Advanced Programming

Code: 100126
ECTS Credits: 6

2025/2026

Degree	Type	Year
Mathematics	OT	4

Contact

Name: Vicente Soler Ruíz

Email: vicenc.soler@uab.cat

Teaching groups languages

You can view this information at the end of this document.

Prerequisites

It is convenient to have learned the knowledge in basic programming in C in the first courses of the Degree or in other subjects, although it is not essential.

Objectives and Contextualisation

The subject has two differentparts. A first objective is to achieve a good level of programming in C, so that students will be able to structure an application and to program a complex algorithm. A second objective is to achieve a good training in object-oriented programming (mainly Python, but also using Java), creation of libraries and that serves as the basis for learning any other object-oriented language.

The practical exercises to turn in are based on implementing two games: one in C and the other in Java or Python.
The development of games as a basis for the practices of the subject provides the obligation to use many different aspects of the programming languages and makes the practical exercises more attractive for the student. All this motivates the student to be more interested in the development of the subject.

Competences

- Actively demonstrate high concern for quality when defending or presenting the conclusions of one's work.
- Develop critical thinking and reasoning and know how to communicate it effectively, both in one's own languages and in a third language.
- Distinguish, when faced with a problem or situation, what is substantial from what is purely chance or circumstantial.
- Effectively use bibliographies and electronic resources to obtain information.
- Generate innovative and competitive proposals for research and professional activities.
- Recognise the presence of Mathematics in other disciplines.
- Students must be capable of applying their knowledge to their work or vocation in a professional way and they should have building arguments and problem resolution skills within their area of study.
- Students must be capable of communicating information, ideas, problems and solutions to both specialised and non-specialised audiences.

- Students must develop the necessary learning skills to undertake further training with a high degree of autonomy.
- Students must have and understand knowledge of an area of study built on the basis of general secondary education, and while it relies on some advanced textbooks it also includes some aspects coming from the forefront of its field of study.
- Use computer applications for statistical analysis, numeric and symbolic calculus, graphic display, optimisation or other purposes to experiment with Mathematics and solve problems.
- Work in teams.

Learning Outcomes

1. Actively demonstrate high concern for quality when defending or presenting the conclusions of one's work.
2. Be able to fully or partially implement the program product and be able to design the verification and validation methods.
3. Develop critical thinking and reasoning and know how to communicate it effectively, both in one's own languages and in a third language.
4. Effectively use bibliographies and electronic resources to obtain information.
5. Generate innovative and competitive proposals for research and professional activities.
6. Have an understanding of communication machinery and systems.
7. Know how to analyse, design and implement computer-based systems in general, using methods and techniques that guarantee their efficacy and efficiency.
8. Know how to provide scientifically valid and technologically up-to-date solutions.
9. Know the different activities implied in the phases of the life-cycle of program products and their applications.
10. Master the methodologies and tools for developing information systems, database management systems and tools for automating the development of program products.
11. Students must be capable of applying their knowledge to their work or vocation in a professional way and they should have building arguments and problem resolution skills within their area of study.
12. Students must be capable of communicating information, ideas, problems and solutions to both specialised and non-specialised audiences.
13. Students must develop the necessary learning skills to undertake further training with a high degree of autonomy.
14. Students must have and understand knowledge of an area of study built on the basis of general secondary education, and while it relies on some advanced textbooks it also includes some aspects coming from the forefront of its field of study.
15. Understand the architecture and specifications for operating program products, understanding their nature and the possibilities offered by different programming languages.
16. Work in teams

Content

First part.

1. Reminder of structured and modular programming applied to C.

1.1. Basic reminder

1.1.1. Structures and data types

1.1.2. Sequence, control and iteration structures

1.1.3. Procedures and functions

1.2. Dynamic structures

1.2.1 Organization of pointers and memory adresses

1.2.2. Basic structres: stacks, lists and queues.

1.2.3. Trees

1.2.4. Dynamic matrices

2. Recursivity.

2.1. Fundamentals of recursivity

2.2. Search algorithms using trees:

2.2.1. Priority amplitude and priority depth.

2.2.2. Backtraking.

Second part

3. Object Oriented Programming (Python and Java)

3.1. General concepts and features.

3.2. Classes and objects

3.3. Object oriented analysisand design.

4. Python and Java

4.1. Language characteristics.

4.2 Classes, loops, functions and language structures

***Unless the requirements enforced by the health authorities demand a prioritization or reduction of these contents.**

Activities and Methodology

Title	Hours	ECTS	Learning Outcomes
Type: Directed			
Lectures of theory, problems and practices	45	1.8	2
Type: Autonomous			
Personal study and development of practices	97	3.88	2

Lectures:

The concepts of the subject will be presented. Emphasis will be placed on the interpretation of the results and the relationship between these concepts and their applications. Examples will be presented in order to allow students to autonomously address how to solve problems and the resolution of the proposed problems will be discussed.

