Universidad de Alcalá

TEACHING GUIDE

Algorithms and Complexity

Degree in
Computer Science Engineering (GII)
Computer Science Engineering and Business Management and Administration (GII-ADE)

Universidad de Alcalá

Academic Year 2025/2026

2nd Year - 2nd Semester (GII)

1er Curso - 2nd Semester (GII-ADE)

Approved by the EPS School Board on June 16th, 2025

TEACHING GUIDE

Course Name:	**Algorithms and Complexity**
Code:	**780021 (GII+GII-ADE)**
Degree in:	**Computer Science Engineering (GII)** **Computer Science Engineering and Business Management and Administration (GII-ADE)**
Department and area:	**Ciencias de la Computación** **Computer Science**
Type:	**Compulsory (GII+GII-ADE)**
ECTS Credits:	**6.0**
Year and semester:	**2nd Year - 2nd Semester (GII)** **1er Curso - 2nd Semester (GII-ADE)**
Teachers:	Por definir
Tutoring schedule:	Consultar al comienzo de la asignatura
Language:	English

1. COURSE SUMMARY

Algorithm theory is one of the pillars of programming and its relevance is shown in the development of any application, beyond the mere construction of programs. This subject deals with the analysis and design of algorithms, and aims to expose the student to the necessary techniques for its design and implementation, as well as to present the tools that allow it to measure its effectiveness and efficiency.

The objective of the subject is not to provide the student with known solutions to specific problems, but to provide the student with the basic algorithmic techniques that will allow him to approach the development of correct and efficient programs to solve non-trivial problems, studying in the development of the subject the techniques and patterns of design of the most important algorithms (families of algorithms), and deepening in the design and evaluation of the algorithms themselves.

Prerequisites and Recommendations

Because the subject presents different techniques and advanced data structures for solving problems, the student must know and have practice in imperative programming, and especially in the use of recursion. Therefore, it is recommended that the student has successfully passed the subjects of Fundamentals of Programming and Data Structures.

Regarding the mathematical part of the subject, the student should be familiar with the concepts of limit, combinatorial and resolution systems of equations, so the subjects of Mathematical Fundamentals and Discrete Mathematics should have been successfully completed

2. SKILLS

Basic, Generic and Cross Curricular Skills.

This course contributes to acquire the following basic, generic and cross curricular skills:

en_CG8 - Knowledge of the basic subjects and technologies, which enable them to learn and develop new methods and technologies, as well as those that provide them with great versatility to adapt to new situations.

en_CG9 - Ability to solve problems with initiative, decision making, autonomy and creativity. Ability to know how to communicate and transmit the knowledge, skills and abilities of the profession of Computer Engineering Engineer.

en_CB1 - That students have demonstrated to possess and understand knowledge in an area of study that is based on general secondary education, and is usually found at a level that, although supported by advanced textbooks, also includes some aspects that involve knowledge from the forefront of their field of study.

en_CB2 - That the students know how to apply their knowledge to their work or vocation in a professional manner and possess the competencies that are usually demonstrated through the elaboration and defense of arguments and the resolution of problems within their area of study.

en_CB3 - That students have the ability to gather and interpret relevant data (usually within their area of study) to make judgments that include a reflection on relevant social, scientific or ethical issues.

en_CB4 - That students can transmit information, ideas, problems and solutions to both a specialized and non-specialized public.

en_CB5 - That the students have developed those learning skills necessary to undertake further studies with a high degree of autonomy.

en_TRU1 - Capacity of analysis and synthesis.

en_TRU2 - Oral and written competencies.

en_TRU3 - Ability to manage information.

en_TRU4 - Autonomous learning skills.

en_TRU5 - Team work.

Specific Skills

This course contributes to acquire the following specific skills:

en_CC1 - Ability to have a thorough knowledge of the fundamental principles and models of computing and know how to apply them to interpret, select, evaluate, model, and create new concepts, theories, uses and technological developments related to computer science.

en_CC3 - Ability to evaluate the computational complexity of a problem, know algorithmic strategies that can lead to its resolution and recommend, develop and implement the one that guarantees the best performance according to the established requirements.

en_CC4 - Ability to know the fundamentals, paradigms and techniques of intelligent systems and analyze, design and build systems, services and computer applications that use these techniques in any field of application.

Learning Outcomes

After succeeding in this subject the students will be able to:

LO1. Reason the theoretical foundations of algorithmic schemes.

