

Degree programme	Type	Course
Aeronautical Management	OB	2

Contact lecturer

Name : Aura Hernandez Sabate

Email : aura.hernandez@uab.cat

Teaching staff

Carles Ros Visus

Group languages

You can consult this information at the [end](#) of the document.

Prerequisites

From the 1st year of the degree:

- Calculus
- Fundamentals of Programming I

Statistics from previous years show that, in order to successfully follow the course, it is extremely important to have passed Fundamentals of Programming I. Therefore, if you fail Fundamentals of Programming I, we strongly recommend not enrolling in this course.

Objectives

The main objective of this course is to help students in the abstraction process necessary to tackle large-scale problems related to aeronautical management and to provide them with the basic concepts for this purpose. For this reason, the course addresses data processing from two differentiated points of view that converge at the same point: how information is stored and how it is manipulated. Therefore, the course is divided into two modules. On the one hand, the introduction to databases for storing information efficiently, and on the other hand, the programming part as a natural continuation of the Fundamentals of Computing course taken in the 1st year.

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

- Provide advanced programming knowledge through dynamic data structures
- Introduce the principles of object-oriented programming
- Help students achieve the necessary abstraction to separate data representation from its use

With this course, it is intended that students acquire skills to:

- Achieve a good level of advanced programming
- Design and implement an algorithm based on dynamic data structures and the concept of data abstraction
- Know the principles of object-oriented programming

Learning outcomes

- CM01 (Work collaboratively in multidisciplinary teams to develop computer applications applied to aeronautical management.) Work collaboratively in multidisciplinary teams to develop computer applications applied to aeronautical management.
- CM02 (Critically examine the work carried out, identifying problems and proposing innovative and efficient solutions, taking into account the context of the aeronautical sector.) Critically examine the work carried out, identifying problems and proposing innovative and efficient solutions, taking into account the context of the aeronautical sector.
- KM02 (Identify the basic principles of algorithms, imperative programming, and object-oriented programming.) Identify the basic principles of algorithms, imperative programming, and object-oriented programming.
- KM03 (Distinguish simple data types and complex data structures from high-level programming languages.) Distinguish simple data types and complex data structures from high-level programming languages.
- SM07 (Perform software developments of low or medium complexity, using imperative and object-oriented programming.) Perform software developments of low or medium complexity, using imperative and object-oriented programming.
- SM09 (Use tools for programme analysis, design, coding, and debugging.) Use tools for programme analysis, design, coding, and debugging.

Contents

Topic 1. Review of Fundamentals of Programming I.

- Conditional and repetitive programming structures. Functions.

Topic 2. Introduction to Object-Oriented Programming.

- Concept of class and object. Attributes and constructor. Methods. Data encapsulation.

Topic 3. Complex data structures. Lists.

- Basic concepts, manipulation and advanced programming

Topic 4. Complex data structures. Dictionaries.

- Basic concepts, manipulation and advanced programming

Learning activities and methodology

Title	Hours	ECTS	Learning outcomes
Exams preparation	6	0.24	CM02, KM02, KM03
Problem solving	26	1.04	CM01, CM02, SM07, SM09
Monitoring the assimilation of theoretical concepts	4	0.16	CM02, KM02, KM03
Face to face sessions	50	2	CM01, CM02, KM02, KM03, SM07, SM09
Development of a programming project	28	1.12	CM01, CM02, SM07, SM09

Preparation prior to face-to-face sessions	14	0.56	KM02, KM03
Reinforcement and monitoring in problem solving	10	0.4	CM01, CM02, KM02, KM03, SM07, SM09

Given that the background of the course is the accompaniment in the abstraction process, student work is the central axis of their learning, accompanied and guided by the teaching staff. For this reason, face-to-face classes will be highly practical and will focus on helping students consolidate the knowledge that is the learning objective of this course.

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

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

Face-to-face session: the objective of this phase is to consolidate the concepts seen and put them into value within the context of the course. The teaching staff will ensure that students deepen their understanding of these concepts through (more or less) guided exercises during the session, which each student will complete and submit at the end of the session.

Autonomous work: so that students gain fluency in advanced programming, they will have to do part of the work on their own with individual exercises to be submitted for assessment and a programming project that will be developed throughout the entire course.

Programming project: Within the autonomous work required of students, they will need to complete a programming project. This will be related to the course content and students will develop it throughout the course. Some hours of the face-to-face sessions will be dedicated to setting out the work to be done and also to monitoring the correct development, as well as raising any questions.

Note: 15 minutes of a class will be reserved, within the calendar established by the centre/degree programme, for students to complete the teaching staff performance evaluation surveys and the course evaluation surveys.

COMMUNICATION

Assessment dates and submission of activities will be published on the virtual campus and may be subject to possible schedule changes due to adaptation to possible incidents. Information about these changes will always be communicated via the virtual campus, as this is understood to be the usual information exchange platform between teaching staff and students (materials, group management, submissions, grades, communication with teaching staff). Information about exam review dates once grades have been posted will also be communicated through this channel.

Exam reviews will be held in person or through the assessment platform on the dates proposed by the teaching team.

