ORGANIZACIÓN DOCENTE del curso 2009-10

1. DATOS GENERALES DE LA ASIGNATURA

NOMBRE	Estructuras de Datos y Algoritmos	**PÁGINA WEB**	www.ctr.unican.es/asignaturas/eda
CÓDIGO			
DEPARTAMENTO	Matemáticas, Estadística y Computación		
PLAN DE ESTUDIOS	Ingeniero en Informática	**CURSO**	2009/2010

PROFESORADO	**Nombre**	**e-mail**
	Michael González Harbour	mgh@unican.es
	Mario Aldea	aldeam@unican.es
	Profesor asociado (a contratar)	

CRÉDITOS ALUMNO	Teóricos (1)	Prac. Problemas (2)	Prac. Laboratorio	Prac. Computador	TOTALES
	4.5	1.5		3	9

LUGAR DE IMPARTICIÓN	Teóricos	Prac. Problemas	Prac. Laboratorio	Prac. Computador

HORARIO PREVISTO(*)	Teóricos	Prac. Problemas	Prac. Laboratorio	Prac. Computador

(*) Observaciones:	Consultar el cuadro que se expone en el tablón de anuncios

(1) Se corresponde con clases magistrales de teoría en aula
(2) Se corresponde con clases prácticas (problemas, experiencias de cátedra,….) en aula

2. PROGRAMA DE LA ASIGNATURA

1. Introducción.

- Estructuras de datos abstractas
- Eficiencia de las estructuras de datos
- Interfaces y herencia múltiple
- Estructuras de datos genéricas
- Colecciones
- Iteradores
- Relaciones de igualdad y orden

2. Estructuras de datos lineales

- Colecciones o bolsas
- Conjuntos
- Listas y vectores
- Pilas
- Colas
- Colas de prioridad
- Mapas
- Aplicaciones

3. Estructuras de datos jerárquicas

- Árboles
- Recorrido y ordenación de los nudos
- El ADT árbol
- Árboles binarios
- Búsquedas en árboles binarios
- Aplicaciones

4. Grafos y caminos

- Concepto de grafo
- Definiciones
- La interfaz de las aristas
- La interfaz de los grafos
- Recorridos y búsquedas
- El problema del camino mínimo con y sin pesos
- Grafos acíclicos
- Aplicaciones

5. Implementación de Listas, Colas y Pilas

- Introducción a la implementación de estructuras de datos
- Pilas, colas y vectores implementados mediante arrays
- Implementaciones con listas enlazadas simples
- Listas enlazadas con cursores
- Listas doblemente enlazadas

6. Implementación de mapas, árboles y grafos

- Mapas y conjuntos implementados mediante tablas de troceado
- Técnicas de troceado abierto
- Técnicas de troceado cerrado

- Implementaciones de árboles
- Implementaciones de árboles binarios
- Árboles binarios equilibrados y conjuntos ordenados
- Árboles AVL
- Árboles rojinegros
- B-árboles
- Implementación de colas de prioridad y conjuntos ordenados mediante montículos binarios
- Implementación de grafos

Asignaturas que se recomienda al alumno haber cursado o estar cursando

Programación I y Prácticas de Programación

3. OBJETIVOS GENERALES DE LA ASIGNATURA

Conocer los fundamentos del diseño, análisis e implementación de estructuras de datos básicas y de sus algoritmos de manipulación aplicando los principios de abstracción y descomposición orientada a objetos. Conocer las principales aplicaciones de las estructuras de datos básicas.

.

4. OBJETIVOS ESPECIFICOS: APTITUDES/DESTREZAS

Conocimientos

- Conocer las especificaciones abstractas de las principales estructuras de datos
- Conocer los principales algoritmos de manipulación de las estructuras de datos básicas
- Conocer las principales aplicaciones de las estructuras de datos básicas
- Conocer distintas técnicas de implementación de las estructuras de datos y sus propiedades

Habilidades

- Ser capaz de aplicar los principios de abstracción a las estructuras de información
- Saber diseñar la estructura de datos más eficiente para un determinado problema, dados unos requisitos de coste temporal y espacial
- Ser capaz de implementar estructuras de datos básicas usando un lenguaje orientado a objetos
- Ser capaz de razonar sobre la eficiencia de las diferentes implementaciones de una estructura de datos

Capacidades

- Dotar de capacidad para la abstracción (simplificación) y para el razonamiento semiformal pero riguroso.
- Capacidad para comprender los problemas, desde el enunciado, saber especificar y distinguir datos, resultados y datos/resultados.
- Capacidad para enfrentarse a los problemas, siendo capaz de resolver nuevas situaciones desde los conocimientos adquiridos.
- Dotar de capacidad para abordar con éxito el estudio de nuevas materias de la titulación de Ingeniería Informática.

