

**Proyecto docente de la asignatura**

<b>Asignatura</b>	ESTRUCTURAS DE DATOS Y ALGORITMOS		
<b>Materia</b>	ENTORNO SOFTWARE		
<b>Módulo</b>	COMUNES A LA INFORMÁTICA		
<b>Titulación</b>	GRADO EN INGENIERÍA INFORMÁTICA		
<b>Plan</b>	545	<b>Código</b>	46913
<b>Periodo de impartición</b>	1 <sup>er</sup> . CUATRIMESTRE	<b>Tipo/Carácter</b>	Complementos de Informática
<b>Nivel/Ciclo</b>	GRADO	<b>Curso</b>	2º
<b>Créditos ECTS</b>	6 ECTS		
<b>Lengua en que se imparte</b>	CASTELLANO		
<b>Profesor/es responsable/s</b>	César Vaca Rodríguez		
<b>Datos de contacto (E-mail, teléfono...)</b>	TELÉFONO: 983 423000 ext. 5620 E-MAIL: <a href="mailto:cvaca@infor.uva.es">cvaca@infor.uva.es</a>		
<b>Horario de tutorías</b>	Véase <a href="http://www.inf.uva.es">www.inf.uva.es</a> >> Alumno >> Apoyo >> Tutorías		
<b>Departamento</b>	Informática (ATC, CCIA y LSI)		



## 1. Situación / Sentido de la Asignatura

### 1.1 Contextualización

El objetivo de la asignatura es estudiar las distintas formas de almacenar y organizar datos de forma que puedan ser usados eficientemente. Para medir la eficiencia se introducen las técnicas básicas de análisis de eficiencia de algoritmos, y el estudio se organiza en base al concepto de tipo abstracto de datos de manera que sea posible clasificar las distintas alternativas separando la interfaz de su implementación.

### 1.2 Relación con otras materias

La asignatura extiende y da uso a gran parte de los conceptos contemplados en **Fundamentos de Programación**. Se utilizan técnicas estudiadas en **Paradigmas de Programación**, y su primer bloque formativo se amplía en la asignatura **Análisis y Diseño de Algoritmos**.

### 1.3 Prerrequisitos

Aunque no se han establecido prerrequisitos, es fundamental que el alumno posea conocimientos básicos de programación, en particular haber cursado y conseguido las habilidades y destrezas establecidas en la guía docente de la asignatura de **Fundamentos de Programación**. También es recomendable haber cursado la asignatura **Paradigmas de Programación**, ya que se utilizarán varias de las técnicas contempladas en esa asignatura (en concreto Orientación a Objetos, Genericidad y Tipado Algebraico). Por último, es recomendable disponer de un nivel suficiente de inglés que permita al estudiante leer parte de la bibliografía de consulta.

## 2. Competencias

### 2.1 Generales

Código	Descripción
CG1	Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto, de acuerdo con los conocimientos adquiridos según lo establecido en las competencias de formación especificadas a continuación en esta sección de la memoria, la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas
CG2	Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática de acuerdo con los conocimientos adquiridos según lo establecido en las competencias de formación especificadas a continuación en esta sección de la memoria
CG3	Capacidad para diseñar, desarrollar, evaluar y asegurar la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan.
CG5	Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería de software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en las competencias de formación especificadas a continuación en esta sección de la memoria.
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 las competencias de formación especificadas a continuación en esta sección de la memoria.
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 las competencias de formación especificadas a continuación en esta sección de la memoria.

### 2.2 Específicas

Código	Descripción
CI6	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.
CI7	Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema
CI12	Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.

## 3. Objetivos

Código	Descripción
CI7.1	Conocer y utilizar el concepto de Tipo Abstracto de Datos (TAD), así como los TADs fundamentales.
CI6.1	Conocer y aplicar las técnicas básicas para realizar análisis de eficiencia de algoritmos.
CI7.2	Conocer y comprender las distintas implementaciones de árboles y grafos, y tener la capacidad para adaptarlas a las características de un problema particular.
CI7.3	Conocer las distintas implementaciones de tablas de dispersión y la eficiencia de cada una de ellas.
CI12.1	Conocer las diversas alternativas que permiten almacenar información en un sistema de ficheros.
CI7.4	Ser capaz de diseñar implementaciones eficientes para nuevos TADs mediante la extensión o combinación de las implementaciones estudiadas.



#### 4. Bloques temáticos

##### Bloque 1: Análisis de Algoritmos

Carga de trabajo en créditos ECTS: 1.5

##### a. Contextualización y justificación

En este bloque temático se estudiarán las técnicas básicas de análisis de eficiencia de algoritmos, las cuales se utilizarán en los bloques temáticos siguientes para comparar distintas implementaciones y poder determinar la más adecuada para un problema concreto.

