Course guide
270124 - SOA - Advanced Operating Systems

Last modified: 30/01/2024

Unit in charge: Barcelona School of Informatics
Teaching unit: 701 - DAC - Department of Computer Architecture.

Degree: BACHELOR'S DEGREE IN INFORMATICS ENGINEERING (Syllabus 2010). (Optional subject).

Academic year: 2023 **ECTS Credits:** 6.0 **Languages:** Catalan, Spanish

LECTURER

Coordinating lecturer: MANUEL ALEJANDRO PAJUELO GONZALEZ

Others: Primer quadrimestre:
JUAN JOSÉ COSTA PRATS - 12
MANUEL ALEJANDRO PAJUELO GONZALEZ - 11, 12

Segon quadrimestre:
MANUEL ALEJANDRO PAJUELO GONZALEZ - 11, 12

PRIOR SKILLS

The student must have the technical capabilities that will confer the subjects studied previously together with a medium level of technical English to read and understand documentation.

The technical capabilities could be summarized as follow:

-Operating systems: Understanding the basics of an operating system along with the creation of applications using the generic system call interface explained during the Operating System course.
-In terms of computer architecture: Knowledge of the main elements of a computer, how these elements relate to each other, internal representation of data and knowledge of the assembler language.
-In terms of programming: Ability to code complex programs from scratch composed of several moduls. Definition of data types, pointers and references, and assembler code. Compilation and linkage of executables.

REQUIREMENTS

- Prerequisite SO
- Pre-Corequisite SO2

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:
CT6.1. To demonstrate knowledge and capacity to manage and maintain computer systems, services and applications.
CT6.3. To demonstrate knowledge about the characteristics, functionalities and structure of the Operating Systems allowing an adequate use, management and design, as well as the implementation of applications based on its services.
CT6.4. To demonstrate knowledge and capacity to apply the characteristics, functionalities and structure of the Distributed Systems and Computer and Internet Networks guaranteeing its use and management, as well as the design and implementation of application based on them.
CT7.1. To demonstrate knowledge about metrics of quality and be able to use them.
CT7.2. To evaluate hardware/software systems in function of a determined criteria of quality.
CT8.7. To control project versions and configurations.
CTI1.4. To select, design, deploy, integrate, evaluate, build, manage, exploit and maintain the hardware, software and network technologies, according to the adequate cost and quality parameters.
CTI3.4. To design communications software.

Generical:
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

TEACHING METHODOLOGY

The course will present two types of class: Theory and laboratories. The theory classes will explain the concepts, designs and implementing of several common components of a current operating system.

The lab classes will be carried out weekly. The first seven weeks the student will design, implement and evaluate a process scheduler in an operating system called Zeos. The last 7 weeks tthe student will design, implement and evaluate new operating system features.

LEARNING OBJECTIVES OF THE SUBJECT

1.Know the behavior of a real OS from booting the computer, the system initialization, the dynamic management of resources, to the shutdown of the computer.
2.Know the etails of the implementation of some of the basic components of a real OS: initialization code, memory management code, input / output management code, process management and executable file management.
3.Describe the procedure to extend, dynamically, the kernel of a real OS
4.Detail the internal structure of a kernel module identifying the distinct components as well as its relationship with the generic OS interface and the use of structures in memory for controlling the input/ouput.
5.Know the multithreading programming paradigm, the problem of sharing memory and the implementation of mechanisms for mutual exclusion with the required hardware support.
6.Implementing some of the basic components of a real OS: boot code, memory management code, input / output management code, process management code and file execution code using C and assembler on an Intel x86 architecture.
7.Compare and evaluate different alternatives for implementing resource management code using metrics of cost, efficiency and quality
8.Design the communication services for multiprocess and/or multithreaded applications.
9.Use and understand technical documents in English provided with the operating system.
10.Use version control software.

STUDY LOAD

Type	Hours	Percentage
Hours small group	30,0	20.00
Guided activities	6,0	4.00
Self study	84,0	56.00
Hours large group	30,0	20.00

Total learning time: 150 h

CONTENTS

System boot

Description:
This chapter will explain all the actions performed to start all the services offered by an operating system. We also describe and discuss possible implementations of this process in a current operating system.

System calls, exceptions and interrupts.

Description:
This chapter describes and discusses the possible implementations of the mechanisms to enter the system along with its hardware support. We will detail the most important aspects of the mechanisms and the dependencies between the operating system and hardware. We will discuss the benefits of implementing current virtualization techniques. We will present and discuss several implementations of current operating systems.

Memory management

Description:
This chapter will address the following issues: logical address space of the process. Memory-based paging systems and support hardware. Design and implementation of virtual memory. Replacement algorithms for virtual memory. Implementation of shared memory. Alternative design and implementations of memory systems in modern operating systems.

Process management

Description:
This chapter will address the following issues: detailed implementation of the process control block. Implementation of the creation of a process. Implementation of the finalization of a process. Implementation of the execution of an executable file. Implementation of the context switch between processes. Description of the structures and algorithms for process scheduling. Implementation of the routines for process scheduling. Description and discussion of several implementations of process scheduling in current operating systems.

Input/output management and the system file

Description:
This chapter covers the following topics: description and implementation of the independent and dependent parts of an OS with respect to a device. Implementation of a device descriptor. Description, implementation and operation of the structures for the input / output and the file system. Description and implementation of the logical structure of system files in disk. Mechanisms of communication through the file system. Mechanisms of communication through a network device. Description of the implementation of several current file systems.

Shared memory

Description:
This chapter covers the following topics: problems of memory sharing between multiple threads of execution within a process. Race conditions. Areas of mutual exclusion. Description and implementation of mechanisms for mutual exclusion to access shared memory, in particular, test and set, mutex and semaphores. Description, discussion and implementation of mechanisms for mutual exclusion in modern operating systems.

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Extension of the operating system kernel

Description:
This chapter covers the following topics: concept, design and implementation of OS modules. Loading of kernel modules at system initialization time and at run time. Description of the access to features implemented in kernel modules. Understanding the relationship between kernel modules and hardware devices and software.

ACTIVITIES

Familiarisation with the framework

Description:
Acquire the knowledge to develop the laboratory.

Specific objectives:
1, 2, 6

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 6h
Laboratory classes: 2h
Self study: 4h

Develop a RSI

Description:
Develop a RSI for the internal clock

Specific objectives:
1, 2, 6, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 12h
Theory classes: 2h
Laboratory classes: 2h
Self study: 8h

Creation of an OS entry point

Description:
Creation of a new entry point into the system for internal configuration

Specific objectives:
1, 2, 6, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 8h
Theory classes: 2h
Laboratory classes: 1h
Self study: 5h

Memory management

Description:
Assimilate the contents of this topic

Specific objectives:
1, 2, 6, 7

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 13h
Theory classes: 4h
Laboratory classes: 3h
Self study: 6h

Process management

Description:
Implementation of the process management components (creation, scheduler, scheduling policies, ...)

Specific objectives:
1, 2, 6, 7, 9, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 21h
Theory classes: 6h
Laboratory classes: 7h
Self study: 8h

First midterm exam

Specific objectives:
1, 2, 3, 4, 7

Full-or-part-time: 8h
Guided activities: 2h
Self study: 6h

Practical contents midterm exam

Specific objectives:
1, 2, 3, 4, 9, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 8h
Guided activities: 2h
Self study: 6h

Kernel extension

Description:
Assimilate the concepts of this topic

Specific objectives:
1, 2, 3, 4, 9

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 7h
Theory classes: 1h
Self study: 6h

Input and output and the file system

Description:
Assimilate the concepts of this topic

Specific objectives:
1, 2, 6, 7, 8, 9

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 15h
Theory classes: 5h
Laboratory classes: 4h
Self study: 6h

Interprocess communication

Description:
Development of multiprocess and multithreaded applications, with information sharing, and evaluation of their performance

Specific objectives:
1, 2, 5, 6, 7, 8, 9, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 10h
Theory classes: 2h
Laboratory classes: 2h
Self study: 6h

Analysis of the system performance

Description:
The student will think, decide and evaluate various metrics at system level granularity

Specific objectives:
1, 2, 6, 7, 9, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 10h
Theory classes: 1h
Laboratory classes: 4h
Self study: 5h

Project

Specific objectives:
1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 6h
Self study: 6h

Second midterm exam

Specific objectives:
1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 8h
Guided activities: 2h
Self study: 6h

Theory final exam

Specific objectives:
1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 9h
Guided activities: 3h
Self study: 6h

Final lab exam

Specific objectives:
2, 5, 6, 7, 8, 9, 10

Related competencies :
G3. THIRD LANGUAGE: to know the English language in a correct oral and written level, and accordingly to the needs of the graduates in Informatics Engineering. Capacity to work in a multidisciplinary group and in a multi-language environment and to communicate, orally and in a written way, knowledge, procedures, results and ideas related to the technical informatics engineer profession.

Full-or-part-time: 9h
Guided activities: 3h
Self study: 6h

GRADING SYSTEM

The final grade for the course consists of the grade of the core technical competences (CT), and the grade of generic competence (CTr) by the formula:

Final Grade = (CT + CTr) * (10/11)

Where the maximum score of CTr is 1.

The grade of the CT can be obtained by continuous assessment (CTc) or excepcionally by a final exam (CTf). Is calculated as:

CT = max (CTc, CTf)

Where grade CTc is composed of several evaluative acts: theory exams (T) and laboratory exams (L). The formula for calculating this grade is as follows:

CTc = 40% T + 60% L

To calculate T two assessments are used with the same weights:

T = 50% T1 + 50% T2

To calculate L, one assessment (L1), a follow up grade (S) and a project (P) are used:

L = 30% L1+ 20% S + 50% P

The follow up grade (S) is assigned by the teacher as an evaluation of the right progress throughout the laboratory.

The project grade (P) is obtained with the design, follow up and implementation of the final project.

The CTf grade is calculated using a theory exam (T) and a laboratory exam (L). To accomplish this grade is mandatory to do both exams. This option is only available for those students that fail the continuous assessment. The formula is as follows:

CTf = 50% T + 50% L

The grade of generic competence (CTr) is obtained during the semester through various activities. Rating of this competence will have values A, B, C, D or NA: A corresponds to an excellent level, B to a desired level, C to a sufficient level, D to a failed level, and NA to not assessed.

BIBLIOGRAPHY

Basic:
- Silberschatz, A.; Galvin, P.B.; Gagne, G. Operating system concepts. Global ed. (10th ed.). John Wiley & Sons, 2019. ISBN 9781119454083.
- Becerra, Y; Costa, J.; Pajuelo, A. Advanced Operating Systems Course.

Complementary:
- Bovet, D.P.; Cesati, M. Understanding the Linux kernel. 3rd ed. Beijing: O'Reilly, 2005. ISBN 9780596005658.
- Yosifovich, P.; Ionescu, A.; Russinovich, M.E.; Solomon, D.A. Windows internals: part 1: system architecture, processes, threads, memory management, and mor. 7th ed. Redmon, Wa: Microsoft Press, 2017. ISBN 9780735684188.
- Stallings, W. Operating systems: internals and design principles. 9th ed. Harlow: Pearson Education Limited, 2017. ISBN 9781292214306.

RESOURCES

Hyperlink:
- http://docencia.ac.upc.edu/FIB/grau/SOA/