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

Continuous assessment activities

Title	Weight	Hours	ECTS	Learning outcomes
Final Assessment (recovery)	40%	3	0.12	CM02, KM02, KM03, SM07
First Assessment	20%	2	0.08	CM02, KM02, KM03, SM07
Second assessment	20%	2	0.08	CM02, KM02, KM03, SM07

Assessment of activities developed in the classroom	20%	2	0.08	CM02, SM09
Assessment of activities developed outside the classroom	10%	1	0.04	CM02, SM09
Assessment of practical work (project + individual validation)	30%	2	0.08	CM01, CM02, KM02, KM03, SM07, SM09

This course does not include a single assessment system.

Three types of activities will be assessed independently and their weighted sum will give the final grade. These three activities are:

- Individual partial exams (IE)
- Assessable Activities (AA)
- Practical work (P)

1. The first part (IE) consists of two partial exams in which students will be assessed individually. The first partial exam will count for 30% and the second for 70%. The minimum grade to pass the second partial exam is 5 and the weighted average of the two must be at least 5.

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

3. The third part (P) will be assessed with the submission of a project (in pairs) and the assessment of an individual validation test. The final grade will be obtained from the weighted sum of the two previous grades. The minimum grade to pass the project is 5, while the validation test must be passed with a minimum grade of 3.5. The final grade for this 3rd part must be at least 5.

To pass the course, it is necessary that the assessment of each part exceeds the required minimum and that the total assessment exceeds 5 points.

OLD PLAN

Students who remain on the old plan and are retaking advanced computing will have to pass the two modules (programming and databases) separately. Each module will count for 50% and to obtain the final grade they will have to pass each of the two parts separately (Module grade ≥ 5). The scheduling of the programming part (and therefore also the assessment) will be that of the new course. For databases, a project will be proposed at the beginning of the course and a partial exam will be held in January (on the same day as the programming partial exam), together with the project validation test. The grade for this module will be obtained from 50% of the partial exam and 50% of the practical work.

TRANSCRIPT GRADE, HONOURS AND NOT ASSESSABLE

If the course is passed, the transcript grade will be the grade obtained according to the previous section. If the course is not passed because one of the assessment activities does not reach the required minimum grade, the numerical transcript grade will be the minimum between 4 and the grade of the second theory partial exam. With the exception of students who:

- do not participate in any of the assessment activities, who will be given the grade of "not assessable" (any student who submits a practical assignment or a scheduled assessment will have a grade),
- have committed irregularities in an assessment act, who will be given the lesser value between 3.0 and the numerical grade mentioned above (and therefore compensation passing will not be possible).

As many honours distinctions as possible will be awarded within university regulations, provided the final grade obtained is at least 9.

RESIT

In accordance with the UAB Academic Regulations, to be able to participate in the resit, the student must have previously been assessed in a set of activities whose weight is equivalent to a minimum of two-thirds of the total course grade. In addition, a minimum grade of 3 must have been obtained in the course average to be able to sit the resit.

IE: If a student fails or does not sit one of the individual exams, they will have to take the resit exam on the day assigned during the official exam week.

AA: Assessable exercises to be submitted, other than those proposed to be done during sessions, can be resat before the theory exam, counting for 80% of the grade (and therefore never a grade above 8). Those proposed in the classroom will not be resittable.

P: If a grade below 5 is obtained in the practical part, both the project and the corresponding validation test must be resat (the latter, on the day assigned during the official exam week). The grade of the resat project cannot exceed 8.

For students on the old plan, the resit will work the same way, but applies to each module separately.

VALIDATION

There are no validations.

For students on the old plan, only final module grades are kept, provided they are above 5.

PLAGIARISM AND IRREGULARITIES

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 resittable. If it is necessary to pass any of these assessment activities to pass the course, the course will be directly failed, without the opportunity to recover it in the same academic year.

These irregularities include, among others:

- total or partial copying of a practical assignment, report, or any other assessment activity;
- allowing others to copy;
- submitting group work not done entirely by the group members;
- presenting as one's own materials produced by a third party, even if they are translations or adaptations, and in general work with elements not original and exclusive to the student;
- having communication devices (such as mobile phones, smart watches, etc.) accessible during individual theoretical-practical assessment tests (exams);
- using generative artificial intelligence to replace the student's learning activity.

In summary: copying, allowing others to copy or plagiarizing in any of the assessment activities is equivalent to a FAIL with a grade below 3.5.

USE OF AI

For this course, the use of Artificial Intelligence (AI) technologies is permitted exclusively for tasks supporting the student; under no circumstances may it be used to replace the student's learning activity. Assessment tasks suspected of having been generated by these techniques will be graded with a 0.

Bibliography

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.

Interactive electronic books on Python:

- <http://python101.pythonlibrary.org/#>
- <http://www.pythontutor.com/>

Software

The code will be written 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/>).

Course groups and languages

The information provided is provisional until November 30. After this date, you will be able to consult the language of each group through this [link](#). To access the information, you will need to enter the course CODE

Type of teaching	Group	Language	Semester	Shift
(PAUL) Classroom practices	11	Catalan	first semester	afternoon
(PLAB) Practical laboratories	11	Catalan	first semester	afternoon
(PAUL) Classroom practices	12	Catalan	first semester	afternoon
(PLAB) Practical laboratories	12	Catalan	first semester	afternoon