##### b. Objetivos de aprendizaje

Cl6.1	Conocer y aplicar las técnicas básicas para realizar análisis de eficiencia de algoritmos.
-------	--

##### c. Contenidos

###### Tema 1: Análisis de Algoritmos

- Medida de algoritmos
- Notación Asintótica
- Relaciones de Recurrencia: Teorema Maestro
- Caso práctico: Algoritmos de búsqueda y ordenación
- Ficheros: Indices invertidos y Ordenación externa

##### d. Métodos docentes

Actividad	Metodología
Clase de teoría	<ul style="list-style-type: none"> <li>• Clase magistral participativa</li> <li>• Estudio de casos en aula</li> <li>• Resolución de problemas</li> </ul>
Clase práctica	<ul style="list-style-type: none"> <li>• Clase magistral participativa</li> <li>• Realización de un proyecto guiado por el profesor, que encargará y guiará el trabajo que se realizará en grupos (2/3 alumnos), siguiendo un enfoque colaborativo.</li> </ul>
Seminarios	<ul style="list-style-type: none"> <li>• Introducción al entorno práctico</li> </ul>

##### e. Plan de trabajo

Para este bloque se estiman 15 horas presenciales distribuidas en 5 horas teóricas, 4 horas prácticas, 2 de seminario, y 4 para efectuar las evaluaciones. El tiempo de dedicación no presencial del alumno medio es de unas 22 horas.

##### f. Evaluación

La evaluación teórica de este bloque se efectuará mediante una prueba escrita de 1 hora de duración y se realizará en la cuarta semana o en la anterior o posterior.



La evaluación práctica consistirá en la entrega y defensa de un trabajo práctico (2 horas de duración) y se realizará en la sexta semana o en la anterior o posterior.

### g. Bibliografía básica

---

- [Weiss] M.A. Weiss. "Estructuras de datos en Java". Addison-Wesley Iberoamericana, 2004
- [CLRS] Cormen, Leiserson, Rivest, Stein. "Introduction to Algorithms". MIT Press, 2001
- [NyHoff] L. NyHoff. "TADs, estructuras de datos y resolución de problemas con C++". Pearson Prentice Hall, 2005
- [Aho] Aho, Hopcroft, Ullman. "Estructuras de datos y algoritmos". Addison-Wesley Longman, 1998.

### h. Bibliografía complementaria

---

- [Shaffer] C.A. Shaffer. "A Practical Introduction to Data Structures and Algorithm Analysis". Java Edition. Prentice Hall, 1998.
- [Goodrich] M.T. Goodrich, R. Tamassia. "Data Structures and Algorithms in Java". Wiley, 1998.
- [McMillan] M. McMillan. "Data Structures and Algorithms using C#". Cambridge University Press, 2006.
- [Horowitz] E. Horowitz, S. Sahni. Fundamentals of Computer Algorithms. Computer Science Press.

### i. Recursos necesarios

---

Libros de texto, presentaciones audiovisuales, resolución de problemas.

### j. Temporalización

---

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
1.5	3 semanas



## Bloque 2: Tipos Abstractos de Datos (Interfaz)

Carga de trabajo en créditos ECTS:

### a. Contextualización y justificación

En éste bloque temático se estudiará el concepto de Tipo Abstracto de Datos (TAD) y los TADs fundamentales junto con sus características.

### b. Objetivos de aprendizaje

CI7.1	Conocer y utilizar el concepto de Tipo Abstracto de Datos (TAD), así como los TADs fundamentales.
-------	---

### c. Contenidos

#### Tema 2: Tipos Abstractos de Datos

- Definiciones
- El TAD String
- Concepto de Contenedor
- Colecciones e Iteradores
- Tipos de relaciones entre elementos
- TADs Contenedores Fundamentales

### d. Métodos docentes

Actividad	Metodología
Clase de teoría	<ul style="list-style-type: none"> <li>• Clase magistral participativa</li> <li>• Estudio de casos en aula</li> <li>• Resolución de problemas</li> </ul>

### e. Plan de trabajo

Para este bloque se estiman 7 horas presenciales, distribuidas en 5 horas teóricas y 2 para efectuar la evaluación. El tiempo de dedicación no presencial del alumno medio es de unas 11 horas.

### f. Evaluación

La evaluación teórica de este bloque se efectuará mediante una prueba escrita de 2 horas de duración y se realizará en la octava semana o en la anterior o posterior.



