

# ALED: ALgoritmos y Estructuras de Datos

## Guía de Aprendizaje – Información al estudiante

### 1. Datos Descriptivos

<b>Asignatura</b>	Algoritmos y Estructuras de Datos
<b>Materia</b>	Computación
<b>Departamento responsable</b>	Departamento de Ingeniería de Sistemas Telemáticos
<b>Créditos ECTS</b>	6
<b>Carácter</b>	Obligatoria
<b>Titulación</b>	Grado en Ingeniería Biomédica
<b>Curso</b>	Tercero
<b>Especialidad</b>	No aplica

<b>Curso académico</b>	2013-2014
<b>Semestre en que se imparte</b>	Primero
<b>Idioma en que se imparte</b>	Castellano
<b>Página Web</b>	<a href="http://www.dit.upm.es">http://www.dit.upm.es</a>

## 2. Profesorado

NOMBRE Y APELLIDO	DESPACHO	Correo electrónico
Jose María del Álamo Ramiro (Coordinador)	B-204.1	<a href="mailto:jmdela@dit.upm.es">jmdela@dit.upm.es</a>
Miguel Ángel de Miguel Cabello	B-323.1	<a href="mailto:mmiguel@dit.upm.es">mmiguel@dit.upm.es</a>
Juan Antonio de la Puente Alfaro	B-318	<a href="mailto:jpuente@dit.upm.es">jpuente@dit.upm.es</a>

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

Asignaturas superadas	N/A
Otros resultados de aprendizaje necesarios	Fundamentos de Programación

## 4. Objetivos de Aprendizaje

COMPETENCIAS ASIGNADAS A LA ASIGNATURA Y SU NIVEL DE ADQUISICIÓN		
Código	Competencia	Nivel
CG1-18	Todas las asignaturas del Plan de Estudios contribuyen en mayor o menor medida a la consecución de las competencias generales del perfil de egreso.	1
CE19	Capacidad para escribir programas utilizando los recursos de programación más habituales y aplicarlos a problemas de ingeniería.	3
CE20	Conocer y comprender los fundamentos de la informática, los principios de la arquitectura de computadores y manejar los sistemas operativos más comunes.	2
CE21	Conocer, comprender y utilizar herramientas informáticas para la resolución de problemas matemáticos y de simulación de sistemas.	2
CE22	Saber desarrollar algoritmos para la resolución de problemas informáticos en Ingeniería Biomédica.	2

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



RESULTADOS DE APRENDIZAJE DE LA ASIGNATURA			
Código	Resultado de aprendizaje	Competencias asociadas	Nivel de adquisición
RA1	Solucionar problemas mediante la programación de ordenadores.	CE19	3
RA2	Conocer los principios básicos del análisis y diseño de algoritmos y aplicarlos a algoritmos representativos.	CE21 CE22	3
RA3	Conocer los principios del análisis de complejidad de algoritmos y aplicarlos a algoritmos representativos.	CE21 CE22	2
RA4	Conocer los conceptos básicos de la programación concurrente y los mecanismos de sincronización fundamentales.	CE19 CE20	3
RA5	Conocer y comprender los problemas que pueden aparecer en programas concurrentes y las estrategias básicas para evitarlos	CE19 CE20	2
RA6	Conocer un proceso de desarrollo de software para el desarrollo de aplicaciones en movilidad.	CE19	1
RA7	Conocer y comprender el modelo cliente-servidor y distintas arquitecturas de sistemas telemáticos.	CE19 CE20	2
RA8	Conocer, comprender y aplicar los fundamentos de aplicaciones en movilidad, así como tecnologías necesarias para su implementación.	CE19 CE20	2

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

## 5. Sistema de evaluación de la asignatura

INDICADORES DE LOGRO		
Ref	Indicador	Relacionado con RA
I1	Escribir, compilar ejecutar y probar un programa sencillo con el entorno Eclipse y JUnit.	RA1, RA6
I2	Comprender los algoritmos básicos de búsqueda y sus propiedades.	RA2
I3	Saber codificar algoritmos básicos de búsqueda y ser capaces de diseñar variaciones de los mismos.	RA2
I4	Determinar la complejidad de algunos algoritmos representativos.	RA3
I5	Comparar la complejidad de algoritmos con la misma función.	RA3
I6	Realizar programas concurrentes con hebras independientes.	RA4
I7	Identificar correctamente las regiones críticas en un programa con variables compartidas entre varias hebras.	RA4
I8	Utilizar correctamente los mecanismos de sincronización condicional entre hebras.	RA4
I9	Conocer los problemas de bloqueos que se pueden presentar en programas concurrentes y las condiciones para su aparición.	RA5
I10	Saber desarrollar, empaquetar, desplegar y depurar una aplicación en movilidad.	RA6, RA8
I11	Comprender la arquitectura de una aplicación en movilidad.	RA7
I12	Conocimiento y uso de las facilidades ofrecidas por HTML5 y CCS3 para desarrollo de aplicaciones en movilidad.	RA6, RA8
I13	Saber diseñar interfaces de usuario multi-dispositivo.	RA8



