



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

IIC2283 – Diseño y Análisis de Algoritmos  
**Programa de Curso**  
2025-2

**Profesores** : Diego Arroyuelo (Sección 1) ([diego.arroyuelo@uc.cl](mailto:diego.arroyuelo@uc.cl))  
Juan Pablo Castillo (Sección 2) (email temporal [juan.castillog@usm.cl](mailto:juan.castillog@usm.cl))  
**Horario de cátedra** : lunes y miércoles módulo 5, sala CS-102 (Sección 1) y sala B13 (Sección 2)  
**Ayudantes** : Isabella Cherubini, Ayudante Coordinadora ([isabella.cherubini@uc.cl](mailto:isabella.cherubini@uc.cl))  
**Horario de ayudantía** : viernes módulo 5, sala B23 (Sección 1) y BC24 (Sección 2)  
**Plataforma de apoyo** : Canvas

## 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.

## Metodología

El curso se basa en 2 clases expositivas semanales, de 70 minutos cada una. Además de las clases expositivas, durante el semestre se realizará una ayudantía semanal, con el objetivo de complementar, reforzar y guiar el trabajo de los estudiantes.

## 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 estudiados 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  $E$ , el promedio de interrogaciones se calcula como:

$$PE = \frac{I_1 + I_2 + E}{3}.$$

Si las notas en las tareas son  $T_1$ ,  $T_2$  y  $T_3$ , el promedio de tareas se calcula como:

$$PT = \frac{T_1 + T_2 + T_3}{3}.$$

La nota final del curso se calcula de la siguiente forma:

$$NF = \frac{PE}{2} + \frac{PT}{2}.$$

El curso se aprueba con nota  $NF$  si se cumplen las tres siguientes condiciones:

- $PE \geq 3.7$ ,
- $PT \geq 3.7$ , y
- $NF \geq 3.95$ .

Si al menos una de esas condiciones no se cumple, la nota final del curso se calculará como  $\min\{NF, 3.9\}$ .

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

$I_1$	:	sábado 27 de septiembre a las 8:20 hrs.
$I_2$	:	martes 28 de octubre a las 17:30 hrs.
Examen	:	viernes 12 de diciembre a las 13:30 hrs.

En caso de ausentarse a las interrogaciones 1 o 2 de forma justificada ante la Dirección de Pregrado, el estudiante tiene la posibilidad de rendir preguntas adicionales en el Examen para reemplazar su nota en solo una de las evaluaciones faltantes. No será posible acceder a las pruebas recuperativas sin una justificación. La inasistencia al Examen sigue las normas de nota P y debe ser justificada ante la Dirección de Pregrado, quienes están facultados para dejar su nota como pendiente en el curso.

Las fechas de publicación del enunciado y entrega de las tareas se darán a conocer a la brevedad.

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

$$d_i = \begin{cases} 0 & \text{si entrega sin atraso,} \\ 1,5 & \text{si entrega con menos de 24 horas de atraso,} \\ 3 & \text{si entrega con más de 24 pero menos de 48 horas de atraso,} \\ 6,0 & \text{en otro caso.} \end{cases}$$

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 solo 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 48 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 días no hábiles.

## 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.