LO2. Learn to perform the analysis of an algorithm, taking into account its efficiency, cost, cases, asymptotic notation, etc., understanding it as a quality criterion.

LO3. Apply with criteria the basic techniques of efficiency analysis and algorithm design.

LO4. Analyze the correctness of an algorithm using simple verification techniques.

LO5. Recognize the basic algorithmic schemes: divide and conquer, voracious, dynamic programming, backtracking, ramification and dimensioning algorithms.

LO6. Know some classical algorithms for fundamental problems, and recognize situations where you can reuse the repertoire of data structures and classical algorithms.

LO7. Write recursive programs about types that are not necessarily basic and be able to reason about their correctness and efficiency.

LO8. Be able to use advanced algorithm design and analysis techniques, and know how to customize general algorithmic schemes to solve problems.

LO9. Have criteria that allow, during the stages of specification, design and implementation, to choose the most appropriate alternative, and have elements to argue in a reasoned manner the choices made.

LO10. Take contact with some fundamental techniques of design and analysis of algorithms, as well as with some advanced programming techniques.

LO11. Know how to identify the most relevant components of a problem and select the most appropriate algorithmic technique for resolution.

LO12. Be able to select the most suitable types of data to improve the efficiency of an algorithmic solution.

LO13. Justified design of efficient algorithmic solutions for problems typical of the academic level of the subject.

3. CONTENTS

Contents Blocks	Total number of hours
Introduction to algorithm theory Introduction to algorithm theory	12 hours
Basic algorithms Voracious algorithms Divide and conquer algorithms Algorithms of dynamic programming	24 hours
Advanced algorithms Use of intensive recursion	14 hours
Non-determinism and complexity classes Non-determinist algorithms Complexity	6 hours

4. TEACHING - LEARNING METHODOLOGIES. FORMATIVE ACTIVITIES.

4.1. Credits Distribution

Number of on-site hours:	60 hours (56 hours on-site + 4 exams hours)
Number of hours of student work:	90
Total hours	150

4.2. Methodological strategies, teaching materials and resources

Lectures in combination with practices in the laboratory	Exhibition and discussion of the basic knowledge of the subject. Theoretical approach and resolution of exercises and related assumptions. Oriented to the teaching of the specific competences of the subject, especially those related to the techniques of creation and improvement of algorithms.
Cooperative work in groups	Approach and development of practical exercises that contribute to the understanding of the subject and the development of practice in problem analysis, critical reasoning and understanding of resolution methods.
Personal work and study	Analysis and assimilation of the contents of the subject, problem solving, bibliographic consultation, preparation of individual and group work, conducting face-to-face exams and self-evaluations. Especially oriented to the development of methods for self-organization and planning of individual and team work.

5. ASSESSMENT: procedures, evaluation and grading criteria

Preferably, students will be offered a continuous assessment model that has characteristics of formative assessment in a way that serves as feedback in the teaching-learning process.

5.1. PROCEDURES

The evaluation must be inspired by the criteria of continuous evaluation (Learning Assesment Guidelines, LAG, art 3). However, in compliance with the regulations of the University of Alcalá, an alternative process of final evaluation is made available to the student in accordance with the Learning Assesment Guidelines as indicated in Article 10, students will have a period of fifteen days from the start of the course to request in writing to the Director of the Polytechnic School their intention to take the non-continuous evaluation model adducing the reasons that they deem convenient. The evaluation of the learning process of all students who do not apply for it or are denied it will be done, by default, according to the continuous assessment model. The student has two calls to pass the subject, one ordinary and one extraordinary.

Ordinary Call

Continous Assessment:

The main assessment tools will be:

1.-**Intermediate evaluation test (IET1)** Of a written nature, consisting of solving problems and issues on thematic blocks 1 and 2.

2.-**Intermediate evaluation test (IET2)** Of a written nature, consisting of the resolution of problems and questions on thematic blocks 3 and 4

3.-**Intermediate laboratory evaluation test (ILET1),** consisting of the delivery and defense of a series of exercises related to the knowledge taught in laboratory class, as a practical application of them.

Assessment through final exam:

In the case of evaluation by means of a final exam, the evaluation elements to be used will be the following:

1.-**Final evaluation test (FET)** Written, consisting of solving problems and questions about the four thematic blocks of the subject and their laboratory practices, as well as the eventual delivery and oral defense of a work of laboratory.

