

## GUÍA DOCENTE ABREVIADA DE LA ASIGNATURA

### G652 - Algorítmica y Complejidad

Doble Grado en Física y Matemáticas  
Grado en Ingeniería Informática  
Grado en Matemáticas

Curso Académico 2023-2024

1. DATOS IDENTIFICATIVOS					
Título/s	Doble Grado en Física y Matemáticas Grado en Ingeniería Informática Grado en Matemáticas			Tipología y Curso	Optativa. Curso 5 Obligatoria. Curso 2 Optativa. Curso 4
Centro	Facultad de Ciencias				
Módulo / materia	MATERIA PROGRAMACIÓN DE COMPUTADORES MENCION EN INFORMÁTICA MÓDULO OBLIGATORIO				
Código y denominación	G652 - Algorítmica y Complejidad				
Créditos ECTS	6	Cuatrimestre	Cuatrimestral (2)		
Web					
Idioma de impartición	Español	English friendly	No	Forma de impartición	Presencial

Departamento	DPTO. MATEMATICAS, ESTADISTICA Y COMPUTACION				
Profesor responsable	CAMILO PALAZUELOS CALDERON				
E-mail	camilo.palazuelos@unican.es				
Número despacho	Facultad de Ciencias. Planta: + 1. DESPACHO (1053)				
Otros profesores	ALBERTO MARTINEZ FERNANDEZ				

### 3.1 RESULTADOS DE APRENDIZAJE

- Ser capaz de elegir y usar eficazmente los esquemas algorítmicos más habituales, incluidos los métodos de búsqueda heurística y exploración exhaustiva.
- Conocer con familiaridad las consecuencias de los crecimientos logarítmico, lineal, cuadrático, polinómico y exponencial, y ser capaz de valorar la complejidad computacional de un problema dado, explicando así el comportamiento en términos de tiempo de ejecución de un programa que lo resuelva.

**4. OBJETIVOS**

Presentar la notación de Landau para expresar los costes temporal y espacial de los algoritmos, así como las reglas para el cálculo de la complejidad computacional de los algoritmos iterativos.

Caracterizar las clases de complejidad P, NP y NP-completo, poniendo de manifiesto las implicaciones de las relaciones entre ellas, e introducir el concepto de reducción de un problema computacional a otro.

Enunciar las distintas versiones del teorema maestro para el cálculo de la complejidad computacional de los algoritmos «divide y vencerás» y reconocer las diferencias entre sus condiciones de aplicación.

Relacionar problemas clásicos de la teoría de grafos y, en función de sus características, proponer soluciones con programación dinámica, algoritmos voraces o búsquedas exhaustivas.

Proporcionar estrategias de diseño de algoritmos de búsqueda exhaustiva usando técnicas como «vuelta atrás» o «ramificación y poda».

Diseñar algoritmos iterativos a partir de algoritmos recursivos lineales y múltiples utilizando esquemas de transformación simples.

**6. ORGANIZACIÓN DOCENTE**

**CONTENIDOS**

1	<p><b>ANÁLISIS DE ALGORITMOS</b></p> <ul style="list-style-type: none"> <li>· Comparación asintótica de funciones</li> <li>· Complejidad en los casos peor y promedio</li> <li>· Clases de complejidad P, NP y NP-completo</li> </ul>
2	<p><b>DIVIDE Y VENCERÁS</b></p> <ul style="list-style-type: none"> <li>· Algoritmo general y versiones del teorema maestro</li> <li>· Problemas numéricos y de búsqueda en arrays</li> <li>· Ordenamiento de arrays (mergesort y quicksort)</li> </ul>
3	<p><b>PROGRAMACIÓN DINÁMICA</b></p> <ul style="list-style-type: none"> <li>· Principio de optimalidad y memoización</li> <li>· Enfoques de arriba abajo y de abajo arriba</li> <li>· Problemas numéricos, de arrays y de grafos</li> </ul>
4	<p><b>ALGORITMOS VORACES</b></p> <ul style="list-style-type: none"> <li>· Algoritmo general y funciones de selección</li> <li>· Problemas de arrays y de búsqueda en grafos</li> <li>· Introducción a los algoritmos de aproximación</li> </ul>
5	<p><b>VUELTA ATRÁS</b></p> <ul style="list-style-type: none"> <li>· Algoritmo general y representación de la solución</li> <li>· Problemas de arrays y de búsqueda en grafos</li> <li>· Diseño iterativo de algoritmos de vuelta atrás</li> </ul>
6	<p><b>RAMIFICACIÓN Y PODA</b></p> <ul style="list-style-type: none"> <li>· Algoritmo general y representación de la solución</li> <li>· Coste de la solución y cotas inferior y superior</li> <li>· Algoritmos de aproximación como función de cota</li> </ul>
7	<p><b>TRANSFORMACIÓN DE DISEÑOS</b></p> <ul style="list-style-type: none"> <li>· De recursivo lineal final a iterativo</li> <li>· De recursivo lineal no final a iterativo</li> <li>· De recursivo múltiple a iterativo</li> </ul>

**7. MÉTODOS DE LA EVALUACIÓN**

Descripción	Tipología	Eval. Final	Recuper.	%
Examen de teoría y problemas	Examen escrito	Sí	Sí	60,00
Prácticas de laboratorio	Otros	No	Sí	20,00
Problemas y ejercicios	Otros	No	Sí	20,00
<b>TOTAL</b>				<b>100,00</b>
<b>Observaciones</b>				
Los alumnos podrán recuperar las prácticas de laboratorio y los problemas y ejercicios entregando, antes de la realización del examen extraordinario de teoría y problemas, el material corregido según la propuesta de mejora hecha por el profesor.				
<b>Criterios de evaluación para estudiantes a tiempo parcial</b>				
Los alumnos a tiempo parcial podrán superar la asignatura concurriendo a dos exámenes: · de teoría y problemas (80 % de la calificación global), con una calificación mínima de 5,0; · de prácticas de laboratorio (20 % de la calificación global).				

**8. BIBLIOGRAFÍA Y MATERIALES DIDÁCTICOS**

<b>BÁSICA</b>
<p><b>INTRODUCTION TO ALGORITHMS</b> T. H. Cormen, C. E. Leiserson, R. L. Rivest y C. Stein The MIT Press, 4.ª edición, 2022</p>
<p><b>ALGORITHM DESIGN</b> J. Kleinberg y É. Tardos Addison-Wesley, 2005</p>

Esta es la Guía Docente abreviada de la asignatura. Tienes también publicada en la Web la información más detallada de la asignatura en la Guía Docente Completa.