



GUÍA DOCENTE

ESTRUCTURAS DE DATOS

Coordinación: GARRIDO NAVARRO, JUAN ENRIQUE

Año académico 2020-21

Información general de la asignatura

Denominación	ESTRUCTURAS DE DATOS			
Código	102010			
Semestre de impartición	1R Q(SEMESTRE) EVALUACIÓN CONTINUADA			
Carácter	Grado/Máster	Curso	Carácter	Modalidad
	Doble titulación: Grado en Ingeniería Informática y Grado en Administración y Dirección de Empresas	2	OBLIGATORIA	Presencial
	Grado en Ingeniería Informática	2	OBLIGATORIA	Presencial
Número de créditos de la asignatura (ECTS)	6			
Tipo de actividad, créditos y grupos	Tipo de actividad	PRALAB	TEORIA	
	Número de créditos	3	3	
	Número de grupos	3	1	
Coordinación	GARRIDO NAVARRO, JUAN ENRIQUE			
Departamento/s	INFORMATICA E INGENIERIA INDUSTRIAL			
Distribución carga docente entre la clase presencial y el trabajo autónomo del estudiante	Trabajo en clase (20%) Trabajo online (20%) Trabajo autónomo por parte del estudiante (60%)			
Información importante sobre tratamiento de datos	Consulte este enlace para obtener más información.			
Idioma/es de impartición	Catalán y Castellano			
Distribución de créditos	Juan Enrique Garrido (GG): 3 ECTS Juan Manuel Gimeno (GM1, GM2 y GM3): 9 ECTS			

Profesor/a (es/as)	Dirección electrónica\profesor/a (es/as)	Créditos impartidos por el profesorado	Horario de tutoría/lugar
GARRIDO NAVARRO, JUAN ENRIQUE	juanenrique.garrido@udl.cat	3	
GIMENO ILLA, JUAN MANUEL	juanmanuel.gimeno@udl.cat	9	

Información complementaria de la asignatura

Estructura de Datos es una asignatura que se imparte en el primer semestre del segundo curso del Grado en Ingeniería Informática que se ofrece en la Escuela Politécnica Superior de la Universidad de Lleida. Los requisitos previos para cursar esta asignatura son conocimientos en programación, aspectos básicos de programación en Java, y estar familiarizado con la programación orientada a objetos.

Estructura de Datos está diseñada para que sea la continuación natural de Programación 2, porque profundiza en la disciplina de la programación y, especialmente, en el paradigma de la programación orientada a objetos. Este aspecto se tratará utilizando Java. Estructura de Datos también intenta complementar Algoritmos y Complejidad, porque describe algoritmos asociados a estructuras de datos y discute su eficiencia utilizando la notación "Big-Oh".

Estructura de Datos está diseñada de tal manera que la carga de trabajo sea lo más constante posible durante el curso, evitando picos de trabajo. Este aspecto de la asignatura, por el contrario, implicará una dedicación continua, desde el principio, de los estudiantes.

Objetivos académicos de la asignatura

- Conocer y ser capaz de trabajar con las principales estructuras de datos de acceso secuencial y en forma de árbol, y conocer los principales aspectos de las estructuras de datos de acceso directo.
- Profundizar en la programación orientada a objetos: utilizar interfaces, clases abstractas y genéricos en la programación de estructuras de datos mediante el Java Collections Framework.
- Aplicar la notación "Big-Oh" para analizar el coste de las principales operaciones de las estructuras de datos e implementar algoritmos más eficientes.
- Profundizar en la recursividad: diseñar métodos recursivos y transformarlos en iterativos para recorrer y trabajar con estructuras de datos en forma de árbol.
- Diseñar e implementar clases que combinen diferentes estructuras de datos y aspectos de programación orientada a objetos para solucionar problemas concretos.

Competencias

Competencias transversales EPS

- **EPS1.** Capacidad de resolución de problemas y elaboración y defensa de argumentos dentro de su área de estudios.
- **EPS5.** Capacidad para la abstracción y el razonamiento crítico, lógico y matemático.

Competencias específicas de la titulación

- **GII-FB3.** Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.
- **GII-CRI6.** 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.
- **GII-CRI7.** Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.
- **GII-CRI8.** Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

Contenidos fundamentales de la asignatura

1. Introducción al análisis de algoritmos
 - 1.1 Introducción al análisis de algoritmos para estructuras de datos
 - 1.2 Notación Big-Oh
 - 1.3 Caso de Estudio
 - 1.4 Consideraciones para Big-Oh
2. Aspectos de programación orientada a objetos
 - 2.1 Tipos, herencia, clases abstractas e interfaces
 - 2.2 JCF: Introducción guiada
 - 2.3 Conversiones y comprobaciones de tipos
 - 2.4 Genéricos y Comodines
 - 2.5 JCF: Introducción guiada (ampliación)
3. Estructuras de datos secuenciales
 - 3.1 Estructuras de datos
 - 3.2 Pilas
 - 3.3 Listas
 - 3.4 Colas
 - 3.5 Comparativa del coste de las operaciones principales
4. Estructuras de datos con forma arborescente
 - 4.1 Definición y notación
 - 4.2 Árboles binarios
 - 4.3 Árboles m-arios de búsqueda y árboles B
5. Estructuras de datos de acceso directo: Tablas
 - 5.1 Introducción
 - 5.2 Especificación de tablas
 - 5.3 Implementación de tablas

Ejes metodológicos de la asignatura

Grupo Grande: Clases de Teoría (3 créditos)

- Parte teórica: clases soportadas por apuntes.
- Parte de aplicación práctica: se trabaja siempre sobre ejemplos.

