



UNIVERSIDAD
DE MURCIA

1. Identificación

1.1. De la asignatura

Curso Académico	2024/2025
Titulación	GRADO EN INGENIERÍA INFORMÁTICA, 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
Carácter	OBLIGATORIA
Número de grupos	4
Créditos ECTS	6.0
Estimación del volumen de trabajo	150.0 150.0
Organización temporal	2º Cuatrimestre 2º Cuatrimestre
Idiomas en que se imparte	Español

1.2. Del profesorado: Equipo docente

CERVERA LOPEZ, JOAQUIN

Docente: **GRUPO 1**

Coordinación de los grupos: **GRUPO 1, GRUPO 2, GRUPO 3, GRUPO PCEO MATE+INFORM**

Coordinador de la asignatura

Categoría

PROFESORES TITULARES DE UNIVERSIDAD

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

jcervera@um.es <http://webs.um.es/jcervera> Tutoría electrónica: **Sí**

Teléfono, horario y lugar de atención al alumnado

Duración:	Día:	Horario:	Lugar:
C1	Viernes	09:30-12:30	868888334, Facultad de Informática B1.2.048 (DESPACHO CERVERA LOPEZ, JOAQUÍN 2.38)

Observaciones:

despacho 2.38, facultad de informática, por favor, pedir cita previa por mensaje privado de aula virtual

HERNANDEZ LOPEZ, JOSE ANTONIO

Docente: **GRUPO 1, GRUPO 2, GRUPO 3, GRUPO PCEO MATE+INFORM**

Coordinación de los grupos:

Categoría

PROFESOR AYUDANTE DOCTOR

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

joseantonio.hernandez6@um.es Tutoría electrónica: **No**

Teléfono, horario y lugar de atención al alumnado

HOYOS BARCELO, JOSE RAMON

Docente: **GRUPO 1, GRUPO 2, GRUPO 3, GRUPO PCEO MATE+INFORM**

Coordinación de los grupos:

Categoría

PROFESOR COLABORADOR (LICENCIADO)

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

jose.hoyos@um.es Tutoría electrónica: **Sí**

Teléfono, horario y lugar de atención al alumnado

Duración:	Día:	Horario:	Lugar:
C2	Martes	11:30-13:00	868884398, Facultad de Informática B1.2.024

Observaciones:

despacho 2.18

Duración:	Día:	Horario:	Lugar:
------------------	-------------	-----------------	---------------

C2 Jueves 11:30-13:00 868884398, Facultad de Informática B1.2.024

Observaciones:
despacho 2.18

Duración: C1 **Día:** Martes **Horario:** 12:00-13:30 **Lugar:** 868884398, Facultad de Informática B1.2.024

Observaciones:
despacho 2.18

Duración: C1 **Día:** Miércoles **Horario:** 12:00-13:30 **Lugar:** 868884398, Facultad de Informática B1.2.024

Observaciones:
despacho 2.18

HOYOS BARCELO, CARLOS

Docente: GRUPO 1, GRUPO 2, GRUPO 3, GRUPO PCEO MATE+INFORM

Coordinación de los grupos:

Categoría

PROFESOR CONTRATADO PARA SUSTITUCIONES

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

chb4@um.es Tutoría electrónica: No

Teléfono, horario y lugar de atención al alumnado

MONTOYA DATO, FRANCISCO JOSE

Docente: GRUPO 1, GRUPO 2, GRUPO 3, GRUPO PCEO MATE+INFORM

Coordinación de los grupos:

Categoría

PROFESORES TITULARES DE UNIVERSIDAD

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

fmontoya@um.es <https://webs.um.es/fmontoya> Tutoría electrónica: No

Teléfono, horario y lugar de atención al alumnado

Duración: A **Día:** Lunes **Horario:** 05:00-05:10 **Lugar:** 868884620, Facultad de Informática B1.2.008

Observaciones:

Visítese esta dirección (cópíe la y péguela, no siga el enlace): <https://webs.um.es/fmontoya/miwiki/doku.php?id=tutorias>

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 constan

3.2. Requisitos

No constan

3.3. 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 o en 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

- CB1: Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio.
- CB2: Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.

