



## 1. Identificación

### 1.1. De la Asignatura

Curso Académico	2022/2023
Titulación	GRADO EN INGENIERÍA INFORMÁTICA y PROGRAMA ACADÉMICO DE SIMULTANEIDAD DE DOBLE TITULACIÓN CON ITINERARIO ESPECÍFICO DE GRADO EN MATEMÁTICAS Y GRADO EN INGENIERÍA INFORMÁTICA
Nombre de la Asignatura	ALGORITMOS Y ESTRUCTURAS DE DATOS II
Código	1900
Curso	SEGUNDO,SEGUNDO(IC) (SIN DOCENCIA) y SEGUNDO(IC)
Carácter	OBLIGATORIA
N.º Grupos	5
Créditos ECTS	6
Estimación del volumen de trabajo del alumno	150
Organización Temporal/Temporalidad	2 Cuatrimestre y 2 Cuatrimestre(IC)
Idiomas en que se imparte	ESPAÑOL
Tipo de Enseñanza	Presencial

### 1.2. Del profesorado: Equipo Docente

Coordinación de la asignatura  NORBERTO MARIN PEREZ	Área/Departamento	LENGUAJES Y SISTEMAS INFORMÁTICOS/INFORMÁTICA Y SISTEMAS
	Categoría	PROFESORES TITULARES DE ESCUELAS UNIVERSITARIAS
	Correo Electrónico /	nmarin@um.es
	Página web / Tutoría electrónica	http://webs.um.es/nmarin/ Tutoría Electrónica: SÍ



Grupo de Docencia:	Teléfono, Horario y Lugar de atención al alumnado	Duración	Día	Horario	Lugar
1, 2, 3, 9 y B Coordinación de los grupos:1,3,9(IC) y B(IC)		Anual	Martes	16:45- 17:30	868884637, Facultad de Informática B1.2.035
		Anual	Martes	10:15- 11:00	868884637, Facultad de Informática B1.2.035
		Anual	Miércoles	16:45- 17:30	868884637, Facultad de Informática B1.2.035
		Anual	Miércoles	10:15- 11:00	868884637, Facultad de Informática B1.2.035
JOAQUIN CERVERA LOPEZ	Área/Departamento	LENGUAJES Y SISTEMAS INFORMÁTICOS/INFORMÁTICA Y SISTEMAS			
	Categoría	PROFESORES TITULARES DE UNIVERSIDAD			
Grupo de Docencia: 1, 2, 3, 9 y B	Correo Electrónico / Página web / Tutoría electrónica	jcervera@um.es <a href="http://webs.um.es/jcervera">http://webs.um.es/jcervera</a> Tutoría Electrónica: Sí			



Coordinación de los grupos:2	Teléfono, Horario y Lugar de atención al alumnado	Duración	Día	Horario	Lugar	Observaciones
		Primer Cuatrimestre	Jueves	15:00- 16:30	868888334, Facultad de Informática B1.2.048	Concertar cita.
		Primer Cuatrimestre	Jueves	08:30- 10:00	868888334, Facultad de Informática B1.2.048	Concertar cita.
		Segundo Cuatrimestre	Martes	09:30- 12:30	868888334, Facultad de Informática B1.2.048	Periodos exámenes, concertar cita.
FRANCISCO JOSE MONTOYA DATO	Área/Departamento	LENGUAJES Y SISTEMAS INFORMÁTICOS/INFORMÁTICA Y SISTEMAS				
Grupo de Docencia: 1, 2, 3, 9 y B	Categoría	PROFESORES TITULARES DE UNIVERSIDAD				
	Correo Electrónico / Página web / Tutoría electrónica	fmontoya@um.es <a href="https://webs.um.es/fmontoya">https://webs.um.es/fmontoya</a> Tutoría Electrónica: NO				