Extraordinary Call

The procedure will be the same as that described for the assessment by means of a final exam in the ordinary call.

5.2. EVALUATION

EVALUATION CRITERIA

The assessment criteria measure the level in which the competences have been acquired by the student. For that purpose, the following are defined:

AC1. Understanding and use of basic concepts and techniques about algorithms, design, analysis, efficiency, and cost of algorithms

AC2. Understanding and using the elements, techniques and use of basic algorithms, their data structures, basic strategies, voracity, usual efficiencies, reuse of cases and dynamic programming.

AC3. Understanding and using techniques of intensive recursion, advantages and disadvantages, problems of branching and pruning.

AC4. Understanding and use of non-determinism and probabilistic methods, validity of results, particular advantages and disadvantages, as well as the fundamental analysis of the problems of complexity and the algorithmic irresolubility of certain problems.

AC5. Ability for practical application in programmed practical laboratory cases

GRADING TOOLS

The work of the student is graded in terms of the assessment criteria above, through the following tools:

1. Ordinary call
    a. Intermediate evaluation test (IET1) Of a written nature, consisting of solving problems and issues on thematic blocks 1 and 2.
    b. Intermediate evaluation test (IET2) Of a written nature, consisting of the resolution of problems and questions on thematic blocks 3 and 4
    c. Intermediate laboratory evaluation test (ILET1), consisting of the delivery and defense of a series of exercises related to the knowledge taught in laboratory class, as a practical application of them.
    d. Final evaluation test (FET) Written, consisting of solving problems and questions about the four thematic blocks of the subject and their laboratory practices, as well as the eventual delivery and oral defense of a work of laboratory.
2. Extraordinary call. Final assessment (FET)

GRADING CRITERIA

In the ordinary call-continuous assessment the relationship between the competences, learning outcomes, criteria and evaluation instruments is as follows.

Skill	Learning Outcomes	Evaluation criteria	Grading Tool	Contribution to the final mark
CG8, CG9 CC1, CC3	LO1-7, LO11	AC1-4	IET1	30%
CG8, CG9 CC1, CC3, CC4	LO8-13	AC1-4	IET2	35%
CG8, CG9 CC1, CC3, CC4	LO1-13	AC5	IELT1	35%

In the ordinary call-final evaluation, the relationship between the competences, learning outcomes, criteria and evaluation instruments is as follows.

Skill	Learning Outcomes	Evaluation criteria	Grading Tool	Contribution to the final mark
CG8, CG9 CC1, CC3, CC4	LO 1-13	AC1-4	FET	100%

Extraordinary call

In the case of the extraordinary call, the same percentages that have been established in the case of the evaluation by final exam will be maintained, giving the option of taking the IELT1 or maintaining the grade obtained in the continuous evaluation, according to the student's decision.

The teaching-learning methodology and the assessment process will be adapted as needed, in accordance with the guidelines of the Diversity Support Unit, to implement curricular adaptations for students with specific needs.

6. BIBLIOGRAPHY

6.1. Basic Bibliography

There are versions in English of the recommended bibliography. However, the Spanish translations are listed in this section because they are the ones available at the library.

• A.V. Aho, J.E. Hopcroft, J.D. Ullman, Estructuras de datos y algoritmos, Addison-Wesley Iberoamericana, 1988.

• G.Brassard, P. Bratlet, Fundamentos de algoritmia, Prentice Hall, 1999.

• G.H. Gonnet, R. Baeza-Yates, Handbook of Algorithms and Data Structures, AddisonWesley, 1991.

• M.A. Weiss, Estructuras de datos y algoritmos, Addison-Wesley Iberoamericana, 1995.

6.2. Additional Bibliography

• T. Cormen, C. Leirserson, R. Rivest, Introduction to Algorithms, McGraw Hill, 1998.

• R. Peña, Diseño de Programas. Formalismo y Abstracción. Prentice Hall, 1997.

• N. Wirth, Algoritmos + Estructuras de datos = Programas. Editorial del Castillo, 1984.

• N. Ziviani, Diseño de algoritmos con implementaciones en Pascal y C, Thompson, 2007.

Disclosure Note

During the evaluation tests, the guidelines set out in the Regulations establishing the Rules of Coexistence of the University of Alcalá must be followed, as well as the possible implications of the irregularities committed during said tests, including the consequences for committing academic fraud according to the Regulation of Disciplinary Regime of the Students of the University of Alcalá.