EVALUACION SUMATIVA			
Breve descripción de las actividades evaluables	Momento	Lugar	Peso en la calif.
Prácticas	Quincenales	Moodle	35%
Examen Parcial 1	Noviembre	Aulas	40%
Examen Parcial 2	Enero	Aulas	25%
			<b>Total: 100%</b>



## CRITERIOS DE CALIFICACIÓN

En **convocatoria ordinaria**, los alumnos serán evaluados normalmente mediante evaluación continua. No obstante, en cumplimiento de la Normativa de Evaluación de la Universidad Politécnica de Madrid, los alumnos que lo deseen serán evaluados mediante un único examen final, siempre y cuando lo comuniquen al Director del Departamento de Ingeniería de Sistemas Telemáticos mediante solicitud presentada en el registro de la Escuela Técnica Superior de Ingenieros de Telecomunicación antes de la fecha del primer examen parcial. Esta opción supone la **renuncia a la evaluación continua**.

En la **convocatoria extraordinaria** habrá un único examen de la asignatura completa en el horario asignado al examen final, que evaluará la adquisición de las competencias fijadas en la Guía de Aprendizaje, y en el que el alumno deberá obtener una calificación mayor o igual a 5 puntos sobre un total de 10.

La **evaluación continua** se realizará de la siguiente manera:

- la asignatura se aprobará cuando se obtenga una calificación mayor o igual a 5 puntos sobre un total de 10 en la nota final, que se obtendrá mediante la suma ponderada de las calificaciones correspondientes a las diferentes actividades de evaluación, con los siguientes pesos:
  - Examen parcial 1 – 40%
  - Examen parcial 2 – 25%
  - Prácticas – 35%
- Para aprobar la asignatura, es necesario tener unos conocimientos mínimos que se deberán poner de manifiesto tanto en los exámenes parciales como en las prácticas propuestas. Será necesario superar el umbral de 4 puntos sobre 10 en cada uno de estos bloques de evaluación para aprobar la asignatura.
- La **evaluación continua** de la asignatura tendrá una entrega periódica obligatoria de una práctica. La no presentación de una práctica en plazo y forma supone obtener una calificación de 0 puntos sobre 10 en esa práctica. La nota de la parte de prácticas se obtendrá como la media de las notas obtenidas en cada práctica de la asignatura.
- Todos los exámenes y prácticas que se realicen deben ser fruto del trabajo personal del alumno, aunque se fomentará la discusión y el trabajo en grupo para ayudar a entender mejor los problemas que se intentan resolver. **La copia de exámenes o trabajos prácticos supondrá el suspenso de la asignatura de forma automática, tanto para quien copia como para quien se deja copiar.**

## 6. Contenidos y Actividades de Aprendizaje

CONTENIDOS ESPECÍFICOS		
Bloque / Tema / Capítulo	Apartado	Indicador5es Relacionados
Tema 1: Introducción	1.1 Introducción a la asignatura.	I1
	1.2 Repaso de técnicas de programación.	
	1.3 Ciclo de vida, pruebas y depuración de programas.	
Tema 2: Diseño de algoritmos	2.1 Algoritmos recursivos	I1
	2.2 Complejidad de los algoritmos: conceptos básicos y familias de algoritmos.	I2
	2.3 Algoritmos de búsqueda. Complejidad de algoritmos de búsqueda.	I3, I4
	2.4 Optimización: memoria frente a tiempo de ejecución.	I5
Tema 3: Programación Concurrente	3.1 Programas secuenciales y concurrentes. Hebras (threads).	I6
	3.2 Interacción entre hebras. Variables comunes, exclusión mutua y comunicación sincronizada. Monitores.	I7, I8
	3.3 Propiedades de programas concurrentes: seguridad, vivacidad, equidad. Bloqueos.	I9
Tema 4: Aplicaciones en movilidad.	4.1 Web y arquitecturas de referencia.	I11
	4.2 Lado servidor.	I10
	4.3 Lado cliente: HTML/HTML5, CSS, JavaScript	I12, I13



