

**Proyecto/Guía docente de la asignatura**

Asignatura	Programación y Estructuras de datos		
Materia	Proceso de desarrollo del SW		
Módulo	Fundamentos de las tecnologías de la Información		
Titulación	1. Grado en Ingeniería Informática de Servicios y Aplicaciones (Grado IISA – Plan 413) ** Programa de estudios conjunto de Grado IISA y Grado en Matemáticas => <u>Doble Grado INFOMAT</u> : <ul style="list-style-type: none">• Doble Grado INFOMAT (Itinerario desde Valladolid – Plan 5471)2. Doble Grado INFOMAT (Itinerario desde Segovia – Plan 5472)		
Plan	413 / 5471 / 5472	Código	40816
Periodo de impartición	Semestre 4	Tipo/Carácter	Obligatoria (O)
Nivel/Ciclo	Grado	Curso	2
Créditos ECTS	6		
Lengua en que se imparte	Español		
Profesor/es responsable/s	Pilar Grande González		
Datos de contacto (E-mail, teléfono...)	Escuela de Ingeniería Informática Campus "María Zambrano" Plaza de la Universidad, 1 40.005 – Segovia (España) Teléfono: (+34) 921 11 24 51 Fax: (+34) 921 11 24 01 e-mail: pgrande@infor.uva.es		
Departamento	Informática (Área de Lenguajes y Sistemas Informáticos (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. El estudio se organiza en base al concepto de “Tipo Abstracto de Datos” (TAD) de manera que sea posible clasificar las distintas alternativas separando la interfaz de su implementación. Para medir la eficiencia de los algoritmos se hará uso de las técnicas básicas de análisis de eficiencia de algoritmos que estudiamos en la asignatura “*Metodología de la programación*” [MP].

1.2 Relación con otras materias

Esta asignatura se relaciona directamente con otras asignaturas de la materia “Proceso de desarrollo del software” como son “Fundamentos de Programación” (FP), “Metodología de la Programación” (MP) y “Programación Orientada a Objetos” (POO).

1.3 Prerrequisitos

No es prerrequisito de ninguna otra asignatura aunque los conceptos impartidos en ella constituyen una parte fundamental del núcleo principal de la formación del alumno como Ingeniero Informático. Para afrontar de forma adecuada el estudio de esta asignatura es recomendable haber superado con éxito la asignatura *Programación Orientada a Objetos* [POO]. Así mismo, resulta fundamental haber cursado con éxito las asignaturas *FP* y *MP* para comprender conceptos más complejos que serán presentados en la asignatura *Programación y Estructuras de Datos* [PED].

2. Competencias

2.1 Generales

- G01. Conocimientos generales básicos
- G02. Conocimientos básicos de la profesión
- G03. Capacidad de análisis y síntesis
- G04. Capacidad de organizar y planificar
- G05. Comunicación oral y escrita en la lengua propia
- G07. Habilidades básicas en el manejo del ordenador
- G08. Habilidades de gestión de la información
- G09. Resolución de problemas
- G10. Toma de decisiones
- G11. Capacidad crítica y autocrítica
- G12. Trabajo en equipo
- G16. Capacidad de aplicar los conocimientos en la práctica
- G18. Capacidad de aprender
- G20. Capacidad de generar nuevas ideas
- G21. Habilidad para trabajar de forma autónoma
- G22. Diseño y gestión de proyectos



2.2 Específicas

1. **E.3.** Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
2. **E.7.** 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.
3. **E.10.** Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.
4. **E.11.** 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.
5. **E.12.** Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.
6. **E.13.** Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
7. **E.25.** Capacidad para comprender el entorno de una organización y sus necesidades en el ámbito de las tecnologías de la información y las comunicaciones.
8. **E.27.** Capacidad para emplear metodologías centradas en el usuario y la organización para el desarrollo, evaluación y gestión de aplicaciones y sistemas basados en tecnologías de la información que aseguren la accesibilidad, ergonomía y usabilidad de los sistemas.
9. **E.31.** Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos.
10. **E.34.** Capacidad para participar activamente en la especificación, diseño, implementación y mantenimiento de los sistemas de información y comunicación.

3. Objetivos

- Realizar el análisis de algoritmos.
- Utilizar correctamente las distintas técnicas de diseño de algoritmos.
- Dominar la metodología de diseño de tipos abstractos de datos (TAD): abstracción, especificación e implementación.
- Conocer los TADs fundamentales.
- Comprender las distintas implementaciones, incluyendo los algoritmos más relevantes para cada uno de los TADs estudiados.
- Analizar las ventajas e inconvenientes de las diferentes implementaciones de TAD's presentadas.
- Encapsular en módulos de programación los tipos abstractos de datos y utilizar los mismos en base a su especificación y no a su implementación.
- Utilizar un entorno de programación (IDE).

