UAB
Universitat Autònoma de Barcelona

Introduction to Problem Solving and Algorithm Design

Code: 102151
ECTS Credits: 6

2025/2026

Degree	Type	Year
Business and Information Technology	OB	1

Contact

Name: Maria Gloria Estape Dubreuil

Email: gloria.estape@uab.cat

Teachers

(External) David de Palma

Teaching groups languages

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

Prerequisites

Being a subject taught in the first semester of the degree, no specific prior knowledge is needed.

Objectives and Contextualisation

This is a first semester course, designed to familiarize students with tasks involved in tackling real-life problem solving. This will be a significant responsibility for many graduates in Business and Information Technology, particularly in the areas related with decision-making and business information management.

In this sense, the basic aims of the course are:

- To be able to describe the essential elements summarizing a problem, as well as the basic strategies that can be adopted for solving it.
- Understand the role of different components of the problem both in its contextualization and definition, and in making subsequent decisions.
- Identify the characteristics of situations and problems which resolution can be "automated" by means of algorithms.
- Understand the basic structures that can be used in algorithm design.
- To be familiar with some of the most common algorithms used in a business environment.

After completing the course, students should be able to address the basic modeling of a situation and use

specific skills for solving problems. In particular, they should be able to present and argue solutions (and features taken into account in decision-making) for initially poorly defined problems in which sophisticated quantitative tools are not required.

This course provides the basics for others in the degree, including Fundamentals of Programming, taken in the second semester of this first year.

Learning Outcomes

1. CM28 (Competence) Use basic structured programming structures to solve business problems.
2. KM14 (Knowledge) Identify algorithm and program analysis techniques.
3. SM26 (Skill) Coding algorithms using structured programming techniques.
4. SM27 (Skill) Identify the bases of learning algorithms for solving problems related to management in companies and organisations.

Content

The course is divided in two parts, each one with two units. The first part deals with the conceptualization of complex problems; the second one focuses on solving well-structured problems through algorithmic techniques.

Unit 1. The art of problem solving

This first unit aims at presenting the various elements needed in the analysis of a "problematic situation". It also considers methodologies to obtain a solution satisfying given criteria or other criteria formulated during the analysis itself. Particularly, we will deal with:

- Basic methodologies to understand the background of a problem, including graphic analysis and creative thinking. We will also consider problems arising from preconceptions and mechanisms to effectively establishing working assumptions.
- Various definitions of solution according to the scope and possibilities of action.
- The role of different approaches to problem solving, such as simulation or the use of methodologies that are considered "typical" of other disciplines.

Unit 2. Problem solving and decision making

The unit introduces the role of modelling for solving "complex" problem situations. Building models should allow "informed" choices, based on a scientific analysis of the consequences of such decisions. In particular, the following items will be considered:

- Components of the analysis of "complex" situations: definition of a suitable model; role of the various actors; determination of external factors and controllable elements.
- Cause and effect analysis: the solution of a problem considered as an element of change in the original situation.
- Decision making and validation of the "best" solution to a given situation.

Unit 3. Basic algorithmic structures as an approach to "automatic" problem solving

The aim of the unit is the introduction of the main elements describing analgorithm and its use in different contexts. In particular:

- The concept of algorithm and its application to various fields of work
- Methods of specifying an algorithm: flowcharts, pseudocode and programming languages
- Linear algorithms.

- Non-linear control structures: alternative, iterative and recurrent schemes.
- Applications to the determination of algorithmic solutions to simple problems.

Unit 4. Towards good algorithm design

The unit offers a more in-depth knowledge on algorithm design, providing some basic tools to enhance simple "functioning" algorithms. We will consider how to write easy-to-follow well-structured algorithms, having desirable qualities: correctness, generality, simplicity, efficiency.

Activities and Methodology

Title	Hours	ECTS	Learning Outcomes
Type: Directed			
D1. Theoretical lectures	27.5	1.1	CM28, KM14, SM26, SM27, CM28
D2. Classroom problems	10	0.4	KM14, SM26, KM14
D3. Laboratory classes	10	0.4	CM28, KM14, SM26, SM27, CM28
D4. Oral presentations	2	0.08	SM27, SM27
Type: Supervised			
Tutoring and monitoring work in progress	16.5	0.66	CM28, KM14, SM26, SM27, CM28
Type: Autonomous			
A1. Independent study	30	1.2	CM28, KM14, SM26, SM27, CM28
A2. Teamwork in case studies	25	1	CM28, SM26, CM28
A3. Writing reports on resolution proposals or solution implementations	25	1	CM28, SM27, CM28

The teaching methodology focuses on problem-based learning (PBL). Thus, the path followed in analyzing and solving "particular" problems will be the fundamental axis on which the contents of the subject will be unfolded. During the semester, a set of multiple different problems will be presented and analyzed. Teachers will have a mediation and orientation role in this approach. Discussions of the problems will be used as tools, carried out either in small groups or in the whole group-class. It is important to keep in mind that this methodology emphasizes the student, who should be the protagonist of his own learning process.

