UAB
Universitat Autònoma de Barcelona

2023/2024

Advanced Computing

Code: 101765
ECTS Credits: 9

Degree	Type	Year	Semester
2501233 Aeronautical Management	OB	2	1

Contact

Name: Aura Hernandez Sabate

Email: aura.hernandez@uab.cat

Teaching groups languages

You can check it through this link. To consult the language you will need to enter the CODE of the subject. Please note that this information is provisional until 30 November 2023.

Teachers

Carles Ros Visus

Prerequisites

1st grade course:

1. Calculus
2. Fundamentals of Computer Science

Statistics from previous years show that, in order to follow the subject correctly, it is extremely important to have passed the Fundamentals of Computer Science. Therefore, in case of suspending Fundamentals of Computer Science, we strongly recommend not enrolling in this subject.

Objectives and Contextualisation

The main objective of this course is to help students in the process of abstraction necessary to address problems of large dimensions related to aeronautical management and provide the basic concepts for this effect. For this reason, the course deals with the treatment of data from two different points of view that converge at the same point: how information is stored and how it is manipulated.

Thus, the subject is separated into two modules. On the one hand, the introduction to databases in order to store information efficiently, and on the other hand the programming part as a natural continuation of the subject Fundamentals of Computer Science, carried out in the 1st year.

In this way, the general objectives that are proposed are the following:

1. Deepen into data structures and their abstraction through databases

2. Extract information from a database of a certain complexity
3. Provide advanced programming skills through dynamic data structures
4. Introduce the principles of object-oriented programming
5. To help students obtain the abstraction needed to separate the representation of data and its use.

The aim of this course is for students to acquire the skills to:

1. Achieve a good level of advanced programming
2. Design and implement an algorithm based on dynamic data structures and the concept of data abstraction.
3. Familiarisation with the principles of object-oriented programming
4. To know in depth a BD, from the Model E-R from some specifications of the real world that one wants to model the BD.
5. Perform simple and complex queries to a DB using SQL

Competences

- Apply specific software for solving problems in the aeronautical sector.
- Communication.
- Develop software of low or medium complexity.
- Personal work habits.
- Thinking skills.
- Use knowledge of the fundamental principles of mathematics, economics, information technologies and psychology of organisations and work to understand, develop and evaluate the management processes of the different systems in the aeronautical sector.
- Use new technologies in airline management.
- Work in teams.

Learning Outcomes

1. Accept and respect the role of the various team members and the different levels of dependence within the team.
2. Analyze the software and hardware necessary resources related to the use of data base.
3. Apply imperative programming efficiently.
4. Apply object-oriented programming.
5. Apply suitable expert systems to help in making decisions and solving problems in the aeronautical sector.
6. Communicate knowledge and findings efficiently, both orally and in writing, both in professional situations and with a non-expert audience.
7. Create applications to exploit the information stored in databases.
8. Design and implement databases of low complexity to meet the information needs of companies in the sector.
9. Develop independent learning strategies.
10. Develop scientific thinking skills.
11. Develop systemic thinking.
12. Develop the ability to analyse, synthesise and plan ahead.
13. Make efficient use of ICT in communicating ideas and results.
14. Manage time and available resources. Work in an organised manner.
15. Understand the functioning of database systems.
16. Work cooperatively.
17. Work independently.
18. Write programmes to solve problems of medium complexity in the aeronautical sector using imperative and object-oriented programming.

Content

Module 1. Programming

Topic 1. Review of Fundamentals of Computer Science

Topic 2. Introduction to object-oriented programming

- Class concept. Methods and attributes. Builders and destroyers. Data Encapsulation

Topic 3. Dynamic data structures. lists

- Basic concepts, manipulation and advanced programming

Topic 4. Dynamic data structures. Dictionaries

- Basic concepts, manipulation and advanced programming

Module 2. Database systems

Topic 5. Introduction to databases

- Definitions. Components of a Database system
- Advantages and disadvantages of a database system

Topic 6. Data models. Entity-relationship model and relational model

- Introduction
- Data structure
- Integrity rules

Topic 7. Data manipulation

- SQL

Methodology

Given that the background of the subject is the support in the process of abstraction, the work of students is the central axis of their learning, accompanied and guided by teachers. For this reason, the face-to-face classes will be highly practical and will focus on students consolidating the knowledge that is the learning objective of this subject.

