



1. Identificación

1.1. De la asignatura

Curso Académico	2025/2026
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 I
Código	1895
Curso	SEGUNDO SEGUNDO
Carácter	OBLIGATORIA
Número de grupos	4
Créditos ECTS	6.0
Estimación del volumen de trabajo	150.0 150.0
Organización temporal	1º Cuatrimestre 1º Cuatrimestre
Idiomas en que se imparte	Español

1.2. Del profesorado: Equipo docente

GARCIA MATEOS, GINES

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

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

Coordinador de la asignatura

Categoría

CATEDRATICOS DE UNIVERSIDAD

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

ginesgm@um.es Tutoría electrónica: No

Teléfono, horario y lugar de atención al alumnado

CAMPOS MARTINEZ, MANUEL

Docente: GRUPO 1

Coordinación de los grupos:

Categoría

CATEDRATICOS DE UNIVERSIDAD

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

manuelcampos@um.es Tutoría electrónica: No

Teléfono, horario y lugar de atención al alumnado

HERNANDEZ LOPEZ, JOSE ANTONIO

Docente: GRUPO 3

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: Sí

Teléfono, horario y lugar de atención al alumnado

HOYOS BARCELO, CARLOS

Docente: GRUPO 2, GRUPO 3

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 3**

Coordinación de los grupos: **GRUPO 3**

Categoría

PROFESORES TITULARES DE UNIVERSIDAD

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

fmontoya@um.es Tutoría electrónica: **No**

Teléfono, horario y lugar de atención al alumnado

TIGERAS GIL, EDUARDO

Docente: **GRUPO 3**

Coordinación de los grupos:

Categoría

PROFESOR CONTRATADO PARA SUSTITUCIONES

Área

LENGUAJES Y SISTEMAS INFORMÁTICOS

Departamento

INFORMÁTICA Y SISTEMAS

Correo electrónico / Página web / Tutoría electrónica

eduardo.tigerasg@um.es Tutoría electrónica: **No**

Teléfono, horario y lugar de atención al alumnado

2. Presentación

Los algoritmos y las estructuras de datos constituyen los dos pilares básicos de la programación de ordenadores. Esta asignatura se puede ver como una continuación natural de las asignaturas de programación de primer curso. 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 C++, ya que en caso contrario encontrarán muchas dificultades para superar la asignatura.

4. Competencias

4.1. Competencias básicas

- 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
- CB4: Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado

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

- 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
- CR8: Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados

5. Contenidos

5.1. Teoría

Tema 1: Abstracciones y Especificaciones

Introducción. Especificaciones informales. Especificaciones formales.

Tema 2: Conjuntos y diccionarios

Repaso del TAD Conjunto. Implementaciones básicas. El TAD Diccionario. Las tablas de dispersión.

Tema 3: Representación de conjuntos mediante árboles

Árboles Trie. Relaciones de equivalencia. Árboles de búsqueda balanceados. Árboles B.

Tema 4: Grafos

Introducción, notación y definiciones. Representación de grafos. Problemas y algoritmos sobre grafos.

5.2. Prácticas

■ Práctica 1: 1. Especificaciones formales con Maude

Relacionado con:

- Tema 1: Abstracciones y Especificaciones

■ Práctica 2: 2-3. Implementación y manejo de estructuras de datos

Relacionado con:

- Tema 2: Conjuntos y diccionarios
- Tema 3: Representación de conjuntos mediante árboles

■ Práctica 3: 4. Solución de problemas con grafos

Relacionado con:

- Tema 4: Grafos

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.		70.0	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.	10.0	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.	60.0	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.	10.0	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	0.0	0.0
Totales	150,00	

7. Horario de la asignatura

<https://www.um.es/web/estudios/grados/informatica/2025-26#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 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.	<p>Los alumnos pueden optar por realizar la evaluación de la teoría mediante el sistema de evaluación continua, o bien mediante el sistema alternativo de examen final.</p> <p>1 La evaluación continua consistirá en una serie de actividades a lo largo del curso para cada tema, y requerirá un mínimo de asistencias a clase (se tendrán en cuenta situaciones especiales particulares). Si algún tema no es superado, se podrá recuperar en el examen final. Los contenidos, criterios y temporización de las actividades de evaluación continua se presentarán el primer día de clase.</p> <p>2 En el sistema alternativo la nota de teoría se limitará al examen final, no siendo necesario hacer las actividades de evaluación continua.</p>	60.0

En cualquier caso, la nota final de la asignatura se calculará como: 60% de la nota de teoría (todos los temas tienen el mismo peso), más 40% de la nota de prácticas, siempre que ambas partes estén aprobadas.

Además, la nota final, una vez aprobada la asignatura, podrá verse incrementada hasta en un punto por participación en clase o por realización de soluciones programadas de ejercicios propuestos.

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.	Práctica en grupos de 2 alumnos, realizada en C++ usando el juez online Mooshak sobre el contenido de los temas 2 y 3. Se evalúa sobre 10 y se requiere un 5 para realizar media con la teoría. Puede requerirse una entrevista obligatoria con los alumnos.	40.0
-----	---	--	------

9. Fechas de exámenes

<https://www.um.es/web/estudios/grados/informatica/2025-26#exámenes>

10. Resultados del Aprendizaje

- Entender el desarrollo de programas como un proceso metódico e ingenieril, formado por una serie de etapas con distintos niveles de abstracción, frente a la idea de la programación como arte.
- Reconocer la importancia de la abstracción y conocer los tipos de abstracciones que aparecen en programación: funcional, de datos y de iteradores.
- Concienciarse de la utilidad de desarrollar especificaciones completas y precisas, entendiendo la especificación como un punto de acuerdo entre el usuario y el implementador de una abstracción.
- Comprender el método de especificación formal algebraico o axiomático (basado en una definición mediante axiomas) y el método constructivo u operacional (basado en el uso de precondiciones y postcondiciones).
- Conocer la importancia y ubicuidad de los tipos conjunto y diccionario en el desarrollo de programas, independientemente de la estructura que se use para implementarlos.
- Ser capaz de diseñar, implementar y analizar la eficiencia de las principales estructuras de representación no arbóreas para los tipos conjunto y diccionario, adaptando el diseño a las necesidades específicas de cada aplicación.
- Conocer la estructura de datos de tablas de dispersión, sus distintas variantes y los factores que influyen en su eficiencia y uso de memoria.
- Conocer y comprender una variedad de técnicas eficientes de representación de conjuntos y diccionarios mediante estructuras arbóreas.