The two parts in which the syllabus is divided (a "macro" approach in the first two topics and a "micro" approach in the last two) will be developed in parallel in time. This simultaneity is possible due largely to the classroom problems, and to the reduced number of students (up to 25) attending a given laboratory class. The basic purpose throughout will be to deepen the student's abilities in the design and development of specific algorithms to solve assorted situations.

To model several of the problems presented in the course, the structuring possibilities offered by spreadsheets will be used. Their simulation capabilities in the analysis of different alternative solutions will be explained, taking advantage of the tools offered in terms of design and calculation.

Being an introductory course, algorithm design will be supported by vBA, Microsoft Office Visual Basic for Applications, based on ©Excel files, and will include only basic tools. The ease of programming of this language (occasionally even excessive) is combined with the versatility of its application to business and organization environments.

The discussion on the methodologies used in the different cases proposed, as well as their possibilities of extension to other problems, will also constitute a significant part of the course. When necessary, theoretical lectures will also be introduced, aiming to elucidate both concepts and specific techniques.

Some of the problems proposed during the course will be worked on in teams of 4 or 5 students, which will be organized in a stable way throughout the semester. They will be the basis of the different laboratory practices during the course, and an important component of the evaluation of the subject. Nevertheles, it is important to keep in mind that the learning process must be individual, and that the task of modeling, solving and later on generalizing, initiated within the direct activities during the course, must be complemented with individual study.

In this subject the use of Artificial Intelligence (AI) technologies will not be allowed in most of the course activities, as it would hinder the understanding of the scope and use of basic programming tools, which can be used in various equally correct ways depending on the approach and reasoning used by each student. The inclusion of AI-generated fragments in any deliverable will be considered a lack of academic honesty and may result in a partial or total penalty in the grade of the activity, or greater penalties in cases of recurrence.

However, in the last laboratory practice of the course, and exclusively for the purpose of obtaining complementary tools for the final project, a limited use of AI technologies will be allowed. Teams choosing to use them must clearly identify both the tools and its specific use and include a critical reflection on how they have influenced the process and the result of the activity. Lack of transparency in the use of AI in this assessable activity will be considered a lack of academic honesty and may result in a partial or total penalty in the grade of the activity, or greater penalties in cases of seriousness.

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
a. Written submissions of solved problems and laboratory case studies	37%	0	0	CM28, KM14, SM26, SM27
b. Individual presentations of case-analysis and results; oral contributions to the course	18%	0	0	SM26, SM27
c. Written and Lab exams	45%	4	0.16	CM28, KM14, SM26, SM27

The assessment of the course will be formative, thus carried out throughout the semester. It is based on the following learning evidences:

1. Written submission of the description of the solving process for some of problems. The basic objective of these tasks is both to explain the process of reasoning followed and to solidly argue the validity of the proposed solution.

2. Results of the laboratory case-studies programmed during the semester.

3. Oral presentations of case-analysis and results obtained. The students' oral competence is modulated through planned presentations worked out in small groups.

4. Oral contributions discussing problems during the course, thus encouraging the active participation of the students in the problem modeling and solving procedures, one of the key elements of their learning process.

5. Written exams, a midle-term one and the final exam at the end of the course. The aim is to encourage the individual consolidation of the techniques and resolution processes studied.

The students' final grade is obtained from the weighted sum of the grades of the various evidences, with each one of the components having a different specific weight. The following calculation will be used:

N = 37% (written submission of solved problems and laboratory case studies) + 18% (individual oral and written presentations of case-studies) + 45% (individual written exams)

subject to the two following constraints: (1) each one of the four components of the assessment must be a strictly positive value, and (2) the average score obtained in the individual written exams should be at least 4.5 (out of 10).

