

## DATOS BÁSICOS DE LA GUÍA DOCENTE:

|                                  |  |                            |                     |
|----------------------------------|--|----------------------------|---------------------|
| <b>Materia:</b>                  | ESTRUCTURA DE DATOS  |                            |                     |
| <b>Identificador:</b>            | 30054  |                            |                     |
| <b>Titulación:</b>               | GRADUADO EN INGENIERÍA INFORMÁTICA. PLAN 2008 (BOE 15/12/2008) |                            |                     |
| <b>Módulo:</b>                   | PROGRAMACION Y DESARROLLO                                      |                            |                     |
| <b>Tipo:</b>                     | OBLIGATORIA  |                            |                     |
| <b>Curso:</b>                    | 2  | <b>Periodo lectivo:</b>    | Primer Cuatrimestre |
| <b>Créditos:</b>                 | 6  | <b>Horas totales:</b>      | 150                 |
| <b>Actividades Presenciales:</b> | 62   | <b>Trabajo Autónomo:</b>   | 88                  |
| <b>Idioma Principal:</b>         | Castellano   | <b>Idioma Secundario:</b>  | Inglés              |
| <b>Profesor:</b>                 | LAPEÑA MARTI, RAUL (T)   | <b>Correo electrónico:</b> | rlapena@usj.es      |

## PRESENTACIÓN:

La asignatura *Estructuras de Datos* proporciona a los alumnos conocimientos y competencias necesarias para avanzar en su desarrollo como profesional. Para resolver un problema determinado informáticamente, no es suficiente con implementar el primer código que aparece en la mente del programador. Un buen profesional busca la solución más elegante y eficiente que resuelva correctamente la situación planteada. Para ello es necesario conocer en detalle las estructuras de datos adecuadas para cada situación, así como saber analizar el rendimiento de los algoritmos diseñados para poder elegir en cada momento la implementación idónea. En esta asignatura se estudiarán en detalle estos aspectos de la programación de tal forma que el alumno será capaz de diseñar, documentar e implementar correctamente tipos abstractos de datos. Al mismo tiempo se hará especial hincapié en la capacidad de abstracción y optimización de los algoritmos que se implementen para mejorar las competencias de programación del alumno.

## COMPETENCIAS PROFESIONALES A DESARROLLAR EN LA MATERIA:

|  |     |   |
|--|-----|---|
| <b>Competencias Generales de la titulación</b>   | G02 | Capacidad innovadora para proponer y encontrar formas nuevas y eficaces de realizar cualquier tarea y/ o función dentro de su entorno profesional con una elevada motivación por la calidad |
|  | G05 | Capacidad para adaptarse a diferentes entornos con una actitud positiva y optimista y orientar su conducta a la consecución de metas  |
|  | G06 | Capacidad para analizar y resolver los problemas o imprevistos complejos que puedan surgir durante la actividad profesional dentro de cualquier tipo de organización socio-económica        |
|  | G08 | Habilidad para comunicarse eficazmente sobre distintos temas en una variedad de contextos profesionales y con los diferentes medios disponibles   |
|  | G09 | Capacidad para tomar decisiones de manera imparcial y desde un punto de vista racional  |
|  | G10 | Capacidad crítica y analítica en la evaluación de información, datos y líneas de actuación  |
|  | G13 | Capacidad para utilizar estrategias de aprendizaje individuales orientadas a la mejora continua en el ejercicio profesional y para emprender estudios posteriores de forma autónoma         |
|  | G14 | Capacidad de abstracción para manejar diferentes modelos complejos de conocimiento y aplicarlos al planteamiento y resolución de problemas  |
| <b>Competencias Específicas de la titulación</b> | E03 | Capacidad para reconocer los principios teóricos y aplicar satisfactoriamente los métodos prácticos apropiados para el análisis y la resolución de problemas de ingeniería                  |
|  | E04 | Capacidad para mantener una actitud abierta a la innovación y la creatividad en el marco de la profesión de ingeniería  |
|  | E08 | Capacidad para comunicarse productivamente con clientes, usuarios y colegas, tanto de modo oral como por escrito, con el fin de transmitir ideas, resolver conflictos y alcanzar consensos  |
|  | E09 | Capacidad para mantener las competencias profesionales mediante el aprendizaje autónomo y la mejora continua  |
|  | E11 | Capacidad para mantenerse al día en el mundo tecnológico y empresarial en el ámbito de las tecnologías de la informática y comunicaciones   |
|  | E12 | Capacidad para gestionar la complejidad a través de la abstracción, el modelado, las "best practices", los patrones, los estándares y el uso de herramientas apropiadas                     |
|  | E13 | Capacidad para identificar, evaluar y usar tecnologías actuales y emergentes, considerando su aplicabilidad en función de las necesidades de individuos y organizaciones                    |
|  | E16 | Capacidad para comprender un dominio de aplicación hasta el punto de ser capaz de desarrollar aplicaciones IT adecuadas para el mismo   |
|  | E17 | Capacidad para identificar y analizar las necesidades de los usuarios con el objetivo de  |

