

# Algoritmos y Estructura de Datos

## Guía de Aprendizaje – Información al estudiante

### 1 Datos Descriptivos

Asignatura	Algoritmos y Estructura de Datos
Materia	Desarrollo de Software
Departamento responsable	LSIIS
Créditos ECTS	6
Carácter	Obligatoria
Titulación	Grado de Matemáticas e Informática por la Universidad Politécnica de Madrid
Curso	Segundo
Especialidad	No aplica
Curso académico	2011-2012
Semestre en el que se imparte	Septiembre a enero
Semestre principal	Septiembre a enero
Idioma en el que se imparte	Castellano
Página Web	<a href="http://web3.fi.upm.es/AulaVirtual/course/view.php?id=146">http://web3.fi.upm.es/AulaVirtual/course/view.php?id=146</a> <a href="http://lml.ls.fi.upm.es/aed">http://lml.ls.fi.upm.es/aed</a>

### 2 Profesorado

<b>NOMBRE Y APELLIDO</b>	<b>DESPACHO</b>	<b>Correo electrónico</b>
Pablo Nogueira (Coord.)	D-2304	pnogueira@fi.upm.es
Germán Puebla	D-2305	german@fi.upm.es
José Crespo	D-2310	jcrespo@fi.upm.es
Lars-Ake Fredlund	D-2309	lfredlund@fi.upm.es
Manuel Carro	D-2303	mcarro@fi.upm.es
Marta Patiño	D-2311	mpatino@fi.upm.es
Ricardo Jiménez	D-2311	rjimenez@fi.upm.es

### 3 Conocimientos previos requeridos para poder seguir con normalidad la asignatura

---

Asignaturas superadas Debe haber superado Programación I y Programación II

---

Otros resultados de aprendizaje necesarios Capacidad de modelizar y resolver matemáticamente problemas reales.

---

### 4 Objetivos de aprendizaje

Competencias asignadas a la asignatura y su nivel de adquisición

Código	Competencia	Nivel
CG01	Capacidad de resolución de problemas aplicando conocimientos de matemáticas, ciencias e ingeniería.	2
CG05	Capacidad de abstracción, análisis y síntesis.	2
CG06	Capacidad para trabajar dentro de un equipo, organizando, planificando, tomando decisiones, negociando y resolviendo conflictos, relacionándose, y criticando y haciendo autocrítica.	2
CG10	Capacidad para usar las tecnologías de la información y la comunicación.	2

LEYENDA: Nivel de adquisición 1: Básico Nivel de adquisición 2: Medio Nivel de adquisición 3: Alto

Código	Competencia	Nivel
CE07	Conocer los cimientos esenciales y fundacionales de la informática, subrayando los aspectos esenciales de la disciplina que permanecen inalterables ante el cambio tecnológico.	3
CE09	Capacidad de elegir y usar los métodos analíticos y de modelización relevantes, y de describir una solución de forma abstracta.	3
CE11	Comprender intelectualmente el papel central que tienen los algoritmos y las estructuras de datos, así como una apreciación del mismo.	4

CE13	Poseer destrezas fundamentales de la programación que permitan la implementación de los algoritmos y las estructuras de datos en el software.	4
CE14	Poseer las destrezas que se requieren para diseñar e implementar unidades estructurales mayores que utilizan los algoritmos y las estructuras de datos, así como las interfaces por las que se comunican estas unidades.	4
CE43	Capacidad para trabajar de forma efectiva como individuo, organizando y planificando su propio trabajo, de forma independiente o como miembro de un equipo.	3

LEYENDA: Nivel de adquisición 1: Conocimiento Nivel de adquisición 2: Comprensión  
 Nivel de adquisición 3: Aplicación Nivel de adquisición 4: Análisis y síntesis