5. BIBLIOGRAFÍA

Básica

- Mark A. Weiss. "Estructuras de datos en Java" Addison Wesley, 2000

Complementaria

- S. Zakhour, S. Hommel, J. Royal, I. Rabinovitch, T. Risser, M. Hoeber, "The Java Tutorial Fourth Edition". Pearson Education, 2006
- The Java Tutorials. http://java.sun.com/docs/books/tutorial/
- Aho, A.V., J.E. Hopcroft, J.D. Ullman, Estructuras de datos y algoritmos, Addison-Wesley, 1988.
- Arnold, K., J. Gosling, D. Holmes, El Lenguaje de Programación Java, 3ª Ed., Addison-Wesley, 2001.
- David A. Watt, Deryck F. Brown, "Java Collections". Wiley, 2001
- Frank M. Carrano and Janet J. Prichard, "Data Abstraction and Problem Solvig with Java", Addison Wesley, 2001
- Michell Waite, "Data Structures & Algorithms in Java", Waite Group Press, 1998

6. ACTIVIDADES A DESARROLLAR EN LA ASIGNATURA

Se desarrollarán una serie de prácticas de laboratorio en relación con los contenidos de la asignatura.

Cada práctica se desarrollará en dos fases:

- La primera consiste en la presentación por parte del profesor a través de material escrito y explicaciones generales de la traslación al lenguaje de programación de los conceptos estudiados en teoría.

- La segunda fase consiste en la resolución de uno o varios ejercicios por parte del alumno para los que dispone de un tiempo a determinar en cada caso. La resolución de cada ejercicio implica las fases habituales en la construcción de programas: especificación del problema, diseño en pseudocódigo, codificación en el lenguaje de programación, y prueba.

-Adicionalmente se suministrará una colección de ejercicios de la asignatura relacionados con las fases de especificación y diseño en pseudocódigo. Algunos de ellos serán resueltos por el profesor para ilustrar la aplicación de los contenidos suministrados en las clases magistrales, otros seleccionados deberán ser resueltos por los alumnos.

7. MÉTODO DE EVALUACIÓN

PROCEDIMIENTO ORDINARIO: evaluación mixta, con un 40% de peso correspondiente al trabajo continuo en las prácticas de la asignatura y un 60% al examen final de cuestiones y problemas.

PROCEDIMIENTO EXTRAORDINARIO: previsto para la evaluación de Septiembre; se podrá admitir en la convocatoria de Junio en casos excepcionales y justificados (estudiantes a tiempo parcial, enfermedad, etc.) una evaluación única de las prácticas en el laboratorio, mediante un examen de prácticas. A este examen se podrá optar tras la entrega de los informes de las prácticas y ejercicios obligatorios de la asignatura. El examen final de cuestiones y problemas es el mismo que el del procedimiento ordinario. Los pesos de cada examen también (40% examen de prácticas, y 60% examen de cuestiones y problemas).

En ambos procedimientos es preciso superar por separado las prácticas y el examen de cuestiones y problemas.

Descripción de la evaluación continua: actividades que debe desarrollar el alumno y su valoración

La evaluación continua se realizará a través de la valoración de las prácticas y de la colección de ejercicios que deben ser resueltos por el alumno.

Criterios de evaluación de las prácticas:

a) Trabajo en el laboratorio
• Conocimientos: Conocimiento de los conceptos de algoritmos y del lenguaje Java
• Grado de Resolución: Grado de resolución de la práctica alcanzado durante las horas en el laboratorio.

b) Informes sobre las prácticas

• Especificación y diseño: Facilidad para especificar diseñar y comprender algoritmos.
• Estilo: Estilo de programación, claridad del código, documentación del código.
• Organización: Organización del informe, claridad en la exposición
• Fecha de entrega: Entrega del informe dentro del plazo marcado, con el objetivo de conseguir una evaluación realmente continuada.

El profesor de prácticas puede pedir que se muestre el correcto funcionamiento de la práctica en el laboratorio. Las prácticas cuyo informe se presente con retraso de hasta una semana tendrán un punto menos (sobre 10) y las de informes presentados con retraso entre una y dos semanas tendrán tres puntos menos. Las entregas pasadas dos semanas del plazo se calificarán con cero, pero su presentación es obligatoria.

Descripción del examen final (duración, se pueden llevar apuntes o no, tiene partes diferenciadas o no, se promedian teoría y problemas o no, etc).

El examen final escrito de cuestiones y problemas tiene como objetivo evaluar los conocimientos teóricos y prácticos para que lo que se plantearán una serie de preguntas y ejercicios de mediana complejidad. La duración será de 180 minutos, más un descanso en medio. Se podrán utilizar apuntes y bibliografía.

El examen de prácticas del procedimiento extraordinario es un examen en el laboratorio con el objetivo de evaluar la capacidad del alumno de llevar a la práctica sus conocimientos. Tendrá una duración de tres horas en las que se pedirá el diseño de una estructura de datos especificada por el profesor, de una aplicación sencilla que la utilice, y su codificación en lenguaje de programación, utilizando los equipos del laboratorio.

8. OBSERVACIONES

Para superar la asignatura es preciso superar tanto el examen como las prácticas. La nota final estará compuesta por:

- 60% examen escrito de cuestiones y problemas
- 40% nota de prácticas

En caso de que una de estas partes no se supere, la nota final será el mínimo de 4.5 y la media obtenida.

En caso de aprobar únicamente una de las dos partes en la convocatoria de junio, se guardaría la nota de esa parte para septiembre.

Software

Entorno de desarrollo Bluej, con compilador Java de Sun, versión 6 o superior, y sistema operativo Linux. Editor de texto emacs.