|                                  |     |  |
|----------------------------------|-----|--|
|                                  |     | diseñar soluciones IT efectivas y usables que puedan integrarse en el entorno operativo del usuario.   |
|                                  | E18 | Capacidad para identificar y definir los requisitos que deben ser satisfechos por los sistemas IT para satisfacer las necesidades planteadas por organizaciones o individuos                       |
|                                  | E20 | Capacidad para realizar el diseño detallado de los componentes del proyecto (procedimientos, interfaz de usuario, características de equipos, parámetros de los sistemas de comunicaciones, etc.). |
|                                  | E27 | Capacidad para elaborar y mantener documentación descriptiva de la génesis, producción y operatividad de los sistemas informáticos   |
| <b>Resultados de Aprendizaje</b> | R01 | Crear algoritmos que cumplan unas necesidades establecidas de una manera óptima y cumpliendo unos estándares de calidad específicos  |
|                                  | R02 | Implementar de manera eficiente algoritmos   |
|                                  | R03 | Comentar código fuente con calidad   |
|                                  | R04 | Comprobar errores de implementación  |
|                                  | R05 | Manejar entornos de desarrollo de programación   |
|                                  | R06 | Crear y manejar las estructuras de datos   |
|                                  | R07 | Optimizar y evaluar algoritmos   |

### REQUISITOS PREVIOS:

Haber cursado la asignatura *Fundamentos de Programación* o equivalente.

### PROGRAMACIÓN DE LA MATERIA:

#### Contenidos de la materia:

|   |
|---|
| <b>1 - Estructuras elementales de datos</b>     |
| 1.1 - Vectores                                  |
| 1.2 - Estructuras                               |
| 1.3 - Punteros                                  |
| 1.4 - Cadenas                                   |
| <b>2 - Tipos abstractos de datos</b>            |
| 2.1 - Abstracción                               |
| 2.2 - Modularidad                               |
| 2.3 - Abstracción en lenguajes de programación  |
| 2.4 - Tipos abstractos de datos                 |
| 2.5 - TAD                                       |
| 2.6 - Entornos de desarrollo                    |
| <b>3 - Listas</b>                               |
| 3.1 - Listas enlazadas                          |
| 3.2 - Listas doblemente enlazadas               |
| 3.3 - Listas circulares                         |
| 3.4 - Implementación de listas                  |
| 3.5 - Comprobación de errores de implementación |
| <b>4 - Pilas y colas</b>                        |
| 4.1 - Concepto de pila                          |
| 4.2 - Implementación de pilas                   |
| 4.3 - Concepto de cola                          |
| 4.4 - Implementación de colas                   |
| 4.5 - Colas dinámicas                           |
| 4.6 - Bicolos                                   |
| 4.7 - Colas con prioridades                     |
| <b>5 - Tablas de dispersión</b>                 |
| 5.1 - Concepto de tabla de dispersión           |
| 5.2 - Funciones de dispersión                   |
| 5.3 - Colisiones                                |
| <b>6 - Análisis de algoritmos</b>               |
| 6.1 - Eficiencia de algoritmos                  |
| 6.2 - Notación "O-grande"                       |