	Teléfono, Horario y Lugar de atención al alumnado	Duración	Día	Horario	Lugar	Observaciones
		Primer Cuatrimestre	Jueves	10:40- 12:10	868884620, Facultad de Informática B1.2.008	Solicitar cita previa min 24h lectivas antes por mensaje privado AV
		Primer Cuatrimestre	Jueves	15:20- 18:50	868884620, Facultad de Informática B1.2.008	Solicitar cita previa min 24h lectivas antes por mensaje privado AV
		Segundo Cuatrimestre	Martes	09:20- 12:20		Solicitar cita previa min 24h lectivas antes por mensaje privado AV
		Segundo Cuatrimestre	Miércoles	09:20- 12:20	868884620, Facultad de Informática B1.2.008	Solicitar cita previa min 24h lectivas antes por mensaje privado AV

## 2. Presentación

Los algoritmos y las estructuras de datos constituyen los dos pilares básicos de la programación. Las asignaturas AED I y II se pueden ver como una continuación natural de las asignaturas de programación de primer curso. En particular, en AED II, partiendo de una base previa en los principios de la programación, se



intenta desarrollar en los alumnos las habilidades de resolución de problemas, con herramientas como los esquemas algorítmicos, el razonamiento inductivo, la descomposición modular, la aplicación de procesos metódicos, y en general el pensamiento algorítmico. Los conocimientos adquiridos en la asignatura resultan fundamentales en todas las asignaturas posteriores donde se requieran conocimientos de programación.

### 3. Condiciones de acceso a la asignatura

#### 3.1 Incompatibilidades

No consta

#### 3.2 Recomendaciones

Para cursar esta asignatura es altamente recomendable haber cursado y aprobado las asignaturas de programación de primer curso. En concreto, los alumnos deben saber programar con soltura en el lenguaje C ya que, en caso contrario, encontrarán muchas dificultades para superar la asignatura.

Es aconsejable, a su vez, haber cursado y aprobado la asignatura de Algoritmos y Estructuras de Datos I del primer cuatrimestre.

### 4. Competencias

#### 4.1 Competencias Básicas

No disponible

#### 4.2 Competencias de la titulación

- CGII1. Capacidad de análisis y síntesis.
- CGII7. Resolución de problemas.
- CEII2. Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática de acuerdo con los conocimientos adquiridos.
- CR1. Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
- CR6. Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CR7. Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.



#### 4.3 Competencias transversales y de materia

### 5. Contenidos

#### Bloque 1: Algorítmica

##### TEMA 1. Análisis de algoritmos

1.1. Introducción. 1.2. Notaciones asintóticas. 1.3. Ecuaciones de recurrencia. 1.4. Ejemplos.

##### TEMA 2. Divide y vencerás

2.1. Método general. 2.2. Análisis de tiempos de ejecución. 2.3. Ejemplos de aplicación.

##### TEMA 3. Algoritmos voraces

3.1. Método general. 3.2. Análisis de tiempos de ejecución. 3.3. Ejemplos de aplicación.

##### TEMA 4. Programación dinámica

4.1. Método general. 4.2. Análisis de tiempos de ejecución. 4.3. Ejemplos de aplicación.

##### TEMA 5. Backtracking

5.1. Método general. 5.2. Análisis de tiempos de ejecución. 5.3. Ejemplos de aplicación.

### PRÁCTICAS

Práctica 1. Eficiencia y Divide y Vencerás.: Relacionada con los contenidos Tema 1 (Bloque 1) y Tema 2 (Bloque 1)

Resolución de ejercicios programados y estudio de complejidad.

Práctica 2. Algoritmos voraces y Backtracking: Relacionada con los contenidos Tema 3 (Bloque 1) y Tema 5 (Bloque 1)

Resolución de ejercicios programados y estudio de complejidad.

