Advanced programming

Code: 100126
ECTS Credits: 6

Degree	Type	Year	Semester
2500149 Mathematics	OT	4	0

Contact

Name: Vicente Soler Ruíz

Email: Vicenc.Soler@uab.cat

Use of Languages

Principal working language: catalan (cat)

Some groups entirely in English: No

Some groups entirely in Catalan: Yes

Some groups entirely in Spanish: No

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 training in object-oriented programming (with Java), creation of libraries and that serves as the basis for learning any object-oriented language.

The practices are based on making two games: one in C and the other in Java.
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 practices 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 ones work.
- Develop critical thinking and reasoning and know how to communicate it effectively, both in ones 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 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.

Learning Outcomes

1. Actively demonstrate high concern for quality when defending or presenting the conclusions of ones 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 ones 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. Know the different activities implied in the phases of the life-cycle of program products and their applications.
7. Master the methodologies and tools for developing information systems, database management systems and tools for automating the development of program products.
8. 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.
9. Students must develop the necessary learning skills to undertake further training with a high degree of autonomy.
10. 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.
11. Understand the architecture and specifications for operating program products, understanding their nature and the possibilities offered by different programming languages.

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:

Methodology

Lectures:

The concepts of the subject will be presented, a part of which will represent a review for some students, although the point of view will be different. Emphasis will be placed on the interpretation of the results and the relationship between these concepts and their applications. Examples will be presented that will allow students to face autonomously jow to solve problems.

The resolution of the proposed problems will be discussed on the board, whose statements students will have in advance.

The basis of the subject will be absolutely practical. This implies that classes are done entirely on the computer. The theoretical concepts will be explained in each session and will be complemented with exercises proposed by the teacher which must be programmed directly to the computer, with the supervision of the teacher.

Tutoring:

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

In the first tutoring session, the teacher will gather information on the training of each student and advise him on how to correct possible errors. In the rest of the tutoring sessions, doubts will be resolved and the resolution of problems will be discussed.

Autonomous activities:

The exercises and practices must be solved individually. It does not make any sense, in this subject, to train groups of practices of more than one student to develop the practices.

Activities

___

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

Assessment

Theory and problems module (weight 33%).

Final exam, making the student answer the theoretical concepts from programming problems.

Practices module (weight 66%)
The two practices of the subject will count each one as 33% of the final grade

Examns

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

Assessment Activities

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

Bibliography

- "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.