



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 - 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 : Ernesto Ayala (ernesto.ayala@uc.cl), Mauricio Cari (mauricio.cari@uc.cl), Felipe Guzmán (faguzman2@uc.cl), Ricardo Rodríguez (rrrodriguez@uc.cl), Amaranta Salas (afsalas@uc.cl), Tomás Vergara (tomvergara@uc.cl).
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 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 las notas en las tareas

son T_1 , T_2 y T_3 , la nota final del curso se calcula de la siguiente forma:

$$NF = \frac{1}{2} \cdot \left(\frac{T_1 + T_2 + T_3}{3} \right) + \frac{1}{2} \cdot \left(\frac{I_1 + I_2 + E}{3} \right).$$

La condición de aprobación es $NF \geq 3,95$.

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

Las fechas de publicación del enunciado y entrega de las tareas aparecen en la siguiente tabla:

	Publicación enunciado	Entrega
Tarea 1	Miércoles 13 de septiembre	Martes 26 de septiembre
Tarea 2	Miércoles 11 de octubre	Viernes 27 de octubre
Tarea 3	Lunes 6 de noviembre	Lunes 20 de noviembre

La entrega será en la fecha estipulada hasta las 23:59 horas. La publicación del enunciado será durante la mañana de la fecha estipulada.

Adicionalmente, cada estudiante puede acceder a realizar entregas atrasadas para las tareas obteniendo un descuento sobre su nota como penalización. El descuento d_i asociado a la i -ésima tarea es

$$d_i = \begin{cases} 0 & \text{si entrega sin atraso,} \\ 0,5 & \text{si entrega con menos de 24 horas de atraso,} \\ 1,5 & \text{si entrega con más de 24 pero menos de 48 horas de atraso,} \\ 3,0 & \text{si entrega con más de 48 pero menos de 72 horas de atraso,} \\ 7,0 & \text{en otro caso.} \end{cases}$$

Así, siendo T'_i la nota correspondiente según la rúbrica de la i -ésima tarea, la nota T_i de la i -ésima tarea se calcula como

$$T_i = \max \{T'_i - d_i, 1, 0\}$$

Dicho eso, cada estudiante cuenta con la opción de utilizar un cupón de extensión de plazo para la entrega de una tarea. Este cupón se puede utilizar sólo una vez durante el semestre y permitirá extender el plazo de entrega de una tarea sin necesidad de una justificación. La extensión permite al estudiante entregar hasta 72 horas después de la entrega original, anulando el descuento por atraso.

Vale mencionar que la recepción de entregas atrasadas es un beneficio excepcional para el estudiante, por lo que las horas de atraso serán contabilizadas incluso si es que los días posteriores a la entrega original son feriados (por ejemplo si la entrega es un jueves, como máximo se puede entregar atrasado hasta el siguiente domingo). Fuera del uso del cupón de extensión, no se harán excepciones sobre estos plazos.

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.