## 5 Resultados de aprendizaje

### 5.1 Resultados de aprendizaje de la asignatura

Código	Resultado de aprendizaje	Competencias asociadas	Nivel de adquisición
RA1	Programar aplicaciones mediante librerías existentes de TADs, iteradores, etc.	CE13, CE14 CE43 Todas las CG	4
RA2	Resolver problemas algorítmicos no triviales	CE07, CE09, CE11, CE13, CE43 Todas las CG	4
RA3	Razonar sobre la complejidad algorítmica	CE07, CE09, CE11, CE13, CE43 Todas las CG	2
RA4	Razonar sobre la terminación de algoritmos y programas.	CE07, CE09, CE11, CE13, CE43 Todas las CG	2
RA5	Usar y definir estructuras de datos eficientes y adecuadas a cada	CE07, CE09, CE11, CE13,	4

problema

CE14, CE43

Todas las CG

## 5.2 Sistema de evaluación de la asignatura

### 5.2.1 Indicadores de logro

Ref	Indicador	Relación con RA
I1	Sabe elegir de entre varias opciones el TAD más adecuado para resolver un problema.	RA5
I2	Comprende las implicaciones prácticas de los diferentes niveles de complejidad asintótica.	RA3
I3	Comprende y sabe calcular la complejidad asintótica de algoritmos y programas.	RA3
I4	Demuestra informalmente la terminación de un algoritmo o un programa.	RA4
I5	Comprende el diseño de TADs relativamente avanzados.	RA5
I6	Es capaz de implementar TADs relativamente avanzados.	RA5
I7	Conoce y utiliza las colecciones estándar de Java.	RA1
I8	Comprende y utiliza librerías de código de tamaño considerable que definen e implementan TADs avanzados.	RA1
I9	Comprende y sabe aplicar algoritmos de búsqueda y de ordenación, generando código apropiado.	RA2
I10	Comprende y sabe aplicar técnicas algorítmicas de divide y vencerás, búsqueda con retroceso y selección rápida, generando código apropiado	RA2

### 5.2.2 Evaluación sumativa

Breve descripción de las actividades evaluables	Momento	Lugar	Peso en la calificación
Exámenes de teoría	Tres en el semestre (evaluación continua) o uno en el semestre (evaluación prueba final)	Aulas de evaluación	40%

---

Ejercicios de laboratorio	Casi todas las semanas	Aula informática	35%
Práctica	Una en el semestre	Aulas de ordenadores o en casa.	25%

---

## 6 Criterios de calificación

### 6.1 Introducción

El año académico se divide en dos periodos de matriculación o semestres: primer semestre de septiembre a enero y segundo semestre de febrero a junio. Las normas de esta sección se aplican a los dos semestres.

Los criterios de evaluación de la asignatura se establecen en conformidad con la "Normativa Reguladora de los Sistemas de Evaluación" (en adelante "Normativa Reguladora") actualmente vigente en la Universidad Politécnica de Madrid para los planes de estudio adaptados al R.D. 1393/2007.

Según dicha normativa, por cada periodo de matrícula de las asignaturas (es decir, por cada semestre) se establecen dos convocatorias de evaluación:

1. Convocatoria ordinaria, que se corresponde con las actividades de evaluación que se realizan durante el desarrollo de la asignatura en cada semestre y, en su caso, en el periodo inmediatamente posterior a su finalización que se fije en el calendario académico de la universidad.

Cada semestre tiene su convocatoria ordinaria a la que concurren los alumnos matriculados en el semestre.

2. Convocatoria extraordinaria, que se corresponde con las actividades de evaluación que deben realizar aquellos estudiantes que no logren superar la asignatura en la convocatoria ordinaria.

La convocatoria extraordinaria tiene lugar en el mes de julio y pueden concurrir a ella los alumnos matriculados en cualquiera de los semestres del año académico que no hayan superado la asignatura en la convocatoria ordinaria de su semestre.

A continuación se detallan los normas y sistemas de evaluación para cada convocatoria.