4. Contenidos y/o bloques temáticos

Bloque 1: Unidad I: Algoritmia. Diseño de algoritmos

Tema 1.- Técnicas de diseño de algoritmos

Carga de trabajo en créditos ECTS: 0'5

a. Contextualización y justificación

Ampliamos conocimientos básicos de programación adquiridos en las asignaturas “Fundamentos de programación” (FP) y “Metodología de la programación” (MP), presentando conceptos clave asociados a la algoritmia y estudiando diferentes técnicas de diseño de algoritmos.



b. Objetivos de aprendizaje

1. Realizar el análisis de algoritmos.
2. Utilizar correctamente las distintas técnicas de diseño de algoritmos.
3. Utilizar un entorno de programación (IDE).

c. Contenidos

Estudio de los conceptos básicos asociados al análisis y diseño de algoritmos. Recordatorio de los principales algoritmos básicos de búsqueda y ordenación en vectores.

d. Métodos docentes

- 1.- Lección magistral: exposición de teoría
- 2.- Prácticas en aula: resolución de problemas
- 3.- Prácticas en el laboratorio: resolución de prácticas de laboratorio utilizando un IDE
- 4.- Evaluación
- 5.- Estudio autónomo por parte del alumno, incluyendo la realización de problemas, consulta bibliográfica, realización de prácticas y preparación de pruebas de evaluación .

e. Plan de trabajo

- Alternar sesiones teóricas con clases de problemas.

f. Evaluación

Ver tabla 7.

g. Bibliografía básica

- Joyanes Aguilar, L., Zahonero Martínez, I. *“Estructuras de Datos en Java”*. McGraw-Hill, 2008
- Drozdek, A. *“Estructura de Datos y Algoritmos en Java”*. Thomson, 2007
- Hernández Figueroa, Z.J. y otros. *“Fundamentos de Estructuras de Datos. Soluciones en Ada, Java y C++”*. Thomson, 2005.
- Lewis, J. Chase, J. *“Estructuras de datos con Java. Diseño de estructuras y algoritmos”*. Addison-Wesley. 2006.
- Allen Weiss, M. *“Estructuras de datos en Java”*. Addison-Wesley Iberoamericana, 2004.



h. Bibliografía complementaria

- Brassard, G. Bratley, P. “*Fundamentos de Algoritmia*”. Prentice Hall, 2000
- Cormen, T.H. y otros. “*Introduction to algorithms*” – 2ª ed. MIT Press : McGraw-Hill, 2001
- Cohoon, J. Davidson. J.W. “*Programación en Java 5.0*”. McGraw-Hill, 2006
- Sartaj Sahni. “*Data structures, algorithms, and applications in Java*”. McGraw-Hill, 2000.

i. Recursos necesarios

- Aula con pizarra y ordenador con proyector, biblioteca, sala de estudio y despacho para tutorías. Equipo de videoconferencia para impartir las clases a los alumnos del doble Grado INFOMAT.

j. Temporalización

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
0'5	

Bloque 2: Unidad II: Tipos Abstractos de Datos (TAD's)

Tema 2.- Abstracción de datos.

Tema 3.- Estructuras de datos lineales: Listas encadenadas. Pilas y Colas.

Tema 4.- Estructuras de datos jerárquicas: Árboles.

Tema 5.- Tablas de dispersión.

Tema 6.- Estructuras de datos en red: Grafos.

Carga de trabajo en créditos ECTS:

a. Contextualización y justificación

En esta unidad abordamos el estudio de la técnica de abstracción de datos que ayuda al desarrollo de programas más cortos, más legibles y flexibles.

b. Objetivos de aprendizaje

1. Comprender y aplicar correctamente la metodología de diseño de tipos abstractos de datos (TAD): abstracción, especificación e implementación.
2. Conocer los TADs fundamentales.
3. Comprender las distintas implementaciones, incluyendo los algoritmos más relevantes para cada uno de los TADs estudiados.
4. Analizar las ventajas e inconvenientes de las diferentes implementaciones de TAD's presentadas.



