UNIVERSIDAD AUTÓNOMA DE MADRID

17818 - PRACTICAL COMPUTER WORKSHOP

Information of the subject

Code - Course title: 17818 - PRACTICAL COMPUTER WORKSHOP

Degree: 473 - Graduado/a en Ingeniería Informática
722 - Graduado/a en Ingeniería Informática

Faculty: 350 - Escuela Politécnica Superior

Academic year: 2020/21

1. Course details

1.1. Content area

Seminarios-Taller de Informática

1.2. Course nature

Compulsory

1.3. Course level

Grado (EQF/MECU 6)

1.4. Year of study

1

1.5. Semester

First semester

1.6. ECTS Credit allotment

6.0

1.7. Language of instruction

Español, English

1.8. Prerequisites

----- STI-HARDWARE -----

No prerequisites are needed to attend this course.

1.9. Recommendations

----- STI-HARDWARE -----

None

1.10. Minimum attendance requirement

----- STI-HARDWARE -----

In this subject students can choose between two different assessment methods (see section 4). Based on the selected assessment method, the minimum required attendance is:

CONTINUOUS ASSESSMENT: Attendance is mandatory. Student must attend a minimum of 85% of the course hours.

FINAL EXAM: Attendance is highly advisable but not mandatory.

---- STI-SOFTWARE -----

In this subject students can choose between two different assessment methods.

Based on the selected assessment method, the minimum required attendance is:

CONTINUOUS ASSESSMENT: Attendance is mandatory. Student must attend a minimum of 85% of the course hours.

FINAL EXAM: Attendance is highly advisable but not mandatory.

1.11. Subject coordinator/s

David Renato Dominguez Carreta, Ivan Gonzalez Martinez

https://autoservicio.uam.es/paginas-blancas/

1.12. Competences and learning outcomes

1.12.1. Competences

C13 Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los sistemas de información, incluidos los basados en web.
IC1 Capacidad de diseñar y construir sistemas digitales, incluyendo computadores sistemas basados en microprocesador y sistemas de comunicaciones.

1.12.2. Learning outcomes

----- STI-HARDWARE -----

The results of learning are:

- Knowledge of structure of a computer and its technological development.
- Ability to identify, assemble and update the components of a computer.
- Ability to manage analytical and measuring tools used to detect and correct typical hardware problems.

----- STI-SOFTWARE -----

The results of learning are:

- Knowledge of different operating systems at user (advanced) and administrator (intermediate) level
- Get familiar with control flow in a computer program
- Ability to effectively and proficiently use programming environments, including code editing, compilation, a ssembly and debugging
- Ability to design and test a program to check for correctness, efficiency, usability and reliability
- Use of some specialized computer applications in a Linux environment
- Use of project management, release management, collaborative work tools, etc.

1.12.3. Course objectives

----- STI-HARDWARE -----

General objectives of this subject are:

- Ability to identify the architecture of a computer and its different components, including their technology evolution.

- Ability to identify, assemble, and update the components of a computer.
- Knowledge of actions to manage electrical and digital components.
- Knowledge of voltage levels in a computer.
- Knowledge of tools to assemble/dismantle a computer.
- Knowledge of tools to analyze and detect incidents in hardware components.

----- STI-SOFTWARE -----

General objectives of this subject are:

1 – Characterize the computer science discipline.

2- Describe computer structure and operation according to their two main components: hardwar and software.
3- Identify, describe and relate the tasks performed by an operating system.
4- The use of UNIX environment.
5- Design, codify, debug, test, execute and interpret programs in the C and Shell language.

1.13. Course contents

----- STI-HARDWARE -----

Topics

1. Architecture of a computer and the Technological Evolution
    1. Computer architecture.
    2. Computer components.
    3. Microprocessors.
    4. Memories.
    5. Storage.
    6. Buses.
    7. Ethernet and Wireless communications.
    8. Input/Output.
    9. Power supply.
2. Computer components.
    1. Actions to manage electrical and digital components.
    2. Tools to assembly a computer.
    3. Voltage levels in a computer.
        1. Digital and computer systems voltage standards.
        2. Testing voltage levels.
    4. Assembly a computer.
    5. Tools to analyze and detect incidents.
3. Analysis and measurement tools.
    1. Power supply.
    2. Multimeter.
    3. Oscilloscope.
    4. Signal/Function generator.
    5. Others.
4. Digital systems.
    1. Microprocessor-based systems.
    2. Digital systems.