The general methodology of the subject is divided into three phases:

Class preparation: the aim of this phase is for students to prepare the contents to be worked on in the following session by means of various activities offered by the teaching staff, such as watching videos, reading texts, etc.

Guided session: the aim of this phase is to consolidate the concepts seen and put them into value within the context of the subject. The teaching staff will ensure that the students will deepen their understanding of these concepts through exercises (more or less) guided during the session and each student will complete and deliver at the end of the session.

Autonomous work: in order for students to become fluent in programming and SQL queries, they will have to do part of the work on their own with individual exercises that will be handed in for assessment and two projects (one programming and one database project) that they will carry out throughout the course.

Programming and DB projects: As part of the autonomous work required of students, they will have to carry out a programming project for module 1 and a DB project for module 2. These will be related to the syllabus and will be developed throughout each of the modules. Some hours of the face-to-face sessions will be dedicated to present the work to be done and also to follow up the correct development, as well as to raise any doubts.

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.

Activities

Title	Hours	ECTS	Learning Outcomes
Type: Directed			
Guided sessions	75	3	2, 3, 4, 1, 15, 6, 7, 10, 11, 12, 8, 18, 13, 16
Type: Supervised			
Follow-up in the assimilation of theoretical concepts	10	0.4	1, 6, 10, 11, 9, 12, 13, 14
Reinforcement and follow-up in problem solving	16	0.64	3, 15, 6, 7, 10, 11, 12, 18, 13, 14
Type: Autonomous			
Database project development	21	0.84	2, 15, 6, 9, 12, 8, 14, 17
Exams preparation	10	0.4	15, 10, 11, 9, 12, 13, 14, 17
Preparation prior to guided classes	22	0.88	10, 11, 9, 12, 14, 17
Problem solving	38	1.52	2, 3, 4, 1, 15, 7, 9, 8, 18, 13, 14, 16, 17
Programming Project development	21	0.84	3, 4, 6, 7, 9, 12, 18, 14, 17

Assessment

The single assessment system is not foreseen in this subject.

The course is divided into two modules:

1. the first half of the course will be devoted to the deepening of object-oriented programming and data structures (module 1)
2. the second half of the course will be devoted to the study of database systems (module 2)

The two parts will be evaluated independently, and each module will count 50%. To obtain the final grade it will be necessary to pass each of the two parts separately (Note module> = 5).

Each of the parts of the course will be evaluated independently three types of activities and the weighted sum of them will give the final grade. These three activities are:

1. Individual Exams (EI)
2. Assessable activities (AA)
3. Practices (P)

1. The first part (EI) consists of two partial exams in which the students will be evaluated individually. The minimum grade to pass each partial exam is 5.

2. The second part (AA) will be carried out continuously throughout the course and is divided into autonomous activities proposed within the classroom sessions and autonomous activities proposed to be done outside the classroom. The final mark will be the weighted sum of the evidence requested.

3. The third part (P) will be evaluated by the delivery of a project and a validation test. The final mark will be obtained from the weighted sum of the two previous ones. The minimum mark to pass the project is 5, while the validation test must be passed with a minimum grade of 3.5. The final mark of this part must be a minimum of 5.

To pass the course it is necessary that the evaluation of each of the parts exceeds the minimum required and that the total evaluation exceeds 5 points.

GRADE ON THE TRANSCRIPT, HONOURS AND NON-ASSESSABLE

In case of passing the subject, the grade of the transcript will be the grade obtained according to the previous section. In the case of not passingthe subject due to any of the assessment activities not achieving theminimum mark required, the numerical mark on the transcript will be the minimum between a 4 and the average mark between the two theory mid-term exams. With the exception of students who:

1) do not participate in any of the assessment activities, which will be awarded the grade of "non-assessable" (any student who hands in a practical or a scheduled assessment will be awarded a grade),

2) have committed irregularities in an assessment activity, which will be given the lower value between 3.0 and the numerical mark mentioned above (and therefore a pass by compensation will not be possible).

As many registrations as possible will be given within the university regulations, as long as the final mark obtained is at least a 9.

RECOVERY

According to the Academic Regulations of the UAB in order to participate in the student's recovery must have been previously evaluated in a set of activities the weight of which is equivalent to a minimum of two thirds of the total grade of the subject. In addition, it is necessary to have obtained a minimum grade of 3in the average of the subject to be able to presentto the recovery.