- CB3: Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética
- CB4: Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado
- CB5: Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

4.2. Competencias de la titulación

- CGII1: Capacidad de análisis y síntesis.
- CGII6: Capacidad de gestión de la información.
- CGII7: Resolución de problemas.
- CGII10: Trabajo en un equipo de carácter interdisciplinar.
- CGII11: Trabajo en un contexto internacional.
- CGII12: Habilidades en las relaciones interpersonales.
- CGII13: Reconocimiento a la diversidad y la multiculturalidad.
- CGII15: Compromiso ético.
- CGII18: Creatividad
- CGII19: Liderazgo
- CGII20: Conocimiento de otras culturas y costumbres.
- CGII21: Iniciativa y espíritu emprendedor.
- CGII23: Sensibilidad hacia temas medioambientales.
- CGUM2: Comprender y expresarse en un idioma extranjero en su ámbito disciplinar, particularmente el inglés.
- CGUM4: Considerar la ética y la integridad intelectual como valores esenciales de la práctica profesional.
- CGUM5: Ser capaz de proyectar los conocimientos, habilidades y destrezas adquiridos para promover una sociedad basada en los valores de la libertad, la justicia, la igualdad y el pluralismo.
- CGUM7: Desarrollar habilidades de iniciación a la investigación.
- CEII2: Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática de acuerdo con los conocimientos adquiridos.
- CEII7: Capacidad para conocer, comprender y aplicar la legislación necesaria durante el desarrollo de la profesión de Ingeniero Técnico en Informática y manejar especificaciones, reglamentos y normas de obligado cumplimiento
- CEII10: Conocimientos para la realización de mediciones, cálculos, valoraciones, tasaciones, peritaciones, estudios, informes, planificación de tareas y otros trabajos análogos de informática.
- CEII11: Capacidad para analizar y valorar el impacto social y medioambiental de las soluciones técnicas, comprendiendo la responsabilidad ética y profesional de la actividad del Ingeniero Técnico en Informática.
- CEII12: Conocimiento y aplicación de elementos básicos de economía y de gestión de recursos humanos, organización y planificación de proyectos, así como la legislación, regulación y normalización en el ámbito de los proyectos informáticos.
- 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.

- CR3: Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
- 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

No constan

5. Contenidos

5.1. Teoría

Bloque 1: Algorítmica

Tema 1: Análisis de algoritmos

11 Introducción 12 Notaciones asintóticas 13 Ecuaciones de recurrencia 14 Ejemplos

Tema 2: Divide y vencerás

21 Método general 22 Análisis de tiempos de ejecución 23 Ejemplos de aplicación

Tema 3: Algoritmos voraces

31 Método general 32 Análisis de tiempos de ejecución 33 Ejemplos de aplicación

Tema 4: Programación dinámica

41 Método general 42 Análisis de tiempos de ejecución 43 Ejemplos de aplicación

Tema 5: Backtracking

51 Método general 52 Análisis de tiempos de ejecución 53 Ejemplos de aplicación

5.2. Prácticas

■ Práctica 1: Eficiencia y Divide y Vencerás.

Resolución de ejercicios programados y estudio de complejidad

Relacionado con:

- Tema 1: Análisis de algoritmos
- Tema 2: Divide y vencerás

■ Práctica 2: Algoritmos voraces y Backtracking

Resolución de ejercicios programados y estudio de complejidad

Relacionado con:

