Course guide
330219 - TP - Programming Technologies

Last modified: 28/04/2025

Unit in charge: Manresa School of Engineering
Teaching unit: 750 - EMIT - Department of Mining, Industrial and ICT Engineering.

Degree: BACHELOR'S DEGREE IN ICT SYSTEMS ENGINEERING (Syllabus 2010). (Compulsory subject).

Academic year: 2025 **ECTS Credits:** 6.0 **Languages:** Catalan

LECTURER

Coordinating lecturer: MARTA ISABEL TARRÉS PUERTAS

Others:

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:
1. Ability to understand and master the basic concepts of discrete mathematics, logic, algorithms and computational complexity as well as its application to the automatic processing of information through computer systems and the application to solving problems of engineering

2. (ENG) El coneixement de les estructures de dades més habituals i la capacitat d'usar-les de forma escaient en problemes reals. La capacitat de dissenyar estructures de dades específiques quan els problemes així ho requereixin.
3. The ability to analyze, design and maintain computer applications as well as knowledge of the principles and tools of software engineering and its application.
4. Knowledge and ability to use existing tools and instrumentation for the analysis, design, development and verification of electronic, computer and communications systems.

Transversal:
5. 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.
7. SELF-DIRECTED LEARNING - Level 1. Completing set tasks within established deadlines. Working with recommended information sources according to the guidelines set by lecturers.

TEACHING METHODOLOGY

The subject is structured in two classes of two hours per week. Of these four weekly contact hours, one is dedicated to presenting the main contents in an expository way, the second to solving problems on demand from the student body and the remaining two to solving practical problems in the computer laboratory.
The student is told weekly the study and problem solving tasks that are necessary for him to do. It is advisable to do these jobs, at least partially, working as a team. Periodically, the progress of each individual student is evaluated.
The subject also incorporates a program development project of a medium size that has to be worked on as a team.

LEARNING OBJECTIVES OF THE SUBJECT

After passing this subject the student must:
1. Know how to plan oral communication, respond appropriately to the questions asked and write basic level texts with spelling and grammar correction.
2. Know how to identify one's own information needs and use the collections, spaces and services available to design and execute simple searches appropriate to the thematic field.
3. To be able to carry out the tasks entrusted in the scheduled time, working with the indicated sources of information, in accordance with the guidelines set by the teaching staff.
4. To be able to apply the fundamental algorithmic procedures to solve problems using high-level languages.
5. Know the most common data structures and be able to use them as necessary in real problems of moderate complexity.
6. Be able to design specific data structures of medium complexity.
7. Know the concept of computational complexity and be able to calculate the complexity in time and space for the worst case of simple algorithms.
8. Know the principles and some tools for verification and validation of software and be able to apply them to real problems.
9. Be able to write simple technical reports, also in a third language, and present them orally.

STUDY LOAD

Type	Hours	Percentage
Self study	90,0	60.00
Hours small group	30,0	20.00
Hours large group	30,0	20.00

Total learning time: 150 h

CONTENTS

(ENG) TEMA 1: Classes d'objectes

Description:
Classes and Objects

Full-or-part-time: 45h
Theory classes: 10h
Laboratory classes: 10h
Self study : 25h

(ENG) TEMA 2: Recursivitat

Description:
Recursivity

Specific objectives:
Recursivity

Full-or-part-time: 23h
Theory classes: 4h
Laboratory classes: 4h
Self study : 15h

(ENG) TEMA 3: Estructures de dades

Description:
Stack and BST

Specific objectives:
Stack and BST

Full-or-part-time: 45h
Theory classes: 10h
Laboratory classes: 10h
Self study : 25h

(ENG) TEMA 4: Introducció a l'enginyeria del software

Description:
UML

Full-or-part-time: 23h
Theory classes: 4h
Laboratory classes: 4h
Self study : 15h

(ENG) TEMA 5: Complexitat algorítimica

Description:
Algorithm complexity

Full-or-part-time: 14h
Theory classes: 2h
Laboratory classes: 2h
Self study : 10h

ACTIVITIES

(ENG) ACTIVITAT 1: EXAMEN

Full-or-part-time: 2h
Theory classes: 2h

(ENG) ACTIVITAT 2: ESTUDI DE CONTINGUTS

Full-or-part-time: 25h
Self study: 25h

(ENG) ACTIVITAT 3: CLASSE EXPOSITIVA

Full-or-part-time: 12h
Theory classes: 12h

(ENG) ACTIVITAT 4: CLASSE DE PROBLEMES

Full-or-part-time: 12h
Theory classes: 12h

(ENG) ACTIVITAT 5: CLASSE DE LABORATORI

Full-or-part-time: 41h
Self study: 15h
Laboratory classes: 26h

(ENG) ACTIVITAT 6: RESOLUCIÓ DE PROBLEMES

Full-or-part-time: 30h
Self study: 30h

(ENG) ACTIVITAT 7: PROJECTE

Full-or-part-time: 28h
Self study: 20h
Theory classes: 4h
Laboratory classes: 4h

GRADING SYSTEM

The qualification is made based on 3 elements:
1. The evaluation of the autonomous work of the student (A). This component contains both the progress made in the theoretical and practical aspects. Its measurement is carried out on the basis of compulsory exercises delivered during the course.
2. The evaluation of the project (P). It is carried out from a face-to-face delivery of the project in progress that may involve a public presentation and the preparation of a report.
3. The final evaluation (F). It is done through a final exam that is global in nature and integrates all the knowledge and skills acquired during the course.
From these elements the final grade is calculated with the following weightings:

Final = 0.35A + 0.25P + 0.40F

EXAMINATION RULES.

The activities will be carried out following the uses and customs of academic work and, in particular, the following guidelines will be respected:
1. Those activities that are explicitly declared as individual, whether in person or not, will be carried out without any collaboration from other people.
2. The dates, formats and other delivery conditions that are set will be mandatory.
3. The use of the computer laboratory will be reserved exclusively for academic activities and in no case may abuse be made.

BIBLIOGRAPHY

Basic:
- Downey, A. Python for software design: how to think like a computer scientist [on line]. Cambridge: Cambridge University, 2009 [Consultation: 09/11/2020]. Available on: http://openbookproject.net/thinkcs/python/english3e/. ISBN 9780521725965.
- Miller, Bradley N.; Ranum, David L. Problem solving with algorithms and data structures using Python. Wilsonvill, OR: Franklin, Beedle & Associates, 2006. ISBN 9781590280539.
- Necaise, Rance D. Data structures and algorithms using Python. Hoboken: Wiley, 2011. ISBN 9780470618295.

Complementary:
- Metha, D. P.; Sahni, S., ed. Handbook of data structures and applications. Boca Raton, Fla: Chapman and Hall/CRC, 2005. ISBN 1584884355.
- Sommerville, I. Software engineering [on line]. 9th ed. Boston: Pearson / Addison Wesley, 2011 [Consultation: 01/06/2022]. Available on: https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?docID=5185655. ISBN 9780137053469.