## 7. Breve descripción de las modalidades organizativas utilizadas y de los métodos de enseñanza empleados

<b>CLASES DE TEORÍA</b>	Se utilizará la lección magistral para la exposición de contenidos, explicación de conocimientos, demostraciones, presentación de experiencias, etc., y con apoyo de recursos audiovisuales.
<b>CLASES DE PROBLEMAS</b>	Se resolverán en clase ejercicios que servirán para aplicar los conocimientos adquiridos en las clases de teoría.
<b>PRÁCTICAS</b>	Se implementarán en el laboratorio una serie de ejercicios prácticos que cumplirán unas determinadas especificaciones.
<b>TRABAJOS AUTÓNOMOS</b>	El alumno realizará ejercicios para practicar y afianzar los conocimientos aprendidos.
<b>TRABAJOS EN GRUPO</b>	Algunas prácticas y actividades se realizarán en grupo para promover el aprendizaje cooperativo.
<b>TUTORÍAS</b>	El alumno podrá hacer uso de tutorías personalizadas dentro de los horarios establecidos.

## 8. Recursos didácticos

<b>BIBLIOGRAFÍA</b>	Data structures & Problems Solving using Java. Weiss, 2010
	Head first Java. K. Sierra. O'Reilly, 2005
	Java Threads. Scott Oaks, Henry Wong. O'Reilly, 2004
<b>RECURSOS WEB</b>	Página web de la asignatura: <a href="http://www.lab.dit.upm.es/aled">http://www.lab.dit.upm.es/aled</a>
	Sitio Moodle de la asignatura: <a href="http://moodle.lab.dit.upm.es/">http://moodle.lab.dit.upm.es/</a>
	Tutoriales, herramientas y almacenes de software accesibles a través del sitio moodle de la asignatura.
<b>EQUIPAMIENTO</b>	Laboratorio A-127 / B-123
	Aula: Asignada por Jefatura de Estudios

## 9. Cronograma de trabajo de la asignatura

Semana	Actividades en Aula	Actividades en Laboratorio	Trabajo Individual	Trabajo en Grupo	Actividades de Evaluación	Otros
Semana 1 (8h)	Tema 1 (4h)		Estudio (2h) Ejercicios (2h)			
Semana 2 (10h)	Tema 1 (2h).	Laboratorio 1 (2h)	Estudio (2h) Práctica (4h)		Entrega de práctica	
Semana 3 (10h)	Tema 1. (2h) Tema 2 (2h)		Estudio (4h) Ejercicios (2h)			
Semana 4 (10h)	Tema 2 (2h)	Laboratorio 2 (2h)	Estudio (2h) Práctica (4h)		Entrega de práctica	
Semana 5 (10h)	Tema 2 (4h)		Estudio (4h) Ejercicios (2h)			
Semana 6 (10h)	Tema 2 (2h)	Laboratorio 3 (2h)	Estudio (2h) Práctica (4h)		Entrega de práctica	
Semana 7 (10h)	Tema 3 (4h)		Estudio (4h) Ejercicios (2h)			
Semana 8 (10h)	Tema 3 (2h)	Laboratorio 4 (2h)	Estudio (2h) Práctica (4h)		Entrega de práctica	
Semana 9 (10h)	Tema 3 (4h)		Estudio (4h) Ejercicios (2h)			

Semana 10 (12 h)	Tema 3 (2h)	Laboratorio 5 (2h)	Estudio (2h) Práctica (4h) Estudio (2h)		Entrega de práctica	
Semana 11 (10 h)	Tema 4 (2h)		Estudio (6h)		Examen parcial (2h)	
Semana 12 (10 h)	Tema 4 (2h)	Laboratorio 6 (2h)	Estudio (2h) Práctica (4h)		Entrega de práctica	
Semana 13 (10 h)	Tema 4 (4h)		Estudio (4h) Ejercicios (2h)			
Semana 14 (10 h)	Tema 4 (2h)	Laboratorio 7 (2h)	Estudio (2h) Práctica (4h)		Entrega de práctica	
Semana 15 (10 h)	Tema 4 (4h)		Estudio (6h)			
Semana 16 (10 h)	Tema 4 (2h)		Estudio (6h)		Examen parcial (2h)	

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