### 6. Metodología Docente

Actividad Formativa	Metodología	Horas Presenciales	Trabajo Autónomo	Volumen de trabajo
Lección magistral		24	36	60
Seminarios		6	9	15
Prácticas		24	36	60



Actividad Formativa	Metodología	Horas Presenciales	Trabajo Autónomo	Volumen de trabajo
Exámenes		6	9	15
	Total	60	90	150

## 7. Horario de la asignatura

<https://www.um.es/web/estudios/grados/informatica/horarios-examenes>

## 8. Sistema de Evaluación

Métodos / Instrumentos	Examen teórico-práctico. En este instrumento incluimos desde el tradicional examen escrito o tipo test hasta los exámenes basados en resolución de problemas, pasando por los de tipo mixto que incluyen cuestiones cortas o de desarrollo teórico junto con pequeños problemas. También se incluye aquí la consideración de la participación activa del alumno en clase, la entrega de ejercicios o realización de pequeños trabajos escritos y presentaciones.
Criterios de Valoración	Los alumnos realizarán exámenes de ejercicios del tema 1, Análisis de Algoritmos, y del tema 4, Programación Dinámica. Superar cualquiera de esos temas en dichos exámenes los eliminaría del examen final.
Ponderación	50
Métodos / Instrumentos	Informe técnico. En este instrumento incluimos los resultados de actividades prácticas, o de laboratorio, junto con sus memorias descriptivas. Los resúmenes del estado del arte o memorias de investigación sobre temas concretos. Y la posibilidad de realizar entrevistas personales o presentaciones de los trabajos realizados también entran en esta categoría.
Criterios de Valoración	Se realizarán dos prácticas en grupos de 2 alumnos, en C++, la primera sobre contenidos de los temas 1 y 2, Análisis de Algoritmos y Divide y Vencerás, y la segunda sobre contenidos de los temas 3 y 5, Algoritmos Voraces (Avance rápido) y Backtracking. Para esta última práctica se usará el juez on-line Mooshak.  Cada una de las prácticas se evalúa sobre 10, y se requiere un 5 en cada una de ellas para aprobarlas; cada práctica tiene un peso del 50% de la nota de prácticas. Puede requerirse entrevista con los alumnos.
Ponderación	50



## Fechas de exámenes

<https://www.um.es/web/estudios/grados/informatica/horarios-examenes>

## 9. Resultados del Aprendizaje

- Tomar conciencia de la importancia de realizar siempre un análisis y diseño previos del problema, como pasos anteriores a la implementación en un lenguaje de programación.
- Distinguir los distintos tiempos de ejecución que se utilizan en el estudio de algoritmos: caso más favorable, más desfavorable y promedio.
- Aprender las técnicas básicas del estudio teórico de algoritmos: análisis del tiempo por conteo de instrucciones y estudio de la ocupación de memoria. Comprender la importancia de este estudio.
- Aprender a realizar estudios experimentales, comprender su importancia y la necesidad de contrastar los resultados experimentales con los teóricos.
- Conocer las notaciones de complejidad  $O$  (o-grande, orden), Omega, Theta y  $o$  (o-pequeña), diferenciando claramente el significado y utilidad de cada una.
- Saber comparar las complejidades de distintas funciones.
- Conocer la complejidad de las funciones que aparecen más frecuentemente en el estudio de algoritmos, así como la relación entre las complejidades de estas funciones.
- Conocer las técnicas básicas de resolución de ecuaciones de recurrencia: expansión de la recurrencia, método de la ecuación característica y utilización de fórmulas maestras.
- Comprender el papel de las condiciones iniciales (casos base) en la resolución de ecuaciones recurrentes, y saber determinar cuáles se deben aplicar para cada ecuación dada.
- Comprender la técnica de resolución por división en problemas más pequeños.
- Conocer y saber aplicar los esquemas básicos de los algoritmos divide y vencerás.
- Comprender la técnica de resolución de problemas por avance rápido y los distintos casos que se pueden presentar en la resolución de problemas por esta técnica: obtención de la solución óptima, de una solución no óptima, o no obtención de la solución.
- Entender el método de avance rápido como técnica para obtener una solución inicial a partir de la cual se puede realizar una búsqueda local, y conocer algunos ejemplos que se pueden resolver de ese modo.



