
- Description and proposed work: pseudo code, block design
- Description of the procedures used and their functionality
- Description of the problems found during the lab sessions and solutions proposed for the problems.
- Conclusions of the work done in the lab
- Source code of the work
- Procedure steps of building an application from provided source code

Additionally, the last minutes of the lab sessions will be dedicated to a short individual evaluation test. The result of this test will be a PASS/NO PASS evaluation. If the result obtained is a PASS, the lab evaluation will be validated. On the contrary, the student will have to pass a new lab test at the end of the semester in the evaluation period. The final lab grade will be calculated using the average of each of the labs.

#### b) evaluation activities calendar

The actual calendar of the evaluation activities will be published the first day of class in the virtual campus and in the Engineering school website. These dates could be modified, so any modification on calendar will be published in the virtual campus that will be the main channel of official documentation and notifications between students and teachers.

#### c) final examination process

Theoretical content of the subject will be evaluated in two different exams during the course. The second test will not cover the topics of the first part of the subject.

Those students that have not reached the minimum grade to pass the subject will have a final examination of the subject theory at the end of the semester. Labs and problems are not re-examined. Students will have to take a final exam of each part of the subject previously evaluated with a grade smaller than a 5.0. Students must get a grade above 5.0 in each of the two parts of the subject to pass the theoretical part and calculate the average with problems and labs for the final subject evaluation grade.

After the final examination of theory, there will be a lab exam for those students with a grade greater or equal than 4.5 and any NO PASS lab evaluation. The date of this test will be published in the virtual campus in the general planning of the subject. Students that have not passed any of the labs during the semester will have to

pass a specific practical test. This test will be scheduled the same day of the final theory exam.

#### d)Qualification revision

For each evaluated activity, there will be a place, date and hour for the students to review the activity with the teacher. If the student does not attend the revision date the activity will not be revised again.

#### e)Qualifications

A student can get the maximum qualification when the final grade is above 9. In order to provide a selection of this qualification we will consider the maximum amount of honor qualifications that can be given following the general UAB regulations.

Given the continuous evaluation methodology used in this subject, if a student delivers any evaluable proof of work like problem test, lab report or theory test it will be interpreted as the student will be evaluated and to get a different evaluation than Non-Evaluable (NA). A NA qualification can only be obtained when absolutely no delivery is made during the semester.

Last, it is important to consider the importance of correct formal expression in any delivery provided by students. The quality of the text of the deliveries will be evaluated. All errors found related to grammar, spelling or low quality text expressions will account up to 20% of the final grade.

#### f)Irregular behavior and plagiarism

According to the current UAB norms, and apart from any other disciplinary measures, any plagiarism activity detected in an evaluated activity will be qualified as zero (0). Activities qualified in this way will not be reexamined during the course. If plagiarism is detected in a compulsory activity, the subject will be considered not passed and there will be no further opportunities to get a new evaluation during the course. These activities include, among others:

- Total or parcial copy of a lab, report or any other evaluation activity
- encourage any plagiarism activity
- deliver a group work not done by all the components of the group
- deliver material of others as it was done by the components of the group, being those translations or adaptations.
- Have communication devices during the individual evaluation tests: smartphones, smart watches, cameras installed in pens, etc.
- Speak with students during evaluation tests or exams
- Copy or try to copy from other students during the evaluation tests or exams
- Use or try to use written material during the evaluation tests or exams when these have not been explicitly permitted

In summary, any plagiarism activity in any of the evaluation activity will be equivalent to the impossibility of passing the subject, without any opportunity of re-examine in the same course and without any compensation of any part of the subject in the next courses.

#### h)Last year students evaluation

Students taking their third evaluation of the subject can compensate the labs done in the last course. Also, their final grades will be calculated using the average of the exams of both parts of the subject. Students that meet this requirements must request this special evaluation to the teacher during the first two weeks of class

## Assessment Activities

Title	Weighting	Hours	ECTS	Learning Outcomes
2 theory test evaluations	60%	4	0.16	5, 7, 8, 2, 4, 3
Lab evaluation	25%	2	0.08	5, 6, 1, 10
Problems evaluations (3)	10%	1.5	0.06	5, 6, 7, 8, 2, 4, 3, 9

## Bibliography

### Theory:

- "Sistemas Operativos", Pedro de Miguel Anasagasti, Fernando Pérez Costoya. D. Arquitectura y Tecnología de Sistemas Informáticos, ETSE, UPM. 2016. Under license Creative Commons NoComercial CompartirIgual 4.0.
- "Sistemas Operativos: una visión aplicada". Segunda edición. Jesús Carretero, Félix García, Pedro de Miguel y Fernando Pérez. McGraw-Hill, 2007.
- "Operating System Concepts", Avi Silberschatz, Peter Baer Galvin, Greg Gagne, Tenth Edition, John Wiley & Sons, Inc, April 2018, ISBN: 978-1-119-32091-3 (e-book)
- "Sistemas Operativos". Silberschatz, Galvin y Cagne. 7a, Ed. McGrawHill, Edición. 2006.
- "Sistemas Operativos". William Stallings, 5a Edición Prentice Hall 2005

### Lab sessions:

- "Unix Programming Environment", R. Pike & Brian Kernighan, Ed. Mc. Graw-Hill
- "Advanced Unix programming", Rockind M. Ed. Prentice-Hall
- "Administración avanzada del sistema operativo GNU/Linux". Remo Suppi, Josep Jorba. Universitat Oberta de Catalunya, setiembre 2014. <http://hdl.handle.net/10609/61266>