|  |
|--|
| 6.3 - Ejemplos                                 |
| <b>7 - Algoritmos de ordenación y búsqueda</b> |
| 7.1 - Algoritmos de búsqueda                   |
| 7.2 - Algoritmos de ordenación                 |
| 7.3 - Ordenación externa                       |
| <b>8 - Árboles</b>                             |
| 8.1 - Introducción y terminología              |
| 8.2 - Árboles binarios                         |
| 8.3 - Estructuras equilibradas de búsqueda     |
| <b>9 - Grafos</b>                              |
| 9.1 - Introducción                             |
| 9.2 - Representación                           |
| 9.3 - Implementación con listas                |
| 9.4 - Recorrido                                |
| 9.5 - Conexiones                               |
| 9.6 - Caminos                                  |
| <b>10 - Algoritmos con grafos</b>              |
| 10.1 - Introducción                            |
| 10.2 - Ordenación topológica                   |
| 10.3 - Caminos con un solo origen              |
| 10.4 - Caminos para todos los nodos            |
| 10.5 - Árbol de expansión de coste mínimo      |

La planificación de la asignatura podrá verse modificada por motivos imprevistos (rendimiento del grupo, disponibilidad de recursos, modificaciones en el calendario académico, etc.) y por tanto no deberá considerarse como definitiva y cerrada.

## METODOLOGÍAS Y ACTIVIDADES DE ENSEÑANZA Y APRENDIZAJE:

### Metodologías de enseñanza-aprendizaje a desarrollar:

- Sesiones teóricas. Clases magistrales de transmisión de conocimientos por parte del profesor, con participación activa de los estudiantes.
- Sesiones de tutoría. Las sesiones de tutoría se anunciarán a través de la PDU. Se fomentará en estas sesiones el uso de tecnologías no presenciales.
- Trabajo autónomo. Los alumnos deberán estudiar el material presentado, tratar de resolver los ejercicios y problemas propuestos, que serán similares a los que formarán parte de los exámenes parciales y desarrollar los trabajos propuestos.

### Volumen de trabajo del alumno:

| Modalidad organizativa          | Métodos de enseñanza                                | Horas estimadas |
|---------------------------------|---|-----------------|
| <b>Actividades Presenciales</b> | Clase magistral                                     | 28              |
|                                 | Casos prácticos                                     | 6               |
|                                 | Resolución de prácticas, problemas, ejercicios etc. | 12              |
|                                 | Exposiciones de trabajos de los alumnos             | 2               |
|                                 | Prácticas de laboratorio                            | 12              |
|                                 | Actividades de evaluación                           | 2               |
| <b>Trabajo Autónomo</b>         | Asistencia a tutorías                               | 5               |
|                                 | Estudio individual                                  | 36              |
|                                 | Preparación de trabajos individuales                | 35              |
|                                 | Preparación de trabajos en equipo                   | 12              |
| <b>Horas totales:</b>           |   | 150             |

## SISTEMA DE EVALUACIÓN:

**Obtención de la nota final:**

|                        |            |          |
|------------------------|------------|----------|
| Pruebas escritas:      | 35         | %        |
| Trabajos individuales: | 45         | %        |
| Trabajos en equipo:    | 15         | %        |
| Participación:         | 5          | %        |
| <b>TOTAL</b>           | <b>100</b> | <b>%</b> |

\*Las observaciones específicas sobre el sistema de evaluación serán comunicadas por escrito a los alumnos al inicio de la materia.

**BIBLIOGRAFÍA Y DOCUMENTACIÓN:**

**Bibliografía básica:**

|  |
|--|
| L. Joyanes, I. Zahonero, Algoritmos y estructuras de datos una perspectiva en C, McGraw-Hill |
| G. Brassard, P. Bratley, Fundamentos de algoritmia, Prentice Hall                            |
| Weiss, Mark Allen. Estructuras de datos en Java. Addison Wesley                              |

**Bibliografía recomendada:**

**Páginas web recomendadas:**

|                                 |   |
|---------------------------------|---|
| Programación en C               | <a href="http://cprogramming.com/">http://cprogramming.com/</a>   |
| Eclipse (entorno de desarrollo) | <a href="http://www.eclipse.org/downloads/">http://www.eclipse.org/downloads/</a>   |
| Java                            | <a href="http://www.oracle.com/technetwork/es/java/javase/downloads/index.html">http://www.oracle.com/technetwork/es/java/javase/downloads/index.html</a> |