# Algorithms and data structures

**Academic Year: ( 2019 / 2020 )** **Review date: 29-04-2020**

**Department assigned to the subject: Computer Science and Engineering Department**

**Coordinating teacher: SEGURA BEDMAR, ISABEL**

**Type: Compulsory ECTS Credits : 6.0**

**Year : 1 Semester : 2**

## REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

- Programming
- Calculus

## OBJECTIVES

Competences
CECRI1 (1 ECTS) Evidence Assessment: PRACTICE LABORATORY, WORK TEAM (CASE)
CECRI6 (Algorithms) (1.5 ECTS) Evidence Assessment: TESTING, LABORATORY PRACTICE, GROUP AND INDIVIDUAL WORK
CECRI7 (TYPES AND DATA STRUCTURES) (2.5 ECTS) Evidence Assessment: TESTING, LABORATORY PRACTICE, GROUP AND INDIVIDUAL WORK
CGB4 (0.50 ECTS) Evidence Assessment: PRACTICE LABORATORY TEAM WORK
CGB% (0.50 ECTS) Evidence Assessment: LABORATORY PRACTICE, GROUP WORK

- Capacity to analyze and synthesize (PO e).
- Capacity to organize and plan the work (PO d).
- Resolution of problems (PO e).
- Working as a team (PO d).
- Capacity to put in practice theory knowledge (PO e).

2. Specific Competences.

a. Cognitive (to know).

- General knowledge about algorithms (PO a).
- Understanding of basic data structures (PO k).
- Familiarity with advanced data structures (PO k).

b. Procedural/instrumental (to be able to do).

- To be able to design and analyze the algorithms complexity (PO a).
- To be able to understand and use different data structures (PO k).
- To be able to implement program solutions to specific problems using these tools (PO e).

c. Attitude (Being)

- Ability to solve problems through algorithms (PO e).
- Ability to clarify, simplify and efficiency of solving problems (PO e and k).
- Ability to question and conclude various solutions to any problem (PO e and k).

The learning outcomes are:

Individual work:
¿ Resolution by students of problems that must prove they have the ability to combine theory and practice. (PO a, e, k)

Group work:
Case Study on design and implementation of data structures. (PO a, d, e, k)

## DESCRIPTION OF CONTENTS: PROGRAMME

1. Introduction to Abstract Data Type

2. Linear Abstract Data Types
   b. Stacks
   c. Queues.
   d. Lists: singly and doubly linked lists

3. Algorithms I: Complexity
   a. Analysis of Algorithms
   b. Types of complexity
   c. Function Time.
   d. Notation Big-O.
   e. Worst and best cases.

4. Algorithms II: recursion.

5. Hierarchic Abstract Data types: Trees
   a. General Trees
   b. Binary Trees
   c. Tree Trasverse: preorder, inorder, postorder
   d. Search Binary Trees.
   e. Balanced BST.

6. Graphs
   a. Definition Graph ADT. Applications
   b. Implementation based on adjacency matrix.
   c. Implementation based on adjacency list.
   d. Graph trasversal: Depth-first search and breadth-first search.
   e. Dijkstra's algorithm

7. Divide and Conquer

## LEARNING ACTIVITIES AND METHODOLOGY

1. Theory Lectures with the objective of acquire the cognitive specific competences (PO a and k).
2. Academic activities guided by the teacher:
2.1. With the teacher: to solve exercises devoted to analyze, design and implement cases with different level of complexity in collaboration with students (PO a and k). Some of the exercises will be carried out in computer laboratories (PO k).
2.2. Student work: Homework, individually or cooperatively, with exercises, implementation cases and basic readings from bibliography proposed by the teacher (PO k and e).
Moreover, these activities can be performed as:
a. Individual work consisting on developing solutions to the problems and exercises posed by the teacher.
b. Working cooperatively developing solutions to the problems proposed by the teacher (PO d).
3. Mid-term partial exam and final exam (PO a, e, k).
4. There will be a group tutorship for each small group to solve the queries and doubts of students.

## ASSESSMENT SYSTEM

In addition to serve as formative activity, the exercises and examinations serve to be used as evaluation measure. During the course, two worksheets will be published at "Aula Global" site. Students should also try to solve a practical case study using the concepts learned during the course.
The evaluation includes the assessment of the guided academic activities and practical work according to the following weighting:
* Mid- term partial exam : 20 % (Po a, e, k)
* Practical case study: 20% (Po a, e, d, k).
* Weekly problems: During the first five weeks of the semester, a problem will be published on a weekly basis, which the student must try to solve individually and deliver their solution through AulaGlobal within 7 calendar days of its publication. If the student delivers and correctly solves at 4 of them, the grade may be increased by 5%. (Po a, e, k)
*Final exam: 60 % (Po a, e, k). This exam is mandatory for all students. Students must earn a grade of at least 4 (4/10) in order to pass the subject.

The final grade is obtained by adding the note of the partial tests (according to their weight). To pass the subject it is necessary to obtain a final grade equal to or greater than 5.

If a student decides not to follow the continuous assessment, he / she will be entitled to take a final

exam (same date and place as the ordinary exam). The grade obtained in this exam is equivalent to 60% of the final grade.

In the extraordinary call, the final exam will be 100% of the grade. The continuous evaluation may be applied if it is more beneficial for the student (partial tests and 60% of the final exam grade).

| | |
|---|---|
| **% end-of-term-examination:** | 60 |
| **% of continuous assessment (assigments, laboratory, practicals…):** | 40 |

## BASIC BIBLIOGRAPHY


- Karumanchi, N Data Structure and Algorithmic Thinking with Python: Data Structure and Algorithmic Puzzles. , CareerMonk Publications, 2015
- Michael T. Goodrich and Roberto Tamassia Data Structures and Algorithms in Python, , John Wiley & Sons, 2013


## ADDITIONAL BIBLIOGRAPHY


- Isabel Segura Bedmar, Harith AlJumaily, Julian Moreno Schneider, Juan Perea & Nathan D. Ryan Algorithms and Data Structures, OCW-UC3M: http://ocw.uc3m.es/ingenieria-informatica/algorithms-and-data-structures, 2011


## BASIC ELECTRONIC RESOURCES

- Isabel Segura Bedmar, Harith AlJumaily, Julian Moreno Schneider, Juan Perea & Nathan D. Ryan . ALGORITHMS AND DATA STRUCTURES: <a href="http://ocw.uc3m.es/ingenieria-informatica/algorithms-and-data-structures" target="_blank">http://ocw.uc3m.es/ingenieria-informatica/algorithms-and-data-structures</a>