PLAN DOCENTE DE LA ASIGNATURA: Análisis y Diseño de Algoritmos

CÓDIGO: 501307

CURSO ACADÉMICO: **2025/2026**

PLAN DOCENTE DE LA ASIGNATURA

Curso académico 2025-2026

Identificación y características de la asignatura					
Código	501307				
Denominación (español)	Análisis y Diseño de Algoritmos				
Denominación (inglés)	Analysis and Design of Computer Algorithms				
Titulaciones	Grado en Ingeniería Informática en Tecnologías de la Información (TI) PCEO Ing. en Telemática / Ing. Informática en TI				
Centro	Centro Universitario de Mérida				
Módulo	Común a la Rama de Informática				
Materia	Programación				
Carácter	Obligatoria	**ECTS**	6	**Semestre**	4º

Profesor/es			
Nombre	**Despacho**	**Correo-e**	**Página web**
Luis V. Calderita Estévez	8	lvcalderita@unex.es	Campus Virtual UEx

Área de conocimiento	Lenguajes y Sistemas Informáticos
Departamento	Ingeniería de Sistemas Informáticos y Telemáticos
Profesor coordinador (si hay más de uno)	

Competencias*

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

* Los apartados relativos a competencias, breve descripción del contenido, actividades formativas, metodologías docentes, resultados de aprendizaje y sistemas de evaluación deben ajustarse a lo recogido en la memoria verificada del título y en la normativa de evaluación (DOE 12 de diciembre de 2016)

✓	CB5 - Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía
Competencias generales	
✓	CG5 - Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en el anexo-2 de la Resolución de 8 de junio de 2009 de la Secretaría General de Universidades (BOE de 4 de Agosto de 2009) en el ámbito de las Tecnologías de la Información.
	CG6 - Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes de acuerdo con los conocimientos adquiridos según lo establecido en el anexo-2 de la Resolución de 8 de junio de 2009 de la Secretaría General de Universidades (BOE de 4 de Agosto de 2009) en el ámbito de las Tecnologías de la Información.
	CG8 - Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.
✓	CG9 - Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.
✓	CG10 - Conocimientos para la realización de mediciones, cálculos, valoraciones, tasaciones, peritaciones, estudios, informes, planificación de tareas y otros trabajos análogos de informática, de acuerdo con los conocimientos adquiridos según lo establecido en el anexo-2 de la Resolución de 8 de junio de 2009 de la Secretaría General de Universidades (BOE de 4 de Agosto de 2009) en el ámbito de las Tecnologías de la Información.
Competencias específicas	
✓	CE3 - Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.
✓	CE12 - 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.
✓	CE13 - Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.
	CE14 - Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
	CE20 - Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.
	CE21 - Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.
Competencias transversales	
	CT4. Resolución de problemas
	CT5. Toma de decisiones
	CT6. Orientación al aprendizaje
✓	CT11. Comunicación en lengua extranjera
	CT12. Diversidad e interculturalidad
✓	CT13. Resistencia y adaptación al entorno
✓	CT15. Comunicación interpersonal
Contenidos	
Breve descripción del contenido*	

Algoritmia. Complejidad computacional. Resolución de problemas computacionales.
Temario de la asignatura
Denominación del tema 1: Introducción a los algoritmos. Análisis de algoritmos: Contenidos del tema 1: • ¿Qué es un algoritmo? • Notaciones para describir algoritmos • Problemas, Instancias y soluciones • Tipos y clases de problemas • ¿Cómo medimos la eficiencia de un algoritmo? • Notaciones asintóticas • Análisis de algoritmos iterativos • Análisis de algoritmos recursivos

Denominación del tema 2: Divide y Vencerás.
Contenidos del tema 2:
• Introducción. Características Generales
• Eficiencia de los algoritmos DyV
• Ejemplos de aplicación
  o Búsqueda binaria
  o Multiplicación de enteros
  o Multiplicación de matrices

Denominación del tema 3: Algoritmos voraces, heurísticos y aproximados
Contenidos del tema 3:
• Introducción. Características generales
• Eficiencia de los algoritmos voraces
• Cuando usar algoritmos voraces
• Ejemplos de aplicación
  o El problema de la mochila
  o Árbol de recubrimiento mínimo
  o Caminos mínimos (Dijkstra)
  o Algoritmos heurísticos y aproximados
  o Heurístico para coloreado de grafos
  o Heurístico para el problema del viajante
  o Aproximado para el problema del viajante
  o Aproximado para el llenado de cajas

Denominación del tema 4: Programación dinámica
Contenidos del tema 4:
• Introducción a la Programación Dinámica
• Eficiencia de los algoritmos de Programación Dinámica
• Ejemplos de aplicación
  o Algoritmo para "dar cambio"
  o Problema de la mochila 0/1