EI: In the case of failing or not taking any of the individual tests, they can be made up on the day assigned to the official exam week.

AA: The evaluable exercises to be handed in, which are not solved in class, can be recovered before taking the theory exam, counting for 80% of the grade (and therefore never a grade higher than 8).

P: In the case of obtaining a grade lower than 5 in the practical part, it will be necessary to recover both the project and the corresponding validation test (this, on the day assigned to the official exam week). The grade of the recovered project cannot be higher than 8.

CONVALIDATION

From one course to the next, only the final module grades are saved, as long as they are higherthan 5.

PLAGIARISM AND VARIANTS

Without prejudice to other disciplinary measures deemed appropriate, and in accordance with current academic regulations, irregularities committed by a student that may lead to a variation in the grade will be graded with a zero (0). Assessment activities graded in this way and by this procedure will not be recoverable. If it is necessary to pass any of these assessment activities in order to pass the subject, this subject will be directly failed, with no opportunity to recover it in the same course. These irregularities include, among others
- total or partial copying of a practical, report, or any other assessment activity;
- allowing copying;

- submitting group work that has not been done in its entirety by the members of the group;
- presenting as one's own materials produced by a third party, even if they are translations or adaptations, and in general work with elements that are not original and exclusive to the student;
- having communication devices (such as mobile phones, smart watches, etc.) accessible during individual theoretical and practical assessment tests (exams).

In summary: copying, allowing copying or plagiarism in any of the assessment activities is equivalent to a SUSPENSION with a mark lower than 3.5.

COMMUNICATION

The dates of evaluations and delivery of problems will be published in the document manager Caronte (https://caronte.uab.cat/) and may be subject to possible programming changes for reasons of adaptation to possible incidents. The Charon will always be informed about these changes as it is understood that this is the usual platform for the exchange of information between teachers and students (materials, group management, deliveries, notes, communication with teachers).

In order to use it, the following steps must be taken:

1. Register as a user by giving your name, NIU, and aphoto card in JPG format. If you have alreadyregistered for another subject, there is no need to do it again, you can go to the next step.
2. Enroll in the type of teaching "Advanced Computer Science Teaching", giving as subject code "InfoAvan" (without the quotes). Even if you are a repeater, you need to re-register.

Assessment Activities

Title	Weighting	Hours	ECTS	Learning Outcomes
Evaluation of activities developed in module 1	15%	1	0.04	6, 10, 11, 9, 12, 13, 14, 16, 17
Evaluation of activities developed in module 2	15%	1	0.04	6, 10, 11, 9, 12, 13, 14, 16, 17
Final Exam (recuperation)	30%	4	0.16	2, 3, 15, 6, 14
Individual Exam module 1	15%	2	0.08	3, 6, 14
Individual Exam module 2	15%	2	0.08	2, 15, 6, 14
Practical evaluations module 1	20%	1	0.04	3, 5, 4, 1, 6, 18, 14, 16, 17
Practical evaluations module 2	20%	1	0.04	5, 1, 15, 6, 7, 8, 14, 16, 17

Bibliography

A. Silberschatz, H.F. Korth, S. Sudarshan, *Fundamentos de Bases de Datos* , 5a edició, McGraw-Hill, 2002. Disponible en línia

R. Elmasri, S. B. Navathe, *Fundamentos de Sistemas de Bases de Datos*, Addison-Wesley, 1997.

L. Joyanes Aguilar, *Fundamentos De Programación*, 4ª Ed. , McGraw-Hill, 2008.

Mark Lutz and David Ascher, *Learning PYTHON*, 2nd Edition. Safari Tech Books Online.

Milliken, C. P. (2020). *Python Projects for Beginners.* Apress.

Stueben, M. (2018). *Good Habits for Great Coding: Improving Programming Skills with Examples in Python*. Apress.

Llibres electrònics interactius de python:

- http://interactivepython.org/runestone/static/thinkcspy/toc.html#t-o-c
- http://python101.pythonlibrary.org/#
- http://www.pythontutor.com/

Software

For the programming module it will be programmed in Python and Spyder will be used as an open source programming environment. It is recommended to install it from the Anaconda platform (https://www.anaconda.com/)
Oracle's sql developer will be used for the database module (it can be downloaded for free with a student account).