The methodology of the subject will be entirely practical. This implies that classes will be taught on a computer. The theoretical concepts will be explained in each session and complemented by exercises proposed by the teacher, which will be programmed directly into the computer, under his supervision.

Tutoring:

The teacher will supervise and evaluate the student work. Students will propose the doubts that have arisen in the resolution of the proposed exercises.

Doubts will be resolved and the resolution of problems will be discussed.

Autonomous activities:

The practical assignments must be developed individually. There is no point, in this subject, to create groups of more than one student to develop the work to be turned in.

**The proposed teaching methodology may experience some modifications dependingon the restrictions to face-to-face activities enforced by health authorities.*

Annotation: Within the schedule set by the centre or degree programme, 15 minutes of one class will be reserved for students to evaluate their lecturers and their courses or modules through questionnaires.

Assessment

Continous Assessment Activities

Title	Weighting	Hours	ECTS	Learning Outcomes
Exam	33%	4	0.16	1, 3, 10, 14, 11
Practices delivery	66%	0	0	15, 9, 5, 2, 14, 13, 12, 11, 7, 8, 6, 16, 4
Resit exam	33%	4	0.16	1, 3, 10, 14, 11

Theory and problems module (weight 33%).

The evaluation of this part will be done though 2 exams: midterm and final exams.

To pass the subject, a minimum of 2.5 points must be obtained in the part of the exams. The midterm exam has a weight of 30% and the final 70% of the exams grade.

Practical module (weight 66%)
Two practical exercises will be turned in during the course. Each of these exercises will count for 33% of the final grade.

Exams

The maximum grade (Excellent with Honor) will not be awarded to students who must go to the resit exam.

The grade "Not assessable" will be applied to students who do not take any of the exams.

The resit exam is only applicable to students who have not passed the subject.

Single evaluation

Programació Avançada has not single evaluation.

**Student's assessment may experience some modifications depending on the restrictions to face-to-face activities enforced by health authorities.*

Bibliography

C

C/C++. Curso de programación. 4ª Edición.

Fco. Javier Ceballos Sierra;Fco. Javier Ceballos Sierra

eBook | 2015

JAVA

- **Learning Java : An Introduction to Real-World Programming with Java**

Marc Loy;Patrick Niemeyer;Daniel Leuck;Marc Loy;Patrick Niemeyer;Daniel Leu...
https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991010571783506709

eBook | 2020

- **Java Cookbook : Problems and Solutions for Java Developers**

Ian F. Darwin;Ian F. Darwin
https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991010571783606709

eBook | 2020

- **Foundational Java : KeyElements and Practical Programming**

David Parsons;David Parsons

eBook | 2020

https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991010871132306709

PYTHON

- **High Performance Python : Practical Performant Programming for Humans**
Micha Gorelick;Ian Ozsvald;Micha Gorelick;Ian Ozsvald

eBook | 2020

https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991000618769706709

- **Python : An Introduction to Programming**

Parker, James R.;Parker, James R.

eBook | 2021

https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991010344812006709

Guido van Rossum, "El Tutorial de Python" (Una introducción informal a Pyhon),

- Python 2: http://docs.python.org.ar/tutorial/pdfs/TutorialPython2.pdf
- Python 3: http://docs.python.org.ar/tutorial/pdfs/TutorialPython3.pdf

- Mark Lutz, "Learning Python", Ed. O'Reilly

- Raúl González Duque, "Python para todos", http://mundogeek.net/tutorial-python/

-"Python tutorial", https://www.tutorialspoint.com/python/

Other books

- "The C++ Programming Language", B.Stroustrup, 3ªed.,Addison-Wesley, 1997.

- "Java2: The Complete Reference", P.Naughton, H.Schildt, Osborne/McGraw-Hill, 2000

- "Core Java" Volumes I y II, C.S.Horstmann, G.Cornell, Sun Microsystems Press.

Software

https://www.codeblocks.org/downloads/binaries/

https://www.eclipse.org/downloads/

https://code.visualstudio.com/download

Groups and Languages

Please note that this information is provisional until 30 November 2025. You can check it through this link. To consult the language you will need to enter the CODE of the subject.

Name	Group	Language	Semester	Turn
(PLAB) Practical laboratories	1	Catalan	first semester	morning-mixed