

## PROGRAMA DE CURSO

### ALGORITMOS Y ESTRUCTURAS DE DATOS

#### A. Antecedentes generales del curso:

Departamento	Ciencias de la computación					
Nombre del curso	Algoritmos y estructuras de datos	Código	CC3001	Créditos	6	
Nombre del curso en inglés	<i>Algorithms and data structures</i>					
Horas semanales	Docencia	3	Auxiliares	1,5	Trabajo personal	5,5
Carácter del curso	Obligatorio	X		Electivo		
Requisitos	CC1002: Introducción a la programación, 90 créditos					

#### B. Propósito del curso:

El curso Algoritmos y Estructuras de Datos tiene como propósito que los y las estudiantes logren un dominio de los conceptos básicos de diseño de algoritmos eficientes (iterativos y recursivos), de análisis de algoritmos, y de estructuras de datos (arreglos, estructuras enlazadas, grafos).

Los y las estudiantes deben ser capaces de aplicar estos conceptos al diseño, implementación, y evaluación de los tipos de datos abstractos más usados (listas, pilas, colas, colas de prioridad, diccionario), así como de los algoritmos fundamentales (búsqueda, ordenación, algoritmos para grafos y para búsqueda en texto) utilizados en Ciencias de la Computación.

El curso tributa a las siguientes competencias específicas (CE) y genéricas (CG):

CE1: Analizar problemas computacionales, construir modelos, expresándolos en representaciones y lenguajes formales adecuados.

CE2: Analizar, diseñar y/o adaptar algoritmos y estructuras de datos que cumplan con las garantías requeridas de correctitud y eficiencia.

CE3: Gestionar bases de datos utilizando modelos, lenguajes de consulta asociados, técnicas eficientes de acceso a datos y aplicación de políticas de seguridad, con la finalidad de obtener información relevante.

CG1: Comunicación académica y profesional

Comunicar en español de forma estratégica, clara y eficaz, tanto en modalidad oral como escrita, puntos de vista, propuestas de proyectos y resultados de investigación fundamentados, en situaciones de comunicación compleja, en ambientes sociales, académicos y profesionales.

**CG3: Compromiso ético**

Actuar de manera responsable y honesta, dando cuenta en forma crítica de sus propias acciones y sus consecuencias, en el marco del respeto hacia la dignidad de las personas y el cuidado del medio social, cultural y natural.

**C. Resultados de aprendizaje:**

Competencias específicas	Resultados de aprendizaje
CE1	RA1: Modela situaciones reales asociadas a problemas de procesamiento de información, considerando conceptos básicos de programación, tales como, correctitud, invariantes y eficiencia, entre otros.
CE2	RA2: Diseña algoritmos eficientes para proponer diversas soluciones a problemas de procesamiento de información, considerando su descripción formal y el uso de estructuras de datos.
	RA3: Compara distintas soluciones posibles a diversos problemas, a fin de seleccionar aquella que resulta óptima de acuerdo con la naturaleza del problema de procesamiento de información.
CE3	RA4: Implementa, en un lenguaje de programación dado, la descripción formal de una solución a problemas de procesamientos de información, considerando corrección, eficiencia y el uso de herramientas computacionales.
Competencias genéricas	Resultados de aprendizaje
CG1, CG