



Universitat de Lleida

GUÍA DOCENTE
ESTRUCTURAS DE DATOS

Coordinación: GARRIDO NAVARRO, JUAN ENRIQUE

Año académico 2016-17

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 ECTS	6			
Grupos	1GG,3GM			
Créditos teóricos	3			
Créditos prácticos	3			
Coordinación	GARRIDO NAVARRO, JUAN ENRIQUE			
Departamento/s	INFORMATICA I ENGINYERIA INDUSTRIAL			
Distribución carga docente entre la clase presencial y el trabajo autónomo del estudiante	Trabajo en clase (40%) 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, GM2): 6 ECTS Xavier Domingo (GM1): 3 ECTS Juan Manuel Gimeno (GM3): 3 ECTS			
Horario de tutoría/lugar	A concertar por correo electrónico			

Profesor/a (es/as)	Dirección electrónica profesor/a (es/as)	Créditos impartidos por el profesorado	Horario de tutoría/lugar
DOMINGO ALBIN, JAVIER JUAN	xdomingo@diei.udl.cat	3	Con cita previa
GARRIDO NAVARRO, JUAN ENRIQUE	juanenrique.garrido@diei.udl.cat	6	Con cita previa
GIMENO ILLA, JUAN MANUEL	jmgimeno@diei.udl.cat	3	Con cita previa

Información complementaria de la asignatura

Estructura de Datos es una asignatura que se imparte en el primer semestre de segundo 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 saber 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 la 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
2. Aspectos de programación orientada a objetos
 - 2.1 Tipo, interfaz y herencia
 - 2.2 Introducción a la JCF
3. Principales estructuras de datos de acceso secuencial
 - 3.1 Pilas
 - 3.2 Listas: ArrayList y LinkedList
 - 3.2 Colas circulares y de prioridad
4. Estructuras de datos arborescentes
 - 4.1 Definiciones y notación
 - 4.2 Repaso de recursividad
 - 4.3 Árboles binarios
 - 4.4 Recorridos
 - 4.5 Árboles binarios de búsqueda
 - 4.6 árboles n-arios de búsqueda
5. Introducción a otras estructuras de datos

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.

Grups 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 (Netbeans).

Trabajo autónomo (no presencial):

- Estudio de los conceptos
- Completación de los proyectos

Plan de desarrollo de la asignatura

Semana	Grupo Grande	Grupo Mediano	Trabajo Autónomo
1	1. Análisis de Algoritmos	Laboratorio 1	Estudio
2	2. Aspectos POO	Laboratorio 1	Estudio
3	2. Aspectos POO	Laboratorio 2	Estudio
4	2. Aspectos POO	Laboratorio 2	Estudio
5	3. ED Secuenciales	Mini-proyecto 1	Estudio y proyecto
6	3. ED Secuenciales	Mini-proyecto 1	Estudio y proyecto
7	3. ED Secuenciales	Mini-proyecto 2	Estudio y proyecto
8	3. ED Secuenciales	Mini-proyecto 2	Estudio y proyecto
9	Primer parcial		
10	4. ED Arborescentes	Laboratorio 3	Estudio
11	4. ED Arborescentes	Laboratorio 3	Estudio
12	4. ED Arborescentes	Mini-proyecto 3	Estudio y proyecto
13	4. ED Arborescentes	Mini-proyecto 3	Estudio y proyecto
14	4. ED Arborescentes 5. Otras ED	Mini-proyecto 4	Estudio y proyecto
15	5. Otras ED	Mini-proyecto 4	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 Laboratorios 1 i 2	25%	4,0	Sí	Sí	Ind

Acr	Descripción	Peso	Nota mínima	Obligatoria	Recuperable	Ind/Grup
Par2	Segundo parcial Temas 4 i 5 Laboratorio 3 Proyectos 1 a 4	25%	4,0	Sí	Sí	Ind
Lab	(lab1 + lab2 + lab3) /3	25%	5,0	Sí	Sí	2
Pro	(proj1 + proj2 + proj3 + proj4)/ 4	25%	5,0	Sí	Sí	2

Nota final = 0,25 * Par1 + 0,25 * par2 + 0,25 * Lab + 0,25 * Pro

- Las recuperaciones tendrán el mismo peso que las actividades de evaluación ordinarias
- La recuperación de las actividades (parciales, laboratorios, proyectos) no será en ningún caso para subir nota si la actividad ya ha sido superada.
- 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

William J. 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, 2005. USA.

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

Josep Maria Ribó. Apropament a les estructures de dades del del programari lliure. 2014.

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