### g. Bibliografía básica

---

- [Weiss] M.A. Weiss. "Estructuras de datos en Java". Addison-Wesley Iberoamericana, 2004
- [CLRS] Cormen, Leiserson, Rivest, Stein. "Introduction to Algorithms". MIT Press, 2001
- [NyHoff] L. NyHoff. "TADs, estructuras de datos y resolución de problemas con C++". Pearson Prentice Hall, 2005
- [Aho] Aho, Hopcroft, Ullman. "Estructuras de datos y algoritmos". Addison-Wesley Longman, 1998.

### h. Bibliografía complementaria

---

- [Shaffer] C.A. Shaffer. "A Practical Introduction to Data Structures and Algorithm Analysis". Java Edition. Prentice Hall, 1998.
- [Goodrich] M.T. Goodrich, R. Tamassia. "Data Structures and Algorithms in Java". Willey, 1998.
- [McMillan] M. McMillan. "Data Structures and Algorithms using C#". Cambridge University Press, 2006.
- [Horowitz] E. Horowitz, S. Sahni. Fundamentals of Computer Algorithms. Computer Science Press.

### i. Recursos necesarios

---

Libros de texto, presentaciones audiovisuales, resolución de problemas.

### j. Temporalización

---

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
0.7	2 semanas



### Bloque 3: Estructuras de Datos (Implementación)

Carga de trabajo en créditos ECTS: 3.8

#### a. Contextualización y justificación

En este bloque temático, compuesto por 6 temas, se estudian un conjunto de posibles representaciones de los datos e implementación de las operaciones básicas.

#### b. Objetivos de aprendizaje

IC7.2	Conocer y comprender las distintas implementaciones de árboles y grafos, y tener la capacidad para adaptarlas a las características de un problema particular.
IC7.3	Conocer las distintas implementaciones de tablas de dispersión y la eficiencia de cada una de ellas.
IC12.1	Conocer las diversas alternativas que permiten almacenar información en un sistema de ficheros.
IC7.4	Ser capaz de diseñar implementaciones eficientes para nuevos TADs mediante la extensión o combinación de las implementaciones estudiadas.

#### c. Contenidos

##### Tema 3.- Vectores y listas enlazadas.

- Representaciones contiguas y enlazadas.
- Representaciones lineales y circulares.
- Uso y eficiencia para distintos TADs.

##### Tema 4.- Árboles.

- Definiciones. Propiedades.
- Implementaciones del TAD Directorio.
- Árboles binarios: Definiciones y propiedades.
- Montículos.
- Árboles binarios de búsqueda.
- Árboles AVL.
- Árboles B+

##### Tema 5.- Tablas de dispersión.

- Definiciones y objetivos.
- Dispersión abierta (encadenamiento)
- Dispersión cerrada (exploración)
- Análisis de eficiencia.

##### Tema 6.- Estructuras avanzadas (opcional)

- Extensión y combinación de implementaciones
- Caso de prueba: El TAD Partición Disjunta (Union-Find)
- Tries, Filtros de Bloom, Árboles vEB

#### d. Métodos docentes



Actividad	Metodología
<b>Clase de teoría</b>	<ul style="list-style-type: none"> <li>• Clase magistral participativa</li> <li>• Estudio de casos en aula</li> <li>• Resolución de problemas</li> </ul>
<b>Clase práctica</b>	<ul style="list-style-type: none"> <li>• Clase magistral participativa</li> <li>• Realización de un proyecto guiado por el profesor, que encargará y guiará el trabajo que se realizará en grupos (2/3 alumnos), siguiendo un enfoque colaborativo.</li> </ul>
<b>Seminarios</b>	<ul style="list-style-type: none"> <li>• Estudio e implementación de casos prácticos</li> </ul>

#### e. Plan de trabajo

Para este bloque se estiman 38 horas presenciales distribuidas en 20 horas teóricas, 12 horas prácticas, 4 de seminario, y 2 para efectuar las evaluaciones. El tiempo de dedicación no presencial del alumno medio es de unas 57 horas.

#### f. Evaluación

La evaluación teórica de éste bloque se efectuará mediante una prueba escrita de 2 horas de duración y se realizará en el examen final de la asignatura.

La evaluación práctica consistirá en la entrega y defensa de un trabajo práctico (2 horas de duración) y se realizará en la última semana o en la penúltima.

#### g. Bibliografía básica

[Weiss] M.A. Weiss. "Estructuras de datos en Java". Addison-Wesley Iberoamericana, 2004

[CLRS] Cormen, Leiserson, Rivest, Stein. "Introduction to Algorithms". MIT Press, 2001

[NyHoff] L. NyHoff. "TADs, estructuras de datos y resolución de problemas con C++". Pearson Prentice Hall, 2005

[Aho] Aho, Hopcroft, Ullman. "Estructuras de datos y algoritmos". Addison-Wesley Longman, 1998.