Grupos Medianos: Clases de Laboratorio (3 créditos)

- Realización de ejercicios y proyectos en grupos de dos.
- Resolución de dificultades y seguimiento personalizado de las prácticas.
- Uso de un entorno integrado de desarrollo (IntelliJ).

Trabajo autónomo (no presencial):

- Estudio de conceptos.
- Finalización de proyectos.

Plan de desarrollo de la asignatura

Semana	Grupo Grande	Grupo Mediano	Trabajo Autónomo
1	1. Análisis de Algoritmos	Introducción Laboratorio	Estudio y proyecto

Semana	Grupo Grande	Grupo Mediano	Trabajo Autónomo
2	1. Análisis de Algoritmos	Laboratorio 1	Estudio y proyecto
3	2. Aspectos POO	Laboratorio 1	Estudio y proyecto
4	2. Aspectos POO	Laboratorio 2	Estudio y proyecto
5	3. EEDD Secuenciales	Laboratorio 2	Estudio y proyecto
6	3. EEDD Secuenciales	Laboratorio 3	Estudio y proyecto
7	3. EEDD Secuenciales	Laboratorio 3	Estudio y proyecto
8	Preparación parcial	Laboratorio 3	Estudio
9	Primer parcial		
10	4. EEDD Arborescentes	Laboratorio 4	Estudio y proyecto
11	4. EEDD Arborescentes	Laboratorio 4	Estudio y proyecto
12	4. EEDD Arborescentes	Laboratorio 5	Estudio y proyecto
13	5. EEDD Acceso Directo	Laboratorio 5	Estudio y proyecto
14	5. EEDD Acceso Directo	Laboratorio 5	Estudio y proyecto
15	Preparación parcial	Laboratorio 5	Estudio y proyecto
16	Segundo parcial		Estudio
17	Segundo parcial		Estudio
18	Tutorías		Estudio y proyecto
19	Recuperaciones		Estudio

Sistema de evaluación

Acr	Descripción	Peso	Nota mínima	Obligatoria	Recuperable	Ind/Grup
Par1	Primer parcial Temas 1, 2, 3 Laboratorio 1 <ul style="list-style-type: none"> Fecha aproximada publicación enunciado: Semana 2 Fecha aproximada entrega: Semana 4 Laboratorio 2 <ul style="list-style-type: none"> Fecha aproximada publicación enunciado: Semana 4 Fecha aproximada entrega: Semana 6 Laboratorio 3 <ul style="list-style-type: none"> Fecha aproximada publicación enunciado: Semana 6 Fecha aproximada entrega: Semana 8 	25%	4,0	Sí	Sí	Ind
Par2	Segundo parcial Temas 4 y 5 Laboratorio 4 <ul style="list-style-type: none"> Fecha aproximada publicación enunciado: Semana 10 Fecha aproximada entrega: Semana 12 Laboratorio 5 <ul style="list-style-type: none"> Fecha aproximada publicación enunciado: Semana 12 Fecha aproximada entrega: Semana 15 	25%	4,0	Sí	Sí	Ind
Lab	(lab1 + lab2 + lab3 + lab4 + lab5) / 5	50%	5,0	Sí	Sí	2

Nota final = $0,25 * \text{Par1} + 0,25 * \text{Par2} + 0,5 * \text{Lab}$

- La nota máxima en la recuperación de un parcial será de 8.
- La nota máxima en la recuperación de una actividad de laboratorio será de 6.
- Las notas de las actividades superadas se mantendrán para calcular la nota durante este curso académico pero en ningún caso para otros.
- El profesorado también valorará la evolución del alumno durante el curso en el cómputo de la nota final.
- **Para superar el curso todas las actividades obligatorias han de estar superadas.**

Bibliografía y recursos de información

Josep Maria Ribó. Apropament a les estructures de dades del del programari lliure. Edicions de la Universitat de Lleida. 2018.

Michael Goodrich, Roberto Tamassia, M.H. Goldwasser. Data Structures and Algorithms in Java. John Wiley and Sons. 2015.

William Collins. Data Structures and the Java Collections Framework. Third edition. John Wiley & Sons, 2010. USA.

Adam Drozdek. Data Structures and Algorithms in Java. Second edition. Thomson Learning, 2010. USA.

Mark Allen Weiss. Data Structures & Problem Solving Using Java. Fourth Edition. Addison Wesley, 2010. USA.

Maurice Naftalin, Philip Wadler. Java Generics and Collections. O'reilly, 2007. USA.

John Lewis, Joseph Chase. Java Software Structures: Designing and Using Data Structures. Addison Wesley. 2005.