



PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE  
ESCUELA DE INGENIERIA  
DEPARTAMENTO DE CIENCIA DE LA COMPUTACION

**Diseño y Análisis de Algoritmos - IIC2283**  
**Programa de Curso**  
**2<sup>do</sup> semestre - 2023**

**Horario cátedra** : lunes y miércoles módulo 5, sala B13  
**Horario ayudantía** : viernes módulo 5, sala B13  
**Profesor** : Diego Arroyuelo (diego.arroyuelo@uc.cl)  
**Ayudantes** : TBA  
**Plataforma de apoyo** : Canvas, <https://github.com/darroyue/IIC2283>

## Objetivo

El objetivo del curso es introducir al alumno a las técnicas básicas y algunas técnicas avanzadas tanto para el diseño como para el análisis de la complejidad computacional de un algoritmo. Se dará especial énfasis a la comprensión del modelo computacional sobre el cual se diseña y analiza un algoritmo. Además, para cada una de las técnicas mostradas se estudiará algunos algoritmos que permiten entender su potencial, poniendo énfasis en la variedad e importancia de las áreas donde estos algoritmos son utilizados.

## Evaluación

La evaluación del curso estará basada en dos interrogaciones, un examen y al menos tres tareas. Las interrogaciones y el examen estarán orientadas a medir los conceptos fundamentales enseñados en el curso. En las tareas, los alumnos programarán en Python algoritmos para resolver distintos tipos de problemas, donde serán utilizadas las técnicas aprendidas en el curso.

Suponiendo que las notas en las interrogaciones son  $I_1$  e  $I_2$ , la nota del examen es  $E$ , y el promedio de las notas de tareas es  $\bar{T}$ , la nota final del curso se calcula de la siguiente forma:

$$NF = \frac{1}{2} \cdot \bar{T} + \frac{1}{2} \cdot \left( \frac{I_1 + I_2 + E}{3} \right).$$

Las fechas de las interrogaciones y el examen son las siguientes:

$I_1$  : viernes 8 de septiembre a las 17:30 hrs  
 $I_2$  : viernes 3 de noviembre a las 17:30 hrs  
Examen : jueves 7 de diciembre a las 08:20 hrs

# Contenido

1. Introducción
2. Análisis de la eficiencia de un algoritmo
  - a) Notaciones asintóticas
  - b) Ecuaciones de recurrencia: cambio de variables, inducción constructiva, teorema maestro
  - c) Análisis de la complejidad de un algoritmo en el peor caso
3. Análisis de la eficiencia de un algoritmo más allá del peor caso: análisis de caso promedio
4. Técnicas para demostrar cotas inferiores: mejor estrategia del adversario, árboles de decisión y reducciones
5. Técnicas fundamentales de diseño de algoritmos
  - a) Dividir para conquistar
  - b) Programación dinámica
  - c) Algoritmos codiciosos
6. Transformaciones de dominio
  - a) Representación, evaluación e interpolación de polinomios y la transformada rápida de Fourier
7. Algoritmos aleatorizados
  - a) Algoritmos de Monte Carlo: igualdad de polinomios
  - b) Algoritmos de Las Vegas: cálculo de la mediana de una lista
  - c) Hashing universal
8. Algoritmos en teoría de números
  - a) Aritmética modular
  - b) Algoritmos básicos: exponenciación rápida, cálculo del máximo común divisor, el algoritmo de Euclides extendido y el cálculo del inverso modular
  - c) Un algoritmo de Monte Carlo para la verificación de primalidad

# Bibliografía

1. Transparencias de clases.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest y Clifford Stein. *Introduction to Algorithms*, 3ª edición. MIT Press, 2009.
3. Gilles Brassard y Paul Bratley. *Algorithmics: Theory and Practice*, 1ª edición. Prentice Hall, 1988.
4. Rajeev Motwani y Prabhakar Raghavan. *Randomized Algorithms*, 1ª edición, 1995.
5. Jon Kleinberg y Éva Tardos. *Algorithm Design*, 1ª edición. Pearson, 2005.
6. Michael Mitzenmacher y Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.