### 6.2 Convocatoria ordinaria

#### 6.2.1 Sistemas de evaluación

Según el Artículo 20 de la Normativa Reguladora, en la convocatoria ordinaria el alumno puede optar únicamente por uno de los siguientes sistemas de evaluación:

1. Sistema de evaluación continua.
2. Sistema de evaluación mediante prueba final.

El sistema de evaluación continua será el que se aplique en general a todos los alumnos de la asignatura.

El alumno que desee seguir el sistema de evaluación mediante prueba final deberá comunicarlo por escrito a la Jefatura de Estudios de la Facultad de Informática según las

normas, pautas y plazo que determine dicha Jefatura.

## 6.2.2 Sistema de evaluación continua

- Actividades evaluables

Se evalúa al alumno de forma continua a lo largo del semestre mediante las siguientes actividades:

1. Tres exámenes de teoría.

Cada examen abarca una parte del temario de teoría. La parte evaluada por un examen no será evaluada de nuevo por los otros exámenes dentro de la convocatoria ordinaria.

Las fechas de realización y las partes del temario correspondientes a cada examen se indicarán con suficiente antelación de acuerdo a la Normativa Reguladora.

Los exámenes se realizarán en general en el horario de Actividades de Evaluación del semestre, aunque podrá recurrirse a otros horarios, como por ejemplo, el horario de actividades de laboratorio, semanas destinadas al proceso de evaluación en el calendario docente, etc.

Cada examen de teoría se evalúa de 0 a 10. La nota media de los tres exámenes de teoría debe ser al menos 4.5. De lo contrario la calificación final de la asignatura para la convocatoria ordinaria será "suspenseo".

2. Ejercicios de laboratorio.

Se realizarán en las Aulas Informáticas en el horario establecido. Cada ejercicio consiste en la resolución de problemas de programación con algoritmos y estructuras de datos.

Es obligatoria la asistencia a al menos 80% de las clases de laboratorio. De lo contrario, la calificación final de la asignatura para la convocatoria ordinaria será "suspenseo".

Los ejercicios de laboratorio se realizarán en grupos de dos alumnos.

Para poder ser calificados, los ejercicios de laboratorio deben superar las pruebas del sistema de entregas. De no superarlas, el ejercicio se calificará como "no aceptado" y la calificación final de la asignatura para la convocatoria ordinaria será "suspenseo".

Para optar a la máxima nota, los ejercicios deben haber sido aceptados por el sistema de entrega antes de finalizar la clase de laboratorio. Los ejercicios aceptados con posterioridad tendrán una reducción en su nota del 20% por cada 24 horas a partir de la finalización de la clase de laboratorio. Llegado al 100% de penalización se puede seguir entregando el ejercicio pero la nota máxima del mismo será 0.

3. Una práctica.

Consiste en un ejercicio práctico de programación de algoritmos y estructuras de datos a desarrollar a lo largo de varias semanas.

La práctica debe realizarse de forma individual. Para poder ser calificada, la práctica debe superar las pruebas del sistema de entrega. De no superarlas la

práctica se considera no presentada y la calificación final de la asignatura para la convocatoria ordinaria será "suspenso".

Para optar a la máxima nota la práctica debe ser aceptada antes de la fecha y hora tope estipulada en su enunciado. Las prácticas aceptadas con posterioridad tendrán una reducción en su nota acorde al criterio establecido en el enunciado de la práctica.