- Comprender la técnica de resolución de problemas por programación dinámica, e identificar las diferencias con divide y vencerás y con avance rápido.
- Entender la ventaja de la programación dinámica con respecto a otras técnicas en cuanto a tiempo de ejecución, al evitar la repetición de cálculos que aparece en métodos recursivos.
- Saber identificar problemas que cumplen el principio de optimalidad, que es necesario para poder aplicar esta técnica.
- Entender el concepto de árbol de solución, y su utilización en la resolución de problemas. Comprender que en muchos casos el árbol es sólo una representación lógica del conjunto de todas las posibles soluciones y subsoluciones.
- Comprender y saber aplicar la técnica de resolución de problemas por backtracking, entendiendo sus características principales y el concepto de búsqueda en profundidad.
- Comprender y saber aplicar la técnica de resolución de problemas por ramificación y poda, entendiendo la importancia de la estimación del beneficio para guiar la búsqueda, y del cálculo de las cotas para podar el árbol de búsqueda.
- Comprender la importancia de la heurística en la resolución eficiente de problemas de alto coste computacional.

## 10. Bibliografía

### Bibliografía Básica



Heileman, Gregory L. -- Estructuras de datos, algoritmos, programación orientada a o (1998)



Joyanes Aguilar, Luis -- Estructura de datos : algoritmos, abstracción y objetos (1998)



Kernighan, Brian W. -- El language de programación C (1991)



Martí Oliet, Narciso -- Estructuras de datos y métodos algorítmicos : ejercicios resueltos (2004)



Peña Mar, Ricardo -- Diseño de programas : formalismo y abstracción (1998)



Stroustrup, Bjarne -- El lenguaje de programación C (1998)



-  Aho, Alfred V. -- Estructuras de datos y algoritmos (1998)
-  Algoritmos y estructuras de datos (2003)
-  Baase, Sara -- Computer algorithms : introduction to design and analysis (2000)
-  Brassard, Gilles (1955-) -- Fundamentos de algoritmia (1997)
-  Introduction to algorithms (2001)
-  Mehlhorn, K, Data Structures and algorithms, (1984)
-  Weiss, Mark Allen -- Estructuras de datos y algoritmos (1995)
-  Horebeek, Ivo Van -- Algebraic specifications in software engineering : an introduction (1989)

### Bibliografía Complementaria

-  Wirth, Niklaus -- Algoritmos estructuras de datos = programas (1994)

## 11. Observaciones y recomendaciones

Los criterios para establecer la nota que aparecerá en el acta son los siguientes:

- Si el alumno no se presenta al examen de teoría ni realiza ninguna entrega de prácticas, su calificación será "No Presentado".
- Si el alumno supera una parte y no se presenta a la otra, su calificación será "No Presentado".
- Si el alumno suspende una parte y no se presenta a la otra, su calificación será "Suspenso" con la mitad de la nota de la parte suspensa. En este caso, la calificación numérica reflejada en el acta no sería superior a 4,5.
- Si el alumno suspende una parte y aprueba la otra, su calificación será "Suspenso" con la media ponderada. Si esa media fuera igual o superior a 4,5 la calificación numérica reflejada en el acta sería 4,5.
- Cuando el alumno apruebe o suspenda las dos partes, su calificación será el resultado de aplicar la media ponderada. Esto será así en cualquier convocatoria.

FECHAS APROXIMADAS DE LAS ACTIVIDADES EVALUATORIAS:

Examen tema 1 en laboratorio: Análisis de Algoritmos: 6ª semana



Examen parcial tema 1: Análisis de Algoritmos: 8ª semana

Examen tema 4 en laboratorio: Programación Dinámica: 10ª semana

Práctica temas 1-2: Análisis y Divide y Vencerás. Entrega: 9ª semana

Práctica temas 3 y 5: Algoritmos Voraces (Avance rápido) y Backtracking. Entrega: 14ª semana

“NECESIDADES EDUCATIVAS ESPECIALES. Aquellos estudiantes con discapacidad o necesidades educativas especiales podrán dirigirse al Servicio de Atención a la Diversidad y Voluntariado (ADYV; <http://www.um.es/adyv/>) para recibir orientación sobre un mejor aprovechamiento de su proceso formativo y, en su caso, la adopción de medidas de equiparación y de mejora para la inclusión, en virtud de la Resolución Rectoral R-358/2016. El tratamiento de la información sobre este alumnado, en cumplimiento con la normativa vigente, es de estricta confidencialidad.”