Denominación del tema 5: Vuelta atrás
Contenidos del tema 5:
• Introducción a la Vuelta Atrás
• Eficiencia de los algoritmos VA
• Ejemplos de aplicación
  o Problema de la mochila 0/1
  o Problema de la asignación de trabajos
  o Las N Reinas
  o Laberinto

Denominación del tema 6: Ramificación y poda Contenidos del tema 6: • Introducción • Esquema genérico de un algoritmo RyP • Eficiencia de los algoritmos de ramificación y poda • Ejemplos de aplicación

- La mochila 0/1
- Asignación de trabajos

- Resolución de juegos

Denominación del tema 7: Elección del esquema algorítmico
Contenidos del tema 7:

- Resumen de los esquemas algorítmicos
- Caso de estudio: fontanero con penalización

En las sesiones de Seminario/Laboratorio se desarrollarán las siguientes actividades prácticas

- Seminario 1: Análisis de algoritmos
- Seminario 2: Divide y Vencerás
- Seminario 3: Algoritmos voraces
- Seminario 4: Programación dinámica
- Seminario 5: Vuelta atrás
- Seminario 6: Ramificación y poda

El seminario 1 evaluará y comparará la complejidad empírica de varios algoritmos de ordenación vistos en clase. Además, servirá de repaso tanto del entorno de desarrollo como de los conceptos vistos en las asignaturas "Fundamentos de programación" y "Estructuras de datos y de la información".

Los seminarios 2 a 6 plantearán problemas computaciones que se resolverán eficazmente empleando el esquema algoritmo correspondiente.

Actividades formativas*

Horas de trabajo del alumno por tema		Horas teóricas	Actividades prácticas				Actividad de seguimiento	No presencial
Tema	**Total**	**GG**	**CH**	**L**	**O**	**S**	**TP**	**EP**
0	1	1						
1	21	10				7	2	2
2	7	4						3
3	7	4						3
4	7	4						3
5	8	5						3
6	7	4						3
7	2	1					1	
S1	18				8			10
S2	12				2			10
S3	12				2			10
S4	12				2			10
S5	12				2			10
S6	12				2			10
Evaluación	12	2						10
Total	150	35			18	7	3	87

GG: Grupo Grande (85 estudiantes).
PCH: prácticas clínicas hospitalarias (7 estudiantes)
LAB: prácticas laboratorio o campo (15 estudiantes)
ORD: prácticas sala ordenador o laboratorio de idiomas (20 estudiantes)
SEM: clases problemas o seminarios o casos prácticos (40 estudiantes).

TP: Tutorías Programadas (seguimiento docente, tipo tutorías ECTS).
EP: Estudio personal, trabajos individuales o en grupo, y lectura de bibliografía.

Metodologías docentes*

Las metodologías docentes que se emplearán en el desarrollo de la asignatura serán las siguientes:

1. Clases expositivas de teoría y problemas: Presentación y discusión de los contenidos teóricos de la asignatura. Resolución de problemas computacionales en pseudocódigo.
2. Enseñanza participativa: Resolución de problemas computacionales en grupos medianos o pequeños.
3. Tutorización: Actividad de seguimiento para tutela de trabajos dirigidos, consultas de dudas y asesoría en grupos pequeños o individuales.
4. Aprendizaje autónomo. Mediante el estudio de la materia impartida y desarrollo de los supuestos prácticos planteados.
5. Aprendizaje virtual. Uso de herramientas virtuales de comunicación entre profesor y estudiante e incluso entre los estudiantes entre sí.

Resultados de aprendizaje*

Vinculados a competencias específicas:

- Analizar el coste computacional y memoria consumida de algoritmos iterativos y recursivos (CE3, CE12)
- Programar soluciones a problemas computacionales seleccionando, principalmente en términos de su eficiencia computacional, y utilizando el esquema algorítmico más adecuado. (CE3, CE12, CE13)

Vinculados a competencias transversales:

- Comunicarse con soltura de forma argumentada en otra lengua en textos de cierta complejidad (CT11)
- Mantener dinamismo y energía para seguir realizando las tareas en situaciones de presión, de tiempo, desacuerdo y dificultades (CT13)
- Utilizar el diálogo y el entendimiento para colaborar y generar relaciones (CT15)

Sistemas de evaluación*

Modalidad de Evaluación Continua

En la asignatura "Análisis de Algoritmos" se presta atención a:

- Análisis algoritmo
- Estrategias algorítmicas

Basándonos en lo anterior, presentó a continuación un cuadro que resume los elementos imprescindibles a tener en cuenta en la evaluación de la asignatura