- Tema 3: Algoritmos voraces

6. Actividades Formativas

Actividad Formativa	Metodología	Horas	Presencialidad
A1: Actividades con grupo grande de alumnos entre las que se encuentran la presentación en el aula de los conceptos propios de la materia mediante metodología expositiva con lecciones magistrales participativas y medios audiovisuales. También se contemplan en este grupo las actividades de evaluación teórico prácticas.		24.0	40.0
A2: Actividades con grupo mediano en el aula de resolución de problemas, seminarios, charlas, ejercicios basados en el aprendizaje orientado a proyectos, estudios de casos, exposición y discusión de trabajos relativas al seguimiento individual y/o grupal de adquisición de las competencias.		6.0	12.5
A3: Actividades con grupo pequeño en el laboratorio relacionadas con la componente práctica de las asignaturas, desarrollo de trabajos con equipo técnico especializado, desarrollo de programas, etc.		24.0	37.5
A4: Tutorías individualizadas o en grupo muy pequeño orientadas a la dirección, supervisión y asesoría por parte del un profesor de la asignatura, del tutor en el caso de Trabajo Fin de Grado, supervisión del tutor de empresa en el caso de Prácticas de Empresa que de forma periódica constate y redirija el trabajo del alumno hacia la consecución de los objetivo marcados.		6.0	10.0
A5: Estudio y trabajo autónomo orientado a la asimilación de contenidos, realización de problemas, ejercicios o redacción de informes técnicos o memorias descriptivas, desarrollo de proyectos o prácticas individuales o en		90.0	0.0
	Totales	150,00	

7. Horario de la asignatura

<https://www.um.es/web/estudios/grados/informatica/2024-25#horarios>

8. Sistemas de Evaluación

Identificador	Denominación del instrumento de evaluación	Criterios de Valoración	Ponderación
IE1	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	Los alumnos realizarán exámenes de ejercicios del tema 1, Análisis de Algoritmos, y del tema 4,	50.0

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.

Programación Dinámica Superar cualquiera de esos temas en dichos exámenes los eliminaría del examen final

IE2	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.	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	50.0
		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	

9. Fechas de exámenes

<https://www.um.es/web/estudios/grados/informatica/2024-25#exámenes>

10. 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

11. Bibliografía

Bibliografía básica

- [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 \(1997\)](#)
- [Heileman, Gregory L. -- Estructuras de datos, algoritmos, programación orientada a o \(1998\)](#)
- [Horebeek, Ivo Van -- Algebraic specifications in software engineering : an introduction \(1989\)](#)
- [Introduction to algorithms \(2001\)](#)
- [Joyanes Aguilar, Luis -- Estructura de datos : algoritmos, abstracción y objetos \(1998\)](#)
- [Kernighan, Brian W. -- El lenguaje de programación C \(1991\)](#)
- [Martí Oliet, Narciso -- Estructuras de datos y métodos algorítmicos : ejercicios resueltos \(2004\)](#)
- [Mehlhorn, K, Data Structures and algorithms, \(1984\)](#)
- [Peña Mar, Ricardo -- Diseño de programas : formalismo y abstracción \(1998\)](#)

- [Stroustrup, Bjarne -- El lenguaje de programación C \(1998\)](#)
- [Weiss, Mark Allen -- Estructuras de datos y algoritmos \(1995\)](#)

Bibliografía complementaria

- [Wirth, Niklaus -- Algoritmos estructuras de datos = programas \(1994\)](#)

12. Observaciones

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.umes/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.

Esta asignatura no se encuentra vinculada de forma directa con los Objetivos de Desarrollo Sostenible

NECESIDADES EDUCATIVAS ESPECIALES

Aquellos estudiantes con discapacidad o necesidades educativas especiales podrán dirigirse al Servicio de Atención a la Diversidad y Voluntariado (ADYV - <https://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 LOPD, es de estricta confidencialidad.

REGLAMENTO DE EVALUACIÓN DE ESTUDIANTES

El artículo 8.6 del Reglamento de Evaluación de Estudiantes (REVA) prevé que "salvo en el caso de actividades definidas como obligatorias en la guía docente, si el o la estudiante no puede seguir el proceso de evaluación continua por circunstancias sobrevenidas debidamente justificadas, tendrá derecho a realizar una prueba global".

Se recuerda asimismo que el artículo 22.1 del Reglamento de Evaluación de Estudiantes (REVA) estipula que "el o la estudiante que se valga de conductas fraudulentas, incluida la indebida atribución de identidad o autoría, o esté en posesión de medios o instrumentos que faciliten dichas conductas, obtendrá la calificación de cero en el procedimiento de evaluación y, en su caso, podrá ser objeto de sanción, previa apertura de expediente disciplinario".