

Course guide

300017 - PP - Programming Project

Last modified: 01/06/2023

Unit in charge:	Castelldefels School of Telecommunications and Aerospace Engineering	
Teaching unit:	701 - DAC - Department of Computer Architecture.	
Degree:	BACHELOR'S DEGREE IN NETWORK ENGINEERING (Syllabus 2009). (Compulsory subject). BACHELOR'S DEGREE IN TELECOMMUNICATIONS SYSTEMS ENGINEERING (Syllabus 2009). (Compulsory subject).	
Academic year: 2023	ECTS Credits: 6.0	Languages: Catalan, Spanish

LECTURER

Coordinating lecturer: Definit a la infoweb de l'assignatura.

Others: Definit a la infoweb de l'assignatura.

PRIOR SKILLS

Students should understand basic aspects of C# programming.

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:

1. CE 2 TELECOM. Students will acquire basic knowledge of the use and programming of computers, operating systems, databases and computer programs used in engineering. (CIN/352/2009, BOE 20.2.2009)

Generical:

4. PROJECT MANAGEMENT - Level 1: To know project management tools carrying out the different phases of the project established by the professor

6. EFFICIENT USE OF EQUIPMENT AND INSTRUMENTS - Level 1: Using instruments, equipment and software from the laboratories of general or basic use. Realising experiments and proposed practices and analyzing obtained results.

Transversal:

2. SELF-DIRECTED LEARNING - Level 1. Completing set tasks within established deadlines. Working with recommended information sources according to the guidelines set by lecturers.

3. EFFICIENT ORAL AND WRITTEN COMMUNICATION - Level 1. Planning oral communication, answering questions properly and writing straightforward texts that are spelt correctly and are grammatically coherent.

5. TEAMWORK - Level 2. Contributing to the consolidation of a team by planning targets and working efficiently to favor communication, task assignment and cohesion.

TEACHING METHODOLOGY

Students should have a personal computer (a laptop is recommended) with an Internet connection.
There is also a laptop-lending service for students who need it.

The course combines the following teaching methods:

- Independent learning: students work on independent learning materials at home.
- Cooperative learning: students work on assignments in small groups.
- Project-based learning: students work on a project in a team during the second half of the course.
- Self- and peer assessment of assignments.

LEARNING OBJECTIVES OF THE SUBJECT

On completion of the Programming Project course, students will be able to:

- Use the programming environment appropriately to build object-oriented applications with visual interfaces.
- Use appropriately data structures, advanced algorithms and dynamic memory management techniques.
- Design databases, draw up queries and use appropriately a database management system.
- Describe the features of current computer architectures.
- Present their work appropriately verbally and in writing.
- Learn independently: assimilate information from references, search for information relevant to the learning process and identify their errors.
- Work in groups, identify the tasks to be carried out, share out these tasks and integrate the results, resolve any conflicts that arise in the group and identify areas for improvement.

STUDY LOAD

Type	Hours	Percentage
Self study	84,0	56.00
Hours small group	43,0	28.67
Guided activities	23,0	15.33

Total learning time: 150 h

CONTENTS

Current computer architectures

Description:

- 1.1 Portable computers
- 1.2 Parallel computers
- 1.3 Supercomputers

Related activities:

Activity 1: Features of current computer architectures

Full-or-part-time: 13h 12m

Laboratory classes: 4h

Guided activities: 1h 42m

Self study : 7h 30m

Object-oriented and visual programming

Description:

- 2.1 The concepts of object and class: attributes and methods
- 2.2 Criteria to be used in decomposing systems into modules and objects
- 2.3 Constructors and destructors
- 2.4 Passing parameters and objects
- 2.5 Members that are members of other objects
- 2.6 Forms and events
- 2.7 Most common tests and events

Related activities:

Activity 2: Advanced Programming Project

Full-or-part-time: 60h 54m

Laboratory classes: 17h 24m

Guided activities: 9h 30m

Self study : 34h

Structure

Description:

- 3.1 Sorting algorithms
- 3.2 Search algorithms
- 3.3 Dynamic memory management
- 3.4 Recurrence and backtracking
- 3.5 Circular queues
- 3.6 Stacks

Related activities:

Activity 2: Advanced Programming Project

Full-or-part-time: 43h 18m

Laboratory classes: 11h

Guided activities: 5h 48m

Self study : 26h 30m

Databases

Description:

- 4.1 Introduction to database management systems
- 4.2 Data modelling
- 4.3 Database structure design
- 4.4 Data management and queries