Áreas	Criterios	Indicadores	Instrumentos
Análisis algoritmo	Estudio del coste computacional	Determina el coste computación tanto teórico como empírico de algoritmos iterativos y recursivos	1. Realización de trabajos dirigidos 2. Examen de certificación
Estrategias Algorítmicas	Planteamiento y resolución de	Resuelve problemas computacionales usando	1. Realización de trabajos dirigidos

	problemas computacionales	el esquema algorítmico más apropiado	

A continuación, se detallan los instrumentos/sistema se evaluación y el porcentaje que representa cada uno en la nota final:

Sistemas de evaluación	Porcentaje
Examen.	**20** (Entre el 0 y el 70%)
Exposición oral de trabajos realizados.	**0** (Entre el 0 y el 40%)
Realización de trabajos dirigidos (informes, casos prácticos, ejercicios y problemas).	**80** (Entre el 0 y el 80%)
Asistencia y/o participación en el aula, en el aula virtual, en las tutorías, etc.	**0** (Entre el 0 y el 30%)

- **Examen de certificación.** Se realiza al final, en el periodo de exámenes, y consiste en un examen escrito compuesto de preguntas de múltiple opción y problemas de análisis de algoritmos. Supone un **20%** de la nota final.
- **Trabajos dirigidos**. Supone un **80%** de la nota final. La calificación máxima a la que podrá aspirar cada estudiante en cada caso práctico dependerá del tiempo que tarde en resolverlo:

Antes de la fecha límite	Día 2	Día 3	Día 4	Día 5	Día 6	Día 7	Día 8	Día 9	Día 10	A partir del día 11
10	9,5	9	8,5	8	7,5	7	6,5	6	5,5	5

A partir de la fecha límite de entrega, con cada día de retraso se restará 0,5 puntos. Esta regla de puntuación no se aplica al trabajo dirigido del tema 01.

Se podrá exigir una defensa de cualquier trabajo dirigido. Cualquier sospecha de copia implica el suspenso de dicha práctica y notificación a los responsables académicos pertinentes. De igual modo, cualquier sospecha de copia sobre un trabajo dirigido entregado o parte del mismo, implicará inexorablemente el suspenso del trabajo dirigido, tanto a la persona que la copió cómo a la persona que permitió copiarla.

IMPORTANTE: Si el estudiante no se presenta al examen, la calificación final, independientemente de la calificación obtenida en los trabajos dirigidos, será de "No presentado".

En la convocatoria extraordinaria de Julio, si el sistema de evaluación aplicado al estudiante es "Continua" y la nota alcanzada en el aspecto "Trabajos dirigidos" no alcanza el 5 sobre 10, será evaluado siguiendo el itinerario "Global". En caso contrario, sólo se evaluará al estudiante del aspecto "Examen de certificación"

Modalidad de Evaluación Global

Se trata de un examen de certificación que se realiza al final y consiste en un examen escrito compuesto de preguntas de opción múltiple y preguntas de desarrollo donde se valorarán los conocimientos del programa teórico y práctico de la asignatura adquiridos por el estudiante. **La nota máxima que puede alcanzar el estudiante que opta por esta prueba final es de 10.**

Bibliografía

Bibliografía básica

1. Brassard G., Bratley P., Fundamentos de algoritmia. Prentice Hall, 1997.
2. Aho A.V., Hopcroft J.E., Ullman J.D., Estructuras de datos y algoritmos. Addison-Wesley, 1988.
3. Martí Oliet N., Ortega Mallén Y. y Verdejo López J.: Estructuras de datos y métodos algorítmicos. Ejercicios resueltos". Prentice Hall (2004).

Bibliografía complementaria

1. Cormen T.H., Leiserson C.E., Rivest R.L. y Stein C.: Introduction to Algorithms" (second edition). The MIT Press (2001).
2. Sartaj Sahni, Data Structures, Algorithms, and Applications in Java. Mc Graw Hill, 2000

Otros recursos y materiales docentes complementarios

1. Curso "Introduction to Algorithms" en abierto del MIT: https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/
2. Curso "Design and Analysis of Algorthims" en abierto del MIT: https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-design-and-analysis-of-algorithms-spring-2015

Recomendaciones

Se recomienda encarecidamente:

Tener superadas las asignaturas de Estructuras de Datos y de la Información y Fundamentos de Programación antes de cursar esta asignatura.

Observaciones:

En el horario aparecen 5 horas lectivas. 3 de teoría y 2 de prácticas. La hora de teoría del lunes se usa un número limitado de veces y no durante todo el curso.

En el horario aparecen dos grupos de prácticas. El grupo P. ADA-B es posible que no exista, por lo que no debéis tenerlo en cuenta al organizar vuestro horario. El P. ADA-A existirá siempre.