

## Course guide

### 200162 - ALGO - Algorithmics

Last modified: 11/04/2024

<b>Unit in charge:</b>	School of Mathematics and Statistics		
<b>Teaching unit:</b>	723 - CS - Department of Computer Science.		
<b>Degree:</b>	BACHELOR'S DEGREE IN MATHEMATICS (Syllabus 2009). (Compulsory subject).		
<b>Academic year:</b> 2024	<b>ECTS Credits:</b> 7.5	<b>Languages:</b> Catalan	

#### LECTURER

---

<b>Coordinating lecturer:</b>	SALVADOR ROURA FERRET
<b>Others:</b>	Primer quadrimestre: ALBERT ATSERIAS PERI - M-A, M-B AMALIA DUCH BROWN - M-A, M-B ENRIC RODRIGUEZ CARBONELL - M-A, M-B SALVADOR ROURA FERRET - M-A, M-B

#### DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

---

##### Specific:

1. CE-2. Solve problems in Mathematics, through basic calculation skills, taking in account tools availability and the constraints of time and resources.
2. CE-3. Have the knowledge of specific programming languages and software.
3. CE-4. Have the ability to use computational tools as an aid to mathematical processes.

##### Generical:

4. CB-1. Demonstrate knowledge and understanding in Mathematics that is founded upon and extends that typically associated with Bachelor's level, and that provides a basis for originality in developing and applying ideas, often within a research context.
5. CB-2. Know how to apply their mathematical knowledge and understanding, and problem solving abilities in new or unfamiliar environments within broader or multidisciplinary contexts related to Mathematics.
6. CB-3. Have the ability to integrate knowledge and handle complexity, and formulate judgements with incomplete or limited information, but that include reflecting on social and ethical responsibilities linked to the application of their knowledge and judgements.
7. CG-1. Show knowledge and proficiency in the use of mathematical language.
8. CG-2. Construct rigorous proofs of some classical theorems in a variety of fields of Mathematics.
9. CG-3. Have the ability to define new mathematical objects in terms of others already know and ability to use these objects in different contexts.
10. CG-4. Translate into mathematical terms problems stated in non-mathematical language, and take advantage of this translation to solve them.
12. CG-6 Detect deficiencies in their own knowledge and pass them through critical reflection and choice of the best action to extend this knowledge.

##### Transversal:

11. SELF-DIRECTED LEARNING. Detecting gaps in one's knowledge and overcoming them through critical self-appraisal. Choosing the best path for broadening one's knowledge.

#### TEACHING METHODOLOGY

---

(Section not available)

#### LEARNING OBJECTIVES OF THE SUBJECT

---

(Section not available)



## STUDY LOAD

Type	Hours	Percentage
Self study	112,5	60.00
Hours large group	45,0	24.00
Hours small group	30,0	16.00

Total learning time: 187.5 h

## CONTENTS

### COST OF ALGORITHMS

**Description:**

Asymptotic Notation. Analysis of the cost of recursive and iterative algorithms. Recurrences.

### ALGORITHMIC TECHNIQUES

**Description:**

Brute force. Divide-and-conquer. Greedy algorithms. Dynamic programming.

### USE OF BASIC DATA STRUCTURES

**Description:**

Stacks and queues. Priority queues. Sets and maps.

### IMPLEMENTATION OF BASIC DATA STRUCTURES

**Description:**

Heaps. Hash tables. Balanced search trees. MF-sets.

### ALGORITHMS ON GRAPHS

**Description:**

Representation. Breadth-first search and depth-first search, connectivity. Optimal paths. Minimum spanning trees.

## GRADING SYSTEM

---

The final subject mark will be worked out as  $2T/5 + 2L/5 + P/5$ , where T is the theory mark, L is the laboratory mark and P is the mark of the projects. The three marks are obtained independently.

To calculate the theory mark, two conventional exams on paper will be done, a midterm and a final exam, assessing the student knowledge on the subject as well as his problem solving skills. Be PT and FT their respective marks. Then,  $T = \text{Maximum}(PT/3 + 2FT/3, FT)$ .

For the laboratory mark, the students will be asked to do two exams on the computer, in which they will have to program the solution to some diverse algorithmical problems. It will be especially taken into account that the proposed program is correct, efficient, clear and that it uses the proper algorithmic schemes and data structures. Be PL the midterm laboratory exam mark and FL the final laboratory exam mark. Then,  $L = \text{Maximum}(PL/3 + 2FL/3, FL)$ .

Additionally, there will be a projects mark, worked out from the average of the marks of the projects handed over during the term.

An extra exam will take place on July for students that failed during the regular semester.

## BIBLIOGRAPHY

---

### Basic:

- Cormen, T.H. [et al.]. Introduction to algorithms [on line]. 3rd ed. Cambridge: MIT Press, 2009 [Consultation: 21/06/2023]. Available on: <https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?pq-origsite=primo&docID=3339142>. ISBN 9780262033848.
- Dasgupta S.; Papadimitriou C.; Vazirani U. Algorithms. Boston: Mc Graw Hill Higher Education, 2006. ISBN 9780073523408.
- Josuttis, Nicolai M. The C++ standard library : a tutorial and reference [on line]. 2a ed. Upper Saddle River, NJ: Addison-Wesley, 2012 [Consultation: 14/06/2024]. Available on: <https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?pq-origsite=primo&docID=7115780>. ISBN 9780321623218.
- Sedgewick, R. Algorithms in C++. Parts 1-5: fundamentals, data structures, sorting, searching, and graph algorithms. 3rd ed. Reading, Massachusetts: Addison-Wesley, 2002. ISBN 9780201726848.
- Skiena, Steven S. The Algorithm design manual [on line]. 2nd ed. London: Springer, 2012 [Consultation: 21/06/2023]. Available on: <http://link.springer.com/book/10.1007/978-1-84800-070-4>. ISBN 9781848000698.

### Complementary:

- Knuth, Donald E. The Art of computer programming. Vol. 1. 3rd ed. Reading, Massachusetts: Addison-Wesley, 1998. ISBN 0201896834.
- Knuth, Donald E. The Art of computer programming. Vol. 3. 3rd ed. Reading (Mass.): Addison-Wesley, 1998. ISBN 0201038218.
- Meyers, Scott. Effective C++ : 55 specific ways to improve your programs and designs. 3rd ed. Upper Saddle River, NJ: Addison-Wesley, 2005. ISBN 0321334876.
- Meyers, Scott. More effective C++ : 35 new ways to improve your programs and designs. Reading, Massachusetts: Addison-Wesley, 1996. ISBN 020163371X.
- Skiena, S.S.; Revilla, M. Programming challenges : the programming contest training manual [on line]. New York: Springer, 2003 [Consultation: 21/06/2023]. Available on: <https://link.springer-com.recursos.biblioteca.upc.edu/book/10.1007/b97559>. ISBN 0387001638.
- Stroustrup, Bjarne. Programming : principles and practice using C++. Upper Saddle River, NJ: Addison-Wesley, 2009. ISBN 9780321543721.
- Weiss, Mark Allen. Data structures and algorithm analysis in C++. 4th ed. Upper Saddle River: Pearson Education, 2014. ISBN 9780273769385.