

# Estructuras de Datos

## Programa 2007

Andrea Rodríguez  
Esteban Osses

### 1. Identificación

Nombre: Estructuras de Datos

Código: 503404

Nivel Semestre: 5

Duración: Semestral

Horas Teóricas: 3

Horas Prácticas: 2

Créditos: 4

Pre-requisitos: Lenguaje de Programación

Horarios: Lu 10:00-11:00, Mi 9:00 - 11:00 [IS 2-2] (T), Lu, Ma 16:00-18:00 [Lab](P)

### 2. Descripción

Esta asignatura obligatoria presenta a los alumnos las estructuras de datos básicas y su transformación en estructuras más complejas. Se pretende que los alumnos puedan definir o elegir estructuras adecuadas para la resolución de problemas específicos.

### 3. Objetivos

#### 3.1. Objetivos Generales

Entregar al estudiante conceptos básicos utilizados en representación de datos y su organización en estructuras más complejas. Entregar técnicas y algoritmos adecuados para operar sobre las representaciones anteriores.

#### 3.2. Objetivos Específicos

Los estudiantes al término de la asignatura deberán estar en condiciones de:

- comprender estructuras de datos básicas
- especificar formalmente nuevas estructuras a partir de las estructuras de datos estudiadas
- definir y usar algoritmos asociados a estructuras de datos
- evaluar la complejidad de algoritmos básicos
- proponer estructuras a problemas específicos de programación

## 4. Contenido

1. Introducción al Análisis de Algoritmos
  - a) Técnicas de conteo
  - b) Análisis asintótico
  - c) Notaciones
  - d) Clases de complejidad estándares
  - e) Tiempo versus espacio
  - f) Análisis de algoritmos recursivos
2. Formalismo Funcional de especificación de estructuras de datos
3. Pilas, filas:
  - a) Representación contigua
  - b) Representación por encadenamiento
  - c) Manejo circular
4. Listas:
  - a) Listas generalizadas
  - b) Algoritmos con listas
  - c) Aplicaciones: manejo dinámico de memoria, compiladores
5. Tablas Hash
  - a) Acceso directo
  - b) Direccionamiento abierto
6. Árboles:
  - a) Conceptos básicos
  - b) Representación contigua y por encadenamiento
  - c) Árboles binarios
  - d) Operaciones de inserción, eliminación y caminamiento
7. Estructuras avanzadas
  - a) Árboles balanceados AVL
  - b) B-Trees
  - c) Heaps
  - d) Grafos
  - e) Sets
8. Algoritmos de ordenamiento

## Programación Semanal Preliminar (sujeta a imprevistos):

Mes: Día	Tema
<b>Marzo</b>	
Lunes 5	Introducción al Curso
Miércoles 7	Introducción a Técnicas de Conteo
Labs	Laboratorio: reconocimiento de ambiente de desarrollo
Lunes 12	Análisis de Algoritmos
Miércoles 14	Análisis de Algoritmos
Labs	Laboratorio: Introducción a C++
Lunes 19	Especificación funcional
Miércoles 21	Especificación funcional
Labs	Laboratorio: Introducción a C++
<b>Lunes 26</b>	<b>Control 1</b>
Miércoles 28	Pilas
Labs	Laboratorio: Manejo de Pilas
<b>Abril</b>	
Lunes 2	Pilas/Filas
Miércoles 4	Filas
Labs	Laboratorio: Manejo de Filas
Lunes 9	Listas Generalizadas
Miércoles 11	Listas Generalizadas
Labs	<b>Interrogación Labs</b>
Lunes 16	Aplicación de listas
Miércoles 18	Aplicación de listas
Labs	Laboratorio: Listas
<b>Lunes 23</b>	<b>Control 2</b>
Miércoles 25	Tablas Hash
Labs	<b>Interrogación Labs</b>
Lunes 30	Tablas Hash
<b>Mayo</b>	
Miércoles 2	Tablas Hash
Labs	Laboratorio: Tablas
Lunes 7	Arboles
Miércoles 9	Arboles Binarios
Labs	<b>Interrogación Labs</b>
Lunes 14	Arboles binarios
Miércoles 16	Arboles AVLs
Labs	Laboratorio: Arboles binarios
Lunes 21	Arboles AVL
<b>Miércoles 23</b>	<b>Control 3</b>
Labs	Laboratorio: Arboles binarios
Labs	<b>Interrogación labs</b>

Mes: Día	Tema
<b>Junio</b>	
Lunes 4	B-tree
Miércoles 7	Heaps
Labs	Laboratorio: Estructuras Avanzadas
Miércoles 11	Heaps
Labs	Laboratorio: Estructuras Avanzadas
Lunes 18	Ordenamiento
Miércoles 20	Ordenamiento
Labs	Laboratorio: Estructuras Avanzadas
Lunes 25	Ordenamiento
<b>Miércoles 27</b>	<b>Control 4</b>
Labs	Test 7: <b>Interrogación Labs</b>

## 5. Evaluación

La evaluación del curso se basa en 4 controles durante el semestre e interrogaciones. Las interrogaciones consisten en defender un trabajo que se haya hecho durante un Laboratorio, siendo DOS interrogaciones como mínimo al semestre. Las interrogaciones se darán en días específicos, teniendo cada grupo o alumno que interrogarse al menos una vez de cada dos interrogaciones programadas. Es decir, durante las dos primeras interrogaciones programadas, los grupos deben interrogarse al menos una vez, no pudiendo dejar más de una interrogación consecutiva sin darse. Las interrogaciones acerca de trabajos en laboratorios sólo podrán ser dadas para laboratorios a los cuales los alumnos hayan asistido. En caso de trabajo en grupo, los alumnos no podrán cambiar de grupo durante el semestre.

Entre los alumnos que voluntariamente se ofrezcan a dar interrogación, se escogerán los alumnos que tengan menos interrogaciones hasta ese momento.

La ponderación de promedios de tests (PT) y promedio de interrogaciones (PI) para la obtención de la nota final es:

$$NT = 0,8PT + 0,2PI$$

Los Tests son INDIVIDUALES. Los tests y trabajos en laboratorio DEBEN APROBARSE, es decir, se exige un promedio de test 4.0 para tanto laboratorios como tests.

Los alumnos tendrán derecho a dar un examen de repetición, el cual será equivalente a un 50% de la nota final obtenida por los tests en el curso.

## 6. Metodología

El curso contará con clases teóricas y laboratorios. Los trabajos prácticos tendrán un componente fuerte de programación.

## 7. Bibliografía

- Thomas Cormen, Charles Leiserson, Ronald Rivest and Clifford Stein: Introduction to Algorithms, MIT Press y McGraw-Hill Book Company, 2001, Second Edition.

- Narciso Martí, Yolanda Ortega, José Alberto Verdejo López: Estructuras de Datos y Métodos Algorítmicos. Pearson/Prentice Hall, 2003.
- Armando E. De Giusti: Algoritmos, Datos y Programas, con aplicaciones en Pascal, Delphi y Visual Da Vinci. Prentice Hall, 2001.
- Gregory L. Heileman: Estructuras de datos, algoritmos, y programacin orientada al objeto, Editorial McGraw-Hill,1998
- Apuntes de clase.