- **Criterios de calificación**

- Los alumnos deben entregar los tres exámenes de teoría y la nota media de los tres debe ser al menos 4.5. Los alumnos deben asistir al 80% de las clases de laboratorio, tener aceptados todos los ejercicios de laboratorio y la práctica. De no cumplirse alguno de estos requisitos la calificación final de la asignatura para la convocatoria ordinaria será "suspenso", excepto en el caso de que no se entregue nada durante el semestre, en cuyo caso la calificación de la asignatura para la convocatoria ordinaria será "no presentado".
- No hay posibilidad de repetir o recuperar exámenes de teoría, ejercicios de laboratorio o la práctica.
- Todos los ejercicios de laboratorio y la práctica serán sometidos a un sistema de detección de plagio. En caso de verificarse el plagio, los alumnos involucrados tendrán la asignatura suspensa para el año académico.
- Cumplidos los requisitos anteriores, la nota de la asignatura se calcula usando la siguiente fórmula:

$$\text{NotaFinal} = 0.4 * T + 0.35 * L + 0.25 * P$$

donde T es la nota media de los exámenes de teoría, L es la nota media de los ejercicios de laboratorio y P es la nota de la práctica.

### **6.2.3 Sistema de evaluación mediante prueba final**

Se evalúa a los alumnos como en el sistema de evaluación continua en lo tocante a laboratorios y práctica, con la salvedad de que realizan un único examen de teoría al final del semestre que abarca todo el temario de la asignatura.

Además de asistir al 80% de las clases de laboratorio y tener aceptados todos los ejercicios de laboratorio y la práctica, los alumnos deben entregar el examen final y la nota del mismo debe ser al menos 4.5. De no cumplirse alguno de estos requisitos la calificación final de la asignatura para la convocatoria ordinaria será "suspenso".

Las normas y los criterios de calificación para los ejercicios de laboratorio y para la práctica son los mismos que los del sistema mediante evaluación continua.

La nota de la asignatura se calcula usando la misma fórmula que para el sistema de evaluación continua, con la salvedad de que T será la nota del examen final.

### **6.2.4 Liberación de bloques**

Independientemente del sistema de evaluación elegido, no se guardan las notas de ningún bloque (teoría, laboratorio, práctica) para otras convocatorias.

## **6.3 Convocatoria extraordinaria**

Los alumnos que no han superado la asignatura en la convocatoria ordinaria, independientemente del semestre del año académico cursado y del sistema de evaluación

elegido para la convocatoria ordinaria, tienen la posibilidad de concurrir a la convocatoria extraordinaria del mes de julio. En esta convocatoria se evalúa la asignatura completa, debiendo superar los alumnos las siguientes pruebas:

1. Un examen de teoría que abarca todo el temario de la asignatura.
2. Un ejercicio de laboratorio en el Aula Informática de temática similar a los propuestos en el semestre.
3. Una práctica cuyo enunciado, fecha y hora tope de entrega se anunciará con suficiente antelación.

Los alumnos deben entregar el examen de teoría y la nota del mismo debe ser al menos 4.5. Deben tener aceptado por el sistema de entrega el ejercicio de laboratorio y la práctica, esta última antes de la fecha y hora tope establecida. De no cumplirse alguno de estos requisitos la calificación final de la asignatura para el año académico será "suspense", excepto en el caso de que no se entregue nada en cuyo caso será "no presentado".

La nota de la asignatura se calcula usando la siguiente fórmula:

$$\text{NotaFinal} = 0.4 * T + 0.35 * L + 0.25 * P$$

donde T es la nota del examen de teoría, L es la nota del ejercicio de laboratorio y P es la nota de la práctica.

## 7 Contenidos y Actividades de Aprendizaje

Contenidos específicos

<b>Bloque/Tema/Capítulo</b>	<b>Apartado</b>	<b>Indicadores relacionados</b>
Bloque 1: Introducción	1.1 Repaso de programación orientada a objetos en Java	I1
	1.2 Listas clásicas.	I1 I5 I8
Bloque 2: TADs secuenciales, árboles, e iteradores	2.1 Listas indexadas	I1 I5 I6 I8
	2.2 Listas enlazadas	I1 I5 I6 I8
	2.3 Secuencias y conjuntos	I1 I5 I6 I8
	2.4 Complejidad de algoritmos	I2 I3 I4
	2.5 Iteradores	I3 I4 I6
	2.6 Colecciones de la JCF	I5 I7
	2.7 Árboles generales	I1 I3 I4 I5 I6 I8
	2.8 Árboles binarios	I1 I3 I4 I5 I6 I8
Bloque 3: TADs con manejo de prioridades	3.1. Colas con prioridad, comparadores y ordenación.	I1 I3 I4 I5 I6 I8
	3.2 Montículos y ordenación	I1 I3 I4 I5 I6 I8
Bloque 4:	4.1 Funciones finitas	I1 I3 I4 I5 I6 I8

