



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

**Diseño y Análisis de Algoritmos - IIC2283**  
**Programa de Curso**  
**2<sup>do</sup> semestre - 2022**

<b>Profesor</b>	Nicolás Van Sint Jan, <a href="mailto:nicovsj@uc.cl">nicovsj@uc.cl</a> , oficina 10, DCC
<b>Ayudantes</b>	Dante Pinto, <a href="mailto:drpinto@uc.cl">drpinto@uc.cl</a> , (Cátedra/Coordinación) Benjamín Farías, <a href="mailto:bffarias@uc.cl">bffarias@uc.cl</a> (Corrección) Julián García, <a href="mailto:jgarcg@uc.cl">jgarcg@uc.cl</a> (Corrección) Nicholas Mc-Donnell, <a href="mailto:namcdonnell@uc.cl">namcdonnell@uc.cl</a> (Tareas)
<b>Clases</b>	Lunes y miércoles módulo 5 en Sala B13
<b>Ayudantías</b>	Viernes módulo 5 en Sala BC24
<b>URL</b>	<a href="https://github.com/IIC2283/DAA-2022-2">github.com/IIC2283/DAA-2022-2</a>
<b>Calendario</b>	<a href="https://bit.ly/3zJHqTw">https://bit.ly/3zJHqTw</a>

## Objetivo

El objetivo del curso es introducir al estudiante a 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 clases expositivas de 80 mins. cada una. Se realizará un promedio de 2 clases semanales. Además de las clases expositivas, durante el semestre se realizará una ayudantía semanal, con el objetivo de reforzar o guiar el trabajo de los estudiantes.

## Contenidos

### 1. Introducción

### 2. Análisis de la eficiencia de un algoritmo

- 2.1. Notaciones asintóticas
- 2.2. Ecuaciones de recurrencia: cambio de variables, inducción constructiva, teorema maestro
- 2.3. Análisis de la complejidad de un algoritmo en el peor caso

### 3. Técnicas fundamentales de diseño de algoritmos

- 3.1. Dividir para conquistar
- 3.2. Programación dinámica
- 3.3. Algoritmos codiciosos

### 4. Transformaciones de dominio

- 4.1. Representación, evaluación e interpolación de polinomios y la transformada rápida de Fourier

### 5. Algoritmos aleatorizados

- 5.1. Algoritmos de Monte Carlo: igualdad de polinomios
- 5.2. Algoritmos de Las Vegas: cálculo de la mediana de una lista
- 5.3. Hashing universal

### 6. Algoritmos en teoría de números

- 6.1. Aritmética modular
- 6.2. Algoritmos básicos: exponenciación rápida, cálculo del máximo común divisor, el algoritmo de Euclides extendido y el cálculo del inverso modular
- 6.3. 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

- 8.1. Análisis en el caso promedio: Quicksort
- 8.2. Análisis amortizado de algoritmos Técnicas para el análisis amortizado: análisis global, contabilidad de costos, función potencial
- 8.3. Heaps binomiales
- 8.4. Heaps de Fibonacci

## Evaluación

La evaluación del curso estará basada en dos interrogaciones, un examen y cinco tareas de programación.

**Tareas.** Habrán cinco tareas individuales durante el curso. Cada una consistirá en un problema a resolver utilizando Python, donde se pedirá que la solución respete una complejidad algorítmica dada. Para la resolución de estas, será necesario diseñar algoritmos que utilicen las técnicas aprendidas en el curso.

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

	Publicación enunciado	Entrega
Tarea 1	Lunes 29 de agosto	Miércoles 7 de septiembre
Tarea 2	Lunes 12 de septiembre	Jueves 22 de septiembre
Tarea 3	Lunes 3 de octubre	Jueves 13 de octubre
Tarea 4	Lunes 17 de octubre	Miércoles 26 de octubre
Tarea 5	Lunes 7 de noviembre	Miércoles 16 de noviembre

La entrega será para la fecha estipulada hasta las 23:59 horas (entrega mediante un repositorio de GitHub). 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, sin que se aplique descuento alguno.

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 domingo de esa misma semana). Fuera del uso del cupón de extensión, no se harán excepciones sobre estos plazos.

**Interrogaciones y examen.** Las interrogaciones y el examen estarán orientados a medir los conceptos fundamentales enseñados en el curso. Las fechas de rendición son las siguientes:

	Fecha
Interrogación 1	Viernes 30 de septiembre
Interrogación 2	Miércoles 2 de noviembre
Examen	Viernes 2 de diciembre

Las interrogaciones son presenciales y están diseñadas para tener una duración de 2 a 3 horas. Se realizarán a las 18:30 horas en la fecha asignada y el profesor estará disponible para resolver dudas. El examen es presencial a las 9:00 horas en la fecha asignada.

En caso de tener algún problema de fuerza mayor para asistir a una interrogación la nota del examen reemplazará la nota de esta interrogación (referirse al cálculo de notas del curso más abajo). Para hacer efectivo este reemplazo, la ausencia a la interrogación no necesita ser justificada. En caso de tener un problema mayor para rendir el examen, el estudiante debe presentar un justificativo según las reglas de la Escuela de Ingeniería en la Dirección de Pregrado, quedando con Nota P y rendirá el examen a comienzos del próximo semestre.

