



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

Diseño y Análisis de Algoritmos - IIC2283
Programa de Curso
2^{do} semestre - 2020

Horario cátedra : lunes y miércoles módulo 5
Horario ayudantía : viernes módulo 5
Profesor : Marcelo Arenas (marenas@ing.puc.cl)
Ayudantes : Bernardo Barías (bjbarias@uc.cl), Pablo Messina (pamessina@uc.cl),
Martín Muñoz (mmunos@uc.cl) y Ricardo Rodríguez (rrodriguez@uc.cl)
URL : <https://marceloarenas.cl/iic2283-20>

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 cuatro tareas. Las interrogaciones y el examen estarán orientadas a medir los conceptos fundamentales enseñados en el curso. En las tareas, los alumnos programaran 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 : miércoles 7 de octubre
 I_2 : miércoles 18 de noviembre
Examen : sábado 12 de diciembre

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. Técnicas fundamentales de diseño de algoritmos
 - a) Dividir para conquistar
 - b) Programación dinámica
 - c) Algoritmos codiciosos
4. Transformaciones de dominio
 - a) Representación, evaluación e interpolación de polinomios y la transformada rápida de Fourier
5. 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
6. 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
7. Técnicas para demostrar cotas inferiores: mejor estrategia del adversario, árboles de decisión y reducciones
8. Análisis de la eficiencia de un algoritmo más allá del peor caso
 - a) Análisis en el caso promedio: Quicksort
 - b) Análisis amortizado de algoritmos

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.