Funciones finitas y diccionarios	4.2 Tablas de dispersión	I1 I3 I4 I5 I6 I8
	4.3 Diccionarios	I1 I3 I4 I5 I6 I8
	4.2 Funciones finitas con dominio ordenado	I1 I3 I4 I5 I6 I8
	4.3 Árboles binarios de búsqueda	I1 I3 I4 I5 I6 I8
	4.4 Árboles AVL	I1 I3 I4 I5 I6 I8
	4.5 Árboles multicamino, (2,4), (a,b) y B	I1 I3 I4 I5 I6 I8
<hr/>		
Bloque 5: Algoritmos y diseño de algoritmos	5.1 Algoritmos de ordenación básicos	I2 I3 I4 I9
	5.2 Técnica divide y vencerás	I2 I3 I4 I10
	5.4 Ordenación por mezcla y ordenación rápida.	I2 I3 I4 I8 I9 I10
	5.5 Búsqueda con retroceso y selección rápida	I2 I3 I4 I8 I10

---

## 8 Breve descripción de las modalidades organizativas y los métodos de enseñanza empleados

---

CLASES DE TEORÍA	Método expositivo y estudio de casos.
CLASES DE PROBLEMAS	Ejercicios de laboratorio. Aprendizaje basado en resolución de problemas.
PRÁCTICAS	Práctica. Aprendizaje basado en resolución de problemas.
TRABAJOS AUTÓNOMOS	Práctica y ejercicios voluntarios.
TRABAJOS EN GRUPO	Ejercicios de laboratorio.
TUTORÍAS	Atención personalizada en teoría, laboratorio y práctica.

---

## 9 Recursos didácticos

---

BIBLIOGRAFÍA M. T. Goodrich y R. Tamassia,  
*Data Structures and Algorithms in Java*  
International Student Version, 5ed, Wiley.

---

T. Cormen, C. Leiserson, R. Rivest y C. Stein,  
"Introduction to Algorithms" 2ed edición, MIT Press.

---

P. Sestoft, *Java Precisely*, 2ed, MIT Press.

---

M. Augenstein, Y. Langsam, A. M. Tenenbaum,  
*Data Structures Using Java*, 2003, Prentice Hall

---

A. Aho, J. Hopcroft, J. Ullman,  
*Data Structures and Algorithms* Addison-Wesley

---

RECURSOS WEB Sitio Moodle de la asignatura

<http://web3.fi.upm.es/AulaVirtual/course/view.php?id=146>

Java Language Specification, 3rd Edition,

<http://java.sun.com/docs/books/jls>.

Java Platform SE 6, incluye JCF

<http://download.oracle.com/javase/6/docs/api/index.html>.

Librería del libro de la asignatura:

<http://net3.datastructures.net/download.html> (código).

<http://net3.datastructures.net/doc5/index.html> (Javadoc).

<http://ww0.java5.datastructures.net/> (ejemplos, transparencias).

---

EQUIPAMIENTO Aulas de clase con proyector y pizarra. Aula informática con proyector, pizarra y ordenadores para los alumnos.

Compiladores y JRE de Java versión 1.6, entorno de desarrollo integrado (IDE) Eclipse.