#### h. Bibliografía complementaria

[Shaffer] C.A. Shaffer. "A Practical Introduction to Data Structures and Algorithm Analysis". Java Edition. Prentice Hall, 1998.

[Goodrich] M.T. Goodrich, R. Tamassia. "Data Structures and Algorithms in Java". Willey, 1998.

[McMillan] M. McMillan. "Data Structures and Algorithms using C#". Cambridge University Press, 2006.

[Horowitz] E. Horowitz, S. Sahni. Fundamentals of Computer Algorithms. Computer Science Press.

#### i. Recursos necesarios

Libros de texto, presentaciones audiovisuales, resolución de problemas.

#### j. Temporalización

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
3.8	10 semanas

**5. Métodos docentes y principios metodológicos**

Actividad	Metodología
Clase de teoría	<ul style="list-style-type: none"><li>• Clase magistral participativa</li><li>• Estudio de casos en aula</li><li>• Resolución de problemas</li></ul>
Clase práctica	<ul style="list-style-type: none"><li>• Clase magistral participativa</li><li>• Realización de un proyecto guiado por el profesor, que encargará y guiará el trabajo que se realizará en grupos (2/3 alumnos), siguiendo un enfoque colaborativo.</li></ul>
Seminarios	<ul style="list-style-type: none"><li>• Estudio e implementación de casos prácticos</li></ul>

**6. Tabla de dedicación del estudiante a la asignatura**

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	28	Estudio y trabajo autónomo individual	70
Clases prácticas de aula (A)		Estudio y trabajo autónomo grupal	20
Laboratorios (L)	22		
Prácticas externas, clínicas o de campo			
Seminarios (S)	6		
Tutorías grupales (TG)			
Evaluación (fuera del periodo oficial de exámenes)	4		
<b>Total presencial</b>	<b>60</b>	<b>Total no presencial</b>	<b>90</b>

## 7. Sistema y características de la evaluación

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Examen teórico bloque 1 ( $T1 \in [0,1.5]$ )	15 %	Aproximadamente 4ª Semana. Recuperable en los exámenes ordinario y extraordinario.
Entrega primera práctica ( $P1 \in [0,1.5]$ )	15 %	Aproximadamente 6ª Semana. No recuperable
Examen teórico bloque 2 ( $T2 \in [0,1.5]$ )	15 %	Aproximadamente 8ª Semana. Recuperable en los exámenes ordinario y extraordinario.
Entrega segunda práctica ( $P2 \in [0,1.5]$ )	15 %	Aproximadamente 15ª Semana. No recuperable
Examen final ( $T3 \in [0,4]$ ó $T3+T1$ ó $T3+T2$ ó $T3+T2+T1$ )	40-70 %	No se establece nota mínima en las pruebas anteriores para presentarse al examen final. En el examen extraordinario se incorporarán apartados para poder sustituir las calificaciones de los exámenes teóricos parciales.

### Procedimientos y Sistemas de Evaluación

Los exámenes teóricos (incluido el final) consistirán en preguntas tipo test, preguntas de respuesta corta y la resolución de problemas sencillos. En los exámenes finales se incluirán apartados opcionales para la reevaluación de la calificación obtenida en los exámenes teóricos del primer y segundo bloque (y del tercero en el examen final extraordinario).

La parte práctica consistirá en la resolución de problemas mediante la obtención del código adecuado. Además de la corrección del código, se valorará la utilización de las técnicas contempladas en la asignatura.

### CRITERIOS DE CALIFICACIÓN

- **Convocatoria ordinaria:**
  - La nota es la suma  $T1+P1+T2+P2+T3$  y se considera aprobado si supera o iguala el valor 5.0
  - En el examen ordinario se incorporarán apartados para poder sustituir las calificaciones de los exámenes teóricos parciales.
- **Convocatoria extraordinaria:**
  - Igual que en la convocatoria ordinaria, salvo que si  $T1+P1+T2+P2+T3 < 5.0$  pero se cumple que  $T1+T2+T3 > 3.5$  entonces la nota será 5.0 (aprobado)

## 8. Consideraciones finales

Las semanas indicadas para las evaluaciones son orientativas, pudiendo sufrir cambios (semana anterior o siguiente) por mejor acomodación del calendario.

En la página web principal de la asignatura se indicará el cronograma de actividades de la asignatura con sus fechas definitivas.

Página web de la asignatura: <http://www.infor.uva.es/~cvaca/asigs/eda.html>