

## Programa



CURSO	: ESTRUCTURAS DE DATOS Y ALGORITMOS
TRADUCCIÓN	: DATA STRUCTURES AND ALGORITHMS
SIGLA	: IIC2133
CRÉDITOS	: 06 SCT-Chile / 10 UC
MÓDULOS	: 03
REQUISITOS	: IIC2233 y IIC1253
TIPO DE ASIGNATURA	: CÁTEDRA
CALIFICACIÓN	: ESTÁNDAR
DISCIPLINA	: INGENIERÍA

### I. DESCRIPCIÓN

El curso enseña las estructuras de datos fundamentales y sus algoritmos, tanto en la memoria principal y disco duro, haciendo hincapié en el uso típico, las ventajas y limitaciones de cada una. El curso también enseña las principales técnicas algorítmicas para resolver problemas de optimización discreta, colocando hincapié en el análisis cuantitativo de los algoritmos.

### II. OBJETIVOS

1. Identificar y explicar las estructuras de datos fundamentales (listas ligadas, colas, heaps, tablas de hash, árboles de búsqueda, bosques disjuntos), y los tipos de datos abstractos fundamentales (grafos, diccionarios, colas priorizadas, y conjuntos disjuntos).
2. Diseñar, implementar y comparar estructuras de datos.
3. Explicar qué es un algoritmo computacional.
4. Analizar un algoritmo para determinar su desempeño en términos de espacio y tiempo, corrección y completitud; en particular, analizar el desempeño de diferentes algoritmos de ordenación: ordenación por mezclas, ordenación basada en heaps y quicksort.
5. Aplicar técnicas algorítmicas específicas --algoritmos codiciosos, dividir para conquistar, programación dinámica-- para resolver problemas específicos.
6. Explicar los fundamentos de la teoría de grafos, y los siguientes algoritmos sobre grafos: búsquedas en amplitud y en profundidad, árboles de cobertura de costo mínimo, rutas de costo mínimo.

### III. CONTENIDOS

1. Introducción. El papel de los algoritmos en la computación y la importancia de las estructuras de datos en el diseño de algoritmos eficientes.
2. Estructuras de datos. Estructuras de datos elementales; tablas de hash; árboles de búsqueda binarios; árboles rojo-negros; árboles B; algoritmos para manejar estas estructuras: agregar, eliminar y tener acceso a objetos.
3. Algoritmos de ordenación. Heapsort; Quicksort; análisis de desempeño; ordenación en tiempo lineal.

4. Técnicas algorítmicas. Programación dinámica; algoritmos codiciosos; estructuras de datos para conjuntos disjuntos.

5. Representación y manejo de grafos. Representaciones matricial y por listas de adyacencias; algoritmos elementales: búsquedas en amplitud y en profundidad, ordenación topológica; árboles de cobertura mínimos; rutas más cortas.

#### IV. METODOLOGÍA

- Clases expositivas.
- Ayudantías.

#### V. EVALUACIÓN

- Pruebas.
- Proyectos.
- Tareas.

#### VI. BIBLIOGRAFÍA

Minima:

Cormen, T., C. Leiserson y R. Rivest. Introduction to algorithms. Cambridge, MIT, 1990.

Complementaria:

Main, M. y W. Savitch. Data structures and others objects using C++. Reading, Mass.: Addison Wesley, 1997.

Sedgewick, Robert. Algorithms in C++. 3ª Ed. Reading, Mass.: Addison Wesley, 1997. Parts 1-4.

PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
FACULTAD DE INGENIERÍA / Julio 2015