--- STI-SOFTWARE -----

Topics

1. **INTRODUCTION TO COMPUTER SCIENCE**

1. Basic concepts.
2. Computer structure and operation.
3. Computer software.

2. **BASIC CONCEPTS ABOUT OPERATING SYSTEMS**
    1. Operating System (OS) definition.
    2. OS tasks: processor, memory and I/O management.
    3. UNIX and Linux environments.

3. **GETTING STARTED WITH PROGRAMMING**
    1. Use process documentation tools such as flowcharts
    2. Introduction to Shell scripts.
    3. Programming application running in a Linux environment.

4. **ADVANCED USE OF PROGRAMMING TOOLS**
    1. Program structuring and modular programming.
    2. Run, interpret and test programs in Linux environment.
    3. Validating, debugging and executing an application in Linux.

1.14. Course bibliography

----- STI-HARDWARE -----

There is not course bibliography because the subject is mainly practical. The course material is enough and will be provided to students through the web page of the subject.

----- STI-SOFTWARE -----

There is not course bibliography because the subject is mainly practical. The course material is enough and will be provided to students through the web page of the subject.

A short list of auxiliar references is:

- E. ANGUIANO, D. CAMACHO, C. NAVARRETE. *LINUX, Guía de Aprendizaje*. Prentice Hall, 2008
- J. GARCÍA de JALÓN, I. AGUINAGA, A. MORA. *Aprenda LINUX como Si Estuviera en Primero*. (Universidad de Navarra, 2000).
- B. W. KERNIGHAN. *El Entorno de Programación UNIX*. (Prentice Hall Hispanoamericana, 1987).
- A. SIERRA URRECHO. *Programación en C/C++*. (Anaya Multimedia, 2005).

2. Teaching-and-learning methodologies and student workload

2.1. Contact hours

	#hours HW	#hours SW
Contact hours (minimum 33%)	35	34
Independent study time	40	41

2.2. List of training activities

Activity	# hours HW	#hours SW
Lectures	10	8
Seminars	2	
Practical sessions	16	20
Clinical sessions		
Computer lab		
Laboratory		
Work placement		
Supervised study		
Tutorials	3	3

Assessment activities	4	3
Other		

3. Evaluation procedures and weight of components in the final grade

3.1. Regular assessment

NOTICE: Hardware Workshops are part of the Computer Science and Engineering Seminars-Workshops subject and it is necessary a grade of 5 to pass the subject. The final grade of Computer Science and Engineering Seminars-Workshops will be the average of Hardware Workshop and Software Workshop. A grade of 5 will be necessary in both subjects to calculate the average. In other case, the final grade will be: (50% x Minimum (5, Software Workshop grade) + 50% x Minimum (5, Hardware Workshop)).

----- STI-HARDWARE -----

Grading for the continuous assessment method, apart from the required 85% of attendance, is the defined by the following equation:

Grade = 0,6*Practical_Exercises + 0,2*Seminar + 0,2*Tests

- Practical exercises require a min of 3.5 in all tasks. Fail one task will suppose to fail the current session.
- Tests are two modular exams that students will do during the semester.
- Seminar is a presentation about a topic of free election related to the subject. Execution is not mandatory but the grade will be 0.
- Students must do two practical exercises to obtain a final grade. Less than two exercises will grade as “No grade”.

Grading for the final exam method will be only the grade of the final exam. Students must send an email to the coordinator informing of their intention to take the exam, at least 2 days before the exam date.

NOTE: Practical exercises grading will be only valid for the extraordinary exam session of the same academic year and the next one.

---- STI-SOFTWARE -----

The qualification of the continuous assessment method, apart from the requirement of compulsory attendance at 85% of the classes, will be the weighted average of the evaluable deliveries that will have to be greater than or equal to five to pass the course.

Otherwise, the student must perform the Final Exam (EF) that will be done in the assigned period, and involves all the topics developed in this course, along with the students who have opted for the non-continuous evaluation.

In this case, the final grade will be the grade of the final exam that must be greater than or equal to five to pass the course.

The student will only receive the grade "Not evaluated" if he/she doesn't submit any delivery.

3.1.1. List of evaluation activities

Evaluatory activity	% HW	% SW
Final exam	0 (continuous ast.) 100 (final exam ast.)	0 (if continous) 100 (otherwise)
Continuous assessment	100	100

3.2. Resit

----- STI-HARDWARE -----

Grading for the extraordinary exam session will be the grade of the exam.

- Students with no practical exercises grading will have to send all practical exercises to the instructor 1 week before the exam.
- Students with practical exercises grading will not have to do any additional practical exercise before taking the exam,

but they must send an email to the coordinator informing of their intention to take the exam, at least 2 days before the exam date.