- Adquirir la capacidad de evaluar las necesidades de representación de una aplicación específica, tomando decisiones justificadas sobre las estructuras de representación más adecuadas.
- Comprender la necesidad de usar mecanismos de equilibrado o balanceo para conseguir eficiencia en las representaciones arbóreas.
- Ser capaz de diseñar e implementar una estructura de datos para el tipo grafo (en sus distintas variantes) usando listas y matrices de adyacencia.
- Valorar críticamente las ventajas e inconvenientes de las representaciones de grafos mediante listas y matrices de adyacencia, y su influencia en la eficiencia de los algoritmos sobre grafos.
- Conocer y comprender el funcionamiento de una variedad de algoritmos clásicos sobre grafos (tales como los algoritmos de Prim, Kruskal, Dijkstra, Floyd y Warshall), razonando sobre las ideas subyacentes que aportan y analizando su complejidad computacional.
- Ser capaz de usar los algoritmos estudiados como herramientas prácticas para la resolución de problemas en un contexto genérico, a través de la transformación de un problema de interés en un problema sobre grafos.

11. Bibliografía

Bibliografía básica

- [Aho, Alfred V. -- Estructuras de datos y algoritmos \(1998\)](#)
- [Baase, Sara -- Computer algorithms : introduction to design and analysis \(2000\)](#)
- [Brassard, Gilles -- Fundamentos de algoritmia \(1997\)](#)
- [Cormen, T.H., et al. -- Introduction to algorithms, second edition. The MIT Press \(2001\)](#)
- [García Mateos, Ginés, et al. -- Algoritmos y estructuras de datos, Diego Marín \(2003\)](#)
- [Heileman, Gregory L. -- Estructuras de datos, algoritmos, programación orientada a o \(1998\)](#)
- [Horebeek, Ivo Van, et al. -- Algebraic specifications in software engineering: an introduction \(1989\)](#)
- [Kernighan, Brian W. -- El lenguaje de programación C \(1991\)](#)
- [Martí Oliet, Narciso -- Estructuras de datos y métodos algorítmicos : ejercicios res \(2004\)](#)
- [Mehlhorn, K -- Data Structures and algorithms, \(1984\)](#)
- [Peña Marí, Ricardo -- Diseño de programas : formalismo y abstracción \(1998\)](#)
- [Stroustrup, Bjarne -- El lenguaje de programación C++ \(1998\)](#)
- [Weiss, Mark Allen -- Estructuras de datos y algoritmos \(1995\)](#)
- [Wirth, Niklaus -- Algoritmos estructuras de datos = programas \(1994\)](#)

Bibliografía complementaria

- [Joyanes Aguilar, Luis., Estructura de datos : algoritmos, abstracción y objetos /\(1998\) ,McGraw-Hill,](#)

12. Observaciones

Criterio suspenso/no presentado:

Se pueden dar los siguientes casos:

1. Si un alumno suspende una parte (teoría o prácticas) y no se presenta a la otra parte, la calificación total será de Suspenso.
2. Si un alumno aprueba una parte y no presenta la otra parte, la calificación será de No presentado.
3. Si un alumno aprueba una parte y suspende la otra, la calificación será de Suspenso.

Cuando la calificación sea Suspenso, la nota se calculará como la media ponderada de la nota de teoría (60%) y de prácticas (40%), que se redondeará a 4,5 si dicha media fuera superior a 4,5.

El criterio para decidir si la nota de teoría está presentada o no, es la presentación en el examen final de la asignatura. Por lo tanto, un alumno que siga el método de evaluación continua pero no se presente al examen final para recuperar algún tema pendiente, se considerará como no presentado.

Las notas de partes aprobadas se guardarán durante las tres convocatorias del curso académico.

Otras actividades: olimpiada de programación:

Como actividad complementaria y voluntaria (aunque muy recomendable para los alumnos de la asignatura), se preparará a los alumnos para participar en el concurso internacional de programación International Collegiate Programming Contest (ICPC). Para ello se ofrecerá una preparación específica y habrá un concurso de programación (Olimpiada Murciana de Programación) a nivel local. Los alumnos ganadores conseguirán la participación en el concurso internacional (ICPC SWERC), que será considerado como un viaje de prácticas de la asignatura.

FECHAS APROXIMADAS DE LAS ACTIVIDADES EVALUATORIAS:

Ev. continua: Tema 1. Actividad de especificaciones formales: entrega 5ª semana.

Ev. continua: Temas 2 y 3. Examen teórico/práctico: 10ª semana.

Práctica en C/C++ sobre conjuntos y árboles: 12ª semana.

Ev. continua: Tema 4. Actividad entregable realizada en C/C++ sobre el contenido del tema 4 (grafos): 15ª semana.

Relación de los resultados de aprendizaje desarrollados en la asignatura con los ODS:

Por orden del Vicerrectorado de Responsabilidad Social y Cultura, se hace saber que 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".