Related activities:

Activity 2: Advanced Programming Project

Full-or-part-time: 7h

Laboratory classes: 3h

Guided activities: 2h

Self study : 2h

Programming environments

Description:

- 5.1 Microsoft Visual Studio programming environment
- 5.2 Project creation, classes, class libraries
- 5.3 Forms creation
- 5.4 Debugging

Related activities:

Activity 2: Advanced Programming Project

Full-or-part-time: 27h 06m

Laboratory classes: 9h 06m

Guided activities: 4h

Self study : 14h

ACTIVITIES

FEATURES OF CURRENT COMPUTER ARCHITECTURES

Description:

Directed activities and self-directed activities will involve the study of independent learning materials, individual exercises (with frequent self-assessment) and small group exercises.

Class sessions will be used to:

- Work on questions concerning the week's assignment, in small groups
- Help from the lecturer on more frequent questions.
- Several lectures on key aspects
- Individual and group exercises

Cooperative work will be a key element of the work methodology.

Specific objectives:

On completing this activity, students will be able to:

- Describe the elements and blocks in current portable computers
- Understand the main characteristics of parallel computers
- Understand the main characteristics of supercomputers and some of their applications

Material:

It consists of the following:

- Independent learning materials
- Individual and group exercises
- Detailed programme of activities and due dates

The material will be available on Atenea.

Delivery:

The activity includes a series of individual and group assignments (at least one assignment per week). Feedback will be given on these assignments.

Students must hand in at least 80% of the assignments on time to pass the course.

The assignments test basic knowledge and are subject to assessment.

Full-or-part-time: 13h 12m

Laboratory classes: 4h

Guided activities: 1h 42m

Self study: 7h 30m

ADVANCED PROGRAMMING PROJECT

Description:

Students will undertake a programming project in teams of three or four. Project-based learning methods are used, so students have to gain the knowledge required to achieve the project goals independently. Specific cooperative learning tasks will be carried out using the Jigsaw technique.

The directed learning and self-directed learning activities involve students working on:

- Independent learning materials
- Individual and group exercises
- Individual tasks for the project
- Group meetings to carry out project tasks.
- Designs and plans for project prototypes.

Classroom activities involve students working on:

- Work on questions concerning the week's assignment, in small groups
- Help from the lecturer on recurring questions.
- Several lectures on key aspects
- Individual and group exercises
- Individual tasks for the project
- Group meetings to carry out project tasks.

This activity will focus on the written and oral presentation of group assignments.

Specific objectives:

On completing this activity, students will be able to:

- Decompose the problem into objects.
- Encode sorting, search, recurrence and backtracking algorithms.
- Implementing circular stacks and queues.
- Build attractive and functional visual interfaces.

In addition, in the context of this activity students will acquire the generic skills outlined above.

Material:

It consists of the following:

- Independent learning materials
- Microsoft Visual Studio programming environment
- Individual and group exercises
- Detailed programme of activities and due dates

The material will be available on Atenea.

Delivery:

The activity includes a series of individual and group assignments (at least one assignment per week). Feedback will be given on these assignments.

Students must hand in at least 80% of the assignments on time to pass the course.

The assignments test basic knowledge and are subject to assessment.

The activity also includes three substantial assignments that make up 40% of the final mark.

Full-or-part-time: 136h 48m

Laboratory classes: 39h

Guided activities: 21h 18m

Self study: 76h 30m

GRADING SYSTEM

The assessment weighting for the course is as follows:

Assignments (20%)

Examinations (30%)

Project (40%)

Attitude and participation (10%)



EXAMINATION RULES.

In order to pass the course, students must hand in at least 80% of the assignments for the course on time. They must not have failed more than one of the core subjects.

BIBLIOGRAPHY

Basic:

- Platt, David S. Introducing Microsoft .NET. 3rd ed. Redmond: Microsoft Press, 2003. ISBN 0735619182.
- Sharp, John; Jagger, Jon. Microsoft Visual C#. NET : step by step [on line]. Redmond: Microsoft Press, 2003 [Consultation: 22/11/2023]. Available on : <https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?pq-origsite=primo&docID=7117342>. ISBN 0735619093.

Complementary:

- Ceballos Sierra, Francisco Javier. El Lenguaje de programación C#. Madrid: RA-MA, 2002. ISBN 8478975004.
- Archer, Tom; Whitechapel, Andrew. Inside C#. 2nd ed. Redmond, Washington: Microsoft Press, 2002. ISBN 0735616485.