El profesor no se hará responsable por tope de horarios con interrogaciones o exámenes de cursos que se regulen por la programación académica de la Escuela de Ingeniería. Es responsabilidad del estudiante revisar estos topes de horarios para así no tener problemas durante el semestre.

**Notas del curso.** Para el cálculo de las notas finales del curso, usaremos  $AVG_n(N_1, \dots, N_k)$  como el promedio aritmético de los  $n$  mayores valores de la lista de  $k$  notas  $N_1, \dots, N_k$ .

El promedio de tareas **PT** calculada con las notas de las cinco tareas  $(T_1, \dots, T_5)$  se calculará según

$$\mathbf{PT} = AVG_4(T_1, T_2, T_3, T_4, T_5).$$

Por otro lado, el promedio de evaluaciones **PE** de las dos interrogaciones ( $I_1, I_2$ ) y examen ( $E$ ) se calculará acorde a

$$\mathbf{PE} = \text{AVG}_3(I_1, I_2, E, E).$$

El promedio final **PF** se calcula como

$$PF = \frac{1}{2} \cdot \mathbf{PT} + \frac{1}{2} \cdot \mathbf{PE}.$$

El curso se aprueba si, y solo si, todas las siguientes condiciones se cumplen:

- Promedio de tareas mayor o igual a 2.95 ( $\mathbf{PT} \geq 2.95$ ),
- Promedio de evaluaciones mayor o igual a 3.95 ( $\mathbf{PE} \geq 3.95$ ) y
- Promedio final mayor o igual a 3.95 ( $\mathbf{PF} \geq 3.95$ ).

La nota final (**NF**) del curso entonces corresponde a

$$\mathbf{NF} = \begin{cases} \mathbf{PF} & \text{si aprueba el curso} \\ \min\{\mathbf{PF}, 3.9\} & \text{si no aprueba el curso} \end{cases}$$

Las notas del curso **PT**, **PE** y **PF** se computan redondeadas al segundo decimal.

**Proceso de corrección.** El protocolo de corrección y recorreción de las evaluaciones será el siguiente.

- Tareas
  1. Las tareas serán corregidas mediante el uso de tests automatizados de dificultad incremental.
  2. Una vez publicada la nota, se podrá solicitar recorreción mediante GitHub adjuntando un nuevo *commit* donde se realicen cambios pequeños. El proceso de solicitud para recorregir será cuando corresponda.
- Interrogaciones
  1. Después de la fecha de rendición de una interrogación, el cuerpo docente tendrá un plazo de dos semanas aproximadamente para la entrega de notas y el feedback asociado.
  2. Se anunciará un día de recorreción presencial donde cada estudiante puede preguntar y solicitar la recorreción de su interrogación explicando directamente al ayudante corrector. En caso de no poder asistir a esta instancia el estudiante puede solicitar la recorreción por escrito.

**Política de copia.** En caso de copia (ver documento adjunto “Código de Honor de la Escuela de Ingeniería”) se tomarán las medidas dispuestas por la política de integridad académica del Departamento de Ciencia de la Computación (ver documento adjunto “Política de Integridad Académica del Departamento de Ciencia de la Computación”).

## Bibliografía

Durante el curso se recomiendan los siguientes libros de estudio:

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

# COMPROMISO DE CÓDIGO DE HONOR

Este curso adscribe el Código de Honor establecido por la Universidad, el que es vinculante. Todo trabajo evaluado en este curso debe ser propio. En caso de que exista colaboración permitida con otros s, el trabajo deberá referenciar y atribuir correctamente dicha contribución a quien corresponda. Como alumno es su deber conocer el Código de Honor ([www.uc.cl/codigodehonor](http://www.uc.cl/codigodehonor)).

## Política de Integridad Académica del Departamento de Ciencia de la Computación

Se espera que los estudiantes de la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile mantengan altos estándares de honestidad académica, acorde al Código de Honor de la Universidad. Cualquier acto deshonesto o fraude académico está prohibido; los estudiantes que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada estudiante conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Pregrado de la Escuela de Ingeniería (Disponible en SIDING, en la sección Pregrado/Asuntos Estudiantiles/Reglamentos/Reglamentos en Ingeniería/Integridad Académica).

Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un estudiante para los efectos de la evaluación de un curso debe ser hecho **individualmente** por el estudiante, **sin apoyo en material de terceros**. Por “trabajo” se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros.

En particular, si un estudiante copia un trabajo, o si a un estudiante se le prueba que compró o intentó comprar un trabajo, **obtendrá nota final 1.1 en el curso** y se solicitará a la Dirección de Pregrado de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral. Por “copia” se entiende incluir en el trabajo presentado como propio, partes hechas por otra persona. En caso que corresponda a “copia” a otros estudiantes, la sanción anterior se aplicará a todos los involucrados. En todos los casos, se informará a la Dirección de Pregrado de la Escuela de Ingeniería para que tome sanciones adicionales si lo estima conveniente.

Obviamente, está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, **siempre y cuando se incluya la referencia correspondiente**.

Lo anterior se entiende como complemento al Reglamento del Estudiante de la Pontificia Universidad Católica de Chile:

<http://admisionyregistros.uc.cl/alumnos/informacion-academica/reglamentos-estudiantiles>

Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.