Students having not passed the subject, i.e. with a grade less than 5.0 or not having met the above conditions to be graded, will be eligible for the *retakeprocess* described below if one of the following situations apply:
(a) They met conditions (1) and (2) above, and the total grade obtained (N) falls between 3.5 and 5 (3.5 <= N <5); or (b) The set of individual written exams has a grade of less than 4.5, but the rest of the student's marks are higher enough such that if the average were calculated, a mark of 5 or higher would be obtained.

Note:

1. Only students having not participated in any of the assessment activities will be considered "Not evaluable".

2. This subject does NOT offer the option for comprehensive evaluation.

Calendar of evaluation activities

The dates of the evaluation activities (exercises, assignments ...) will be announced well in advance during the semester.

The dates of the midterm and final exams are scheduled in the assessment calendar of the Faculty. It must be taken into account that:

- *"The dates of evaluation activities cannot be modified, unless there is an exceptional and duly justified reason why an evaluation activity cannot be carried out. In this case, the degree coordinator will contact both the teaching staff and the affected student, and a new date will be scheduled within the same academic period to make up for the missed evaluation activity."* **Section 1 of Article 115. Calendar of evaluation activities (Academic Regulations UAB).**
- Students who, in accordance with the previous paragraph, need to change an evaluation activity date must process the request by filling out an Application for exams' reschedule at https://eformularis.uab.cat/group/deganat_feie/application-for-exams-reschedule

Grade revision process

After all grading activities have ended students will be informed of thedate and way in which the course grades will be published. Students will bealso be informed oftheprocedure, place, dateand time of grade revision following Universityregulations.

<u>Retake Process</u>

"To be eligible to participate in the retake process, it is required for students to have been previously been evaluated for at least two thirds of the total evaluation activities of the subject." Section 3 of Article 112 ter. The recovery (UAB Academic Regulations). Additionally, it is required that the student to have achieved an average grade of the subject between 3.5 and 4.9.

The date of the retake exam is posted in the calendar of evaluation activities of the Faculty. Students taking this exam and passing will get a grade of 5 for the subject. The students having not passed the retake will be graded using the mean of their individual tests, and hence, will fail the course.

<u>Irregularities in evaluation activities</u>

Despite other disciplinary measures deemed appropriate, and in accordance with current academic regulations, *"whenever a student makes any irregularity that could lead to a significant variation in the grade of an evaluation activity, it will be graded with a 0, regardless of the disciplinary process that can be instructed. In case of occurrence of various irregularities in the evaluation of the same subject, the final grade of this subject will be 0".* **Section 10 of Article 116. Results of the evaluation. (UAB Academic Regulations).**

Bibliography

The following manuals may serve as a reference to the various parts of the course:

Brassard, G. - Bratley, P. (2008). *Fundamentos de algoritmia*. Pearson-Prentice Hall, Madrid.

Lee, R.C.T. - Tseng, S.S. - Chang, R.C. - Tsai, Y.T. (2007). *Introducción al diseño de algoritmos: un enfoque estratégico.* McGraw-Hill, Madrid.

Furthermore, also:

Adair, J. (2010). *Decision Making and Problem Solving Strategies* (4th. ed.). Kogan Page.

Levitin, A. (2007). *Introduction to The Design and Analysis of Algorithms* (2nd. ed.). Pearson.

Mansfield, Richard. *Mastering VBA for Microsoft Office 2007*, John Wiley & Sons, Incorporated, 2008. *ProQuest Ebook Central*, https://ebookcentral.proquest.com/lib/UAB/detail.action?docID=3057128.

Supplementary material will be added to the website of the subject at the Virtual Campus of the University if deemed necessary. This same site will be the reference of the course in all aspects, including the publication of problems, case studies and further information concerning the course. It will also be used to display the configuration of the student groups, to submit their work, as well as to communicate the grading process and results.

Software

We shall use MS Excel and vBA.

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
(PAUL) Classroom practices	201	Catalan	first semester	morning-mixed
(PAUL) Classroom practices	202	Spanish	first semester	morning-mixed
(PLAB) Practical laboratories	201	Catalan	first semester	morning-mixed
(PLAB) Practical laboratories	202	Catalan	first semester	morning-mixed
(PLAB) Practical laboratories	211	Catalan/Spanish	first semester	morning-mixed
(PLAB) Practical laboratories	212	Catalan/Spanish	first semester	morning-mixed
(PLAB) Practical laboratories	213	Catalan/Spanish	first semester	morning-mixed
(TE) Theory	20	Catalan	first semester	morning-mixed