---

## 10 Cronograma de trabajo

Semana	Actividades en Aula	Actividades en Aula Informática	Trabajo Individual	Trabajo en Grupo	Actividades de Evaluación	Otros
Semana 1	POO en Java, listas clásicas, listas indexadas (2 horas)		Estudio teoría			
(8 horas)			y repaso conceptos básicos anteriores (6 horas)			
Semana	Listas	Laboratorio	Estudio			

2	indexadas, enlazadas y secuencias		teoría	
(9 horas)	(2 horas)	(2 horas)	y repaso ejercicios	
			de laboratorio (5 horas)	
Semana 3	Conjuntos, iteradores, complejidad	Laboratorio	Estudio teoría	
(9 horas)	(2 horas)	(2 horas)	y repaso ejercicios	
			de laboratorio (5 horas)	
Semana 4	Árboles generales y binarios	Laboratorio	Estudio teoría	
(9 horas)	(2 horas)	(2 horas)	y repaso ejercicios	
			de laboratorio (5 horas)	
Semana 5	Colas con prioridad y ordenación		Preparación examen (6 horas)	Primer examen de teoría
(9 horas)	(2 horas)			(1 hora)
Semana 6	Montículos y ordenación	Laboratorio	Estudio teoría	
(10 horas)	(2 horas)	(2 horas)	y repaso ejercicios	
			de laboratorio (2 horas)	
			Realización de práctica (4 horas)	
Semana 7	Funciones finitas y tablas de dispersión	Laboratorio	Estudio teoría	
(10 horas)	(2 horas)	(2 horas)	y repaso ejercicios	
			de	

			laboratorio (2 horas)	
			Realización de práctica (4 horas)	
Semana 8	Funciones finitas y diccionarios	Laboratorio	Estudio teoría y	
(10 horas)	con dominio ordenado (2 horas)	(2 horas)	repaso ejercicios	
			de laboratorio (2 horas)	
			Realización de práctica (4 horas)	
Semana 9	Colecciones de la JCF,	Laboratorio	Estudio teoría y	
(10 horas)	implementación de diccionarios (2 horas)	(2 horas)	repaso ejercicios	
			de laboratorio (2 horas)	
			Realización de práctica (4 horas)	
Semana 10	Árboles binarios de búsqueda		Preparación examen (4 horas)	Segundo examen de teoría
(10 horas)	(2 horas)			(1 hora)
			Realización de práctica (3 horas)	
Semana 11	Árboles binarios de búsqueda	Laboratorio	Estudio teoría y	
(10 horas)	(2 horas)	(2 horas)	repaso ejercicios	
			de laboratorio (2 horas)	

			Realización de práctica (4 horas)	
Semana 12	Árboles AVL	Laboratorio	Estudio teoría y repaso ejercicios de laboratorio (2 horas)	
(10 horas)	(2 horas)	(2 horas)	Realización de práctica (4 horas)	
Semana 13	Árboles multicamino, (2,4), (a,b) y B.	Laboratorio	Estudio teoría y repaso ejercicios de laboratorio (2 horas)	
(10 horas)	(2 horas)	(2 horas)	Realización de práctica (4 horas)	
Semana 14	Algoritmos de ordenación, DyV,	Laboratorio	Estudio teoría y repaso ejercicios de laboratorio (2 horas)	Entrega de la práctica
(10 horas)	ordenación por mezcla (2 horas)	(2 horas)	Realización de práctica (4 horas)	
Semana 15	Ordenación rápida y búsqueda con retroceso		Estudio teoría y repaso ejercicios de laboratorio	
(10 horas)	(2 horas)			

		(8 horas)	
Semana 16	Búsqueda con retroceso y selección rápida	Estudio teoría y	
(10 horas)	(2 horas)	repaso ejercicios	
		de laboratorio	
		(8 horas)	
Semana de		Estudio teoría (7 horas)	Tercer examen de teoría
Examen			(1 hora)
(8-10 horas)			Examen final (3 horas)

Nota: Para cada actividad se especifica la dedicación en horas que implica para el alumno.