



17827 - ANALYSIS OF ALGORITHMS

Information of the subject

Code - Course title: 17827 - ANALYSIS OF ALGORITHMS

Degree: 473 - Graduado/a en Ingeniería Informática
474 - Graduado/a en Ingeniería Informática y Matemáticas
722 - Graduado/a en Ingeniería Informática
734 - Graduado/a en Ingeniería Informática y Matemáticas (2019)

Faculty: 350 - Escuela Politécnica Superior

Academic year: 2020/21

1. Course details

1.1. Content area

Programación y estructuras de datos

1.2. Course nature

Compulsory

1.3. Course level

Grado (EQF/MECU 6)

1.4. Year of study

2

1.5. Semester

First semester

1.6. ECTS Credit allotment

6.0

1.7. Language of instruction

Español

1.8. Prerequisites

None.

1.11. Subject coordinator/s

Secure Verification Code:		Date:	14/01/2021	
Signed by:	This teaching guide is not SVC signed because is not the final version			
URL Verification:		Page:	1/3	

1.12. Competences and learning outcomes

1.12.1. Competences

CC3 Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.

1.13. Course contents

1. Analysis of the efficiency of algorithms

1. Types of analysis. Efficiency metrics.
2. Mathematical tools: o , O , etc. notations.
3. Sums and integrals growth.
4. Worst, best and average cases

2. Basic sorting algorithms

1. Selection and Bubble algorithms.
2. Insertion algorithm: pseudocode, efficiency.
3. Local algorithms: definition, lower bounds for the worst and average cases.

3. Advanced sorting algorithms

1. Divide and conquer methods.
2. Recurrent inequalities. Growth estimation.
3. Mergesort Algorithm: pseudocode, worst case.
4. Quicksort Algorithm: pseudocode, worst case, average case.
5. Heapsort Algorithm: pseudocode, worst case.

4. Decision trees for sorting algorithms

1. Decision trees: concept and construction.
2. Lower bounds for key comparison sorting algorithms.

5. Basic searching algorithms

1. Linear and binary search.
2. Dictionary ADT. Binary trees as DS for dictionaries.
3. Worst and average cases for binary tree search.
4. AVL trees: construction and maximum depth.

6. Hash tables

1. Design of hash functions and collision resolution.
2. Hash tables with chaining.
3. Hash tables with open addressing: linear, quadratic and random probing.

1.14. Course bibliography

There is no a single manual that meets in full the course contents. Anyway, the following references are a good complement to follow the course.

Basic references:

- [Weiss, Data structures and algorithm analysis in C, Benjamin Cummings.](#)

Supplementary references:

- [Cormen, Leiserson, Rivest. Introduction to algorithms. The MIT Press--Mc Graw Hill.](#)
- [Baase, Computer algorithms, Addison-Wesley.](#)

Secure Verification Code:		Date:	14/01/2021	
Signed by:	This teaching guide is not SVC signed because is not the final version			
URL Verification:		Page:	2/3	

Programming references:

- [Kernighan, Ritchie. The C programming language. Prentice hall.](#)

2. Teaching-and-learning methodologies and student workload

2.1. Contact hours

	#horas
Contact hours (minimum 33%)	
Independent study time	

2.2. List of training activities

Activity	# hours
Lectures	
Seminars	
Practical sessions	
Clinical sessions	
Computer lab	
Laboratory	
Work placement	
Supervised study	
Tutorials	
Assessment activities	
Other	

3. Evaluation procedures and weight of components in the final grade

3.1.1. List of evaluation activities

Evaluatory activity	%
Final exam	
Continuous assessment	

3.2.1. List of evaluation activities

Evaluatory activity	%
Final exam	
Continuous assessment	

Secure Verification Code:		Date:	14/01/2021	
Signed by:	This teaching guide is not SVC signed because is not the final version			
URL Verification:		Page:	3/3	