5. Encapsular en módulos de programación los tipos abstractos de datos y utilizar los mismos en base a su especificación y no a su implementación.

c. Contenidos

Estudio detallado de la técnica de abstracción de datos, así como, de los principales tipos abstractos de datos que están presentes en diversos paradigmas de la programación de computadores.

d. Métodos docentes

- 1.- Lección magistral: exposición de teoría
- 2.- Prácticas en aula: resolución de problemas
- 3.- Prácticas en el laboratorio: resolución de prácticas de laboratorio utilizando un IDE
- 4.- Evaluación
- 5.- Estudio autónomo por parte del alumno, incluyendo la realización de problemas, consulta bibliográfica, realización de prácticas y preparación de pruebas de evaluación.

e. Plan de trabajo

- Alternar sesiones teóricas con clases de problemas y prácticas en el laboratorio.

f. Evaluación

- Estos conceptos formarán parte de la prueba escrita que se realizará en los exámenes teórico-prácticos finales ordinario y extraordinario de la asignatura. Ver tabla 7.

g. Bibliografía básica

- Joyanes Aguilar, L., Zahonero Martínez, I. “*Estructuras de Datos en Java*”. McGraw-Hill, 2008
- Drozdek, A. “*Estructura de Datos y Algoritmos en Java*”. Thomson, 2007
- Hernández Figueroa, Z.J. y otros. “*Fundamentos de Estructuras de Datos. Soluciones en Ada, Java y C++*”. Thomson, 2005.
- Lewis, J. Chase, J. “*Estructuras de datos con Java. Diseño de estructuras y algoritmos*”. Addison-Wesley. 2006.
- Allen Weiss, M. “*Estructuras de datos en Java*”. Addison-Wesley Iberoamericana, 2004.

h. Bibliografía complementaria

- Brassard, G. Bratley, P. “*Fundamentos de Algoritmia*”. Prentice Hall, 2000
- Cormen, T.H. y otros. “*Introduction to algorithms*” – 2ª ed. MIT Press : McGraw-Hill, 2001
- Cohoon, J. Davidson. J.W. “*Programación en Java 5.0*”. McGraw-Hill, 2006
- Sartaj Sahni. “*Data structures, algorithms, and applications in Java*”. McGraw-Hill, 2000.



i. Recursos necesarios

Aula con pizarra y ordenador con proyector, biblioteca, sala de estudio y despacho para tutorías. Equipo de videoconferencia para impartir las clases a los alumnos del doble Grado INFOMAT.

j. Temporalización

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
5'5	

5. Métodos docentes y principios metodológicos

- 1.- Lección magistral: exposición de teoría
- 2.- Prácticas en aula: resolución de problemas
- 3.- Prácticas en el laboratorio: resolución de prácticas de laboratorio utilizando un IDE
- 4.- Evaluación
- 5.- Estudio autónomo por parte del alumno, incluyendo la realización de problemas, consulta bibliográfica, realización de prácticas y preparación de pruebas de evaluación

6. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	30	Estudio y trabajo autónomo individual	60
Clases prácticas de aula (A)	--	Estudio y trabajo autónomo grupal	30
Laboratorios (L)	30		
Prácticas externas, clínicas o de campo	--		
Seminarios (S)	---		
Tutorías grupales (TG)	---		
Evaluación	---		
Total presencial	60	Total no presencial	90



7. Sistema y características de la evaluación

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Realización y defensa de varias prácticas de ordenador.	20%	Estas prácticas deberán ser defendidas ante el profesor. La nota final obtenida en esta parte debe ser ≥ 5 (sobre un total de 10 puntos), para que se considere aprobada la <u>parte práctica</u> de la asignatura.
Realización de exámenes escritos de carácter teórico-práctico.	80%	La nota final obtenida en esta parte debe ser ≥ 5 (sobre un total de 10 puntos), para que se considere aprobada la <u>parte teórica</u> de la asignatura.
Importante: Para aprobar la asignatura es necesario aprobar ambas partes (teórica y práctica) por separado.		

CRITERIOS DE CALIFICACIÓN

- **Convocatoria ordinaria:**
 - Ver tabla anterior
- **Convocatoria extraordinaria:**
 - Ver tabla anterior

8. Consideraciones finales

Durante el desarrollo de la asignatura y vinculadas directamente con los conceptos teóricos impartidos en el aula, se realizan varias prácticas en el laboratorio lo que supone un trabajo continuo para el alumno.

La evaluación final de la asignatura se llevará a cabo en las convocatorias finales (ordinaria y extraordinaria), y consistirá en la realización de un examen teórico-práctico sobre los 2 bloques temáticos de la asignatura (6 temas), siendo preciso además, realizar a lo largo del cuatrimestre la entrega y defensa de las prácticas de laboratorio propuestas por el profesor, en las fechas que se establezcan y que serán comunicadas con la suficiente antelación a los alumnos a través del campus virtual (manteniéndose la misma ponderación detallada con anterioridad: 80% para el examen teórico-práctico y 20% para la parte de prácticas).