----- STI-SOFTWARE -----

Grading for the extraordinary exam session will be the grade of the exam averaged with the common assesment of the continuous course:

FS=(NC+FE)/2

where NC is the weighted average grade of the submitted delivery at extraordinary period and FE is the grade of the final exam.

3.2.1. List of evaluation activities

Evaluatory activity	% HW	% SW
Final exam	100	50
Continuous assessment	0	0
Deliveries at the extraordinary period	0	50

4. Proposed workplan

----- STI-SOFTWARE -----

Week	Content	Classroom hours	Hours outside the classroom
1	- Subject presentation and student team formation.	2	-
2	- **Unit 1 Introduction to Computer Science**	2	2: Read Unit 1
3+4	- Workshop on Assessment 1.	2	4: Develop Assessment 1
5	- **Unit 2 Basic Concepts about Linux**	2	2: Read Unit 2
6+7	- Workshop on Assessment 2.	4	4: Develop Assessment 2
8	- **Unit 3 Basic use of programming tools**	2	2: Read Unit 3
9	- Assessment 3 presentation.	2	2: Develop Assessment 3
10+11	- Workshop on Assessment 3.	4	4: Finish Assessment 3
12	**Unit 4 Advance use of programming tools**	2	2: Read Unit 4
13	- Assessment 4 presentation	2	2: Develop Assessment 4
14+15	- Workshop on Assessment 4.	4	4: Finish Assessment 4

----- STI-HARDWARE -----

Week	Content	Contact hours	Independent study time
1	Presentation of the subject using this document. Description of the web tool moodle.	2 - Define partners	1 - Read this document - Log in moodle and register for this subject
2	Topic 1: Architecture of a computer and the Technological Evolution - 1.1. Computer architecture. - 1.2. Computer components. - 1.3. Microprocessors. - 1.4. Memories. - 1.5. Storage. - 1.6. Buses.	2 - Exercise 1 Task 1: Investigate a computer component	1 - Document P1T1 - Submit P1T1
3	Topic 2: Architecture of a computer and the Technological Evolution - 1.7. Ethernet and Wireless communication. - 1.8. Input/Output - 1.9. Power supply	2 - Exercise 1 Task 2: Buy a computer by components	1 - Document P1T2 - Submit P1T2
4	Topic 2: Computer components - 2.1. Actions to manage electrical and digital components. - 2.2. Tools to assembly a computer. - 2.3. Voltage levels in a computer. - 2.4. Assembly a computer. - 2.5. Tools to analyze and detect incidents.	2 - Exercise 2 Task 1: Assemble/dismantle a computer	1 - Document P2T1 - Submit P2T1
5	Topic 2: Computer components. Communications: Serial and RJ45 cables Tutorship	2+1 - Exercise 2 Task 2: Assemble serial and RJ45 cables - Solve questions	1 - Document P2T2 - Submit P2T2
6	Test 1 Electronics theory	2	1
7	Topic 3: Analysis and measurement tools. - 3.1. Power supply. - 3.2. Multimeter.	2 - Exercise 3 Task 1: Power supply and multimeter exercises	1 - Document P3T1 - Submit P3T1
8	Topic 3: Analysis and measurement tools. - Solve questions.	2 - Exercise 3 Task 1: Correction	1 - Go through P3T1
9	Topic 3: Analysis and measurement tools. - 3.3. Oscilloscope. - 3.4. Signal/Function generator. - 3.5. Others. Tutorship	2+1 - Exercise 3 Task 2: Oscilloscope and signal/function generator exercises - Solve questions	1 - Document P3T2 - Submit P3T2
10	Test 2	2	1
11	Topic 4: Digital systems.	2	1

Week	Content	Contact hours	Independent study time
	- 4.1. Microprocessor-based systems. - 4.2. Digital systems Skybot platform for development of an autonomous robot	- Presentation of the Skybot platform	- Study the documentation of the Skybot platform
12	Topic 4 Programming motors in the Skybot platform	2 - Test the motors of the Skybot platform	1 - Document P4 (Skybot y motors)
13	Topics 4 Programming the sensors in the Skybot platform Tutorship	2+1 - Test the sensors of the Skybot platform - Solve questions	1 - Document P4 (sensors) - Develop a program to follow the “black line”
14	Topic 4 Line-Following robot	2 - Test robots	1 - Document P4 - Submit P4
15	Seminar	2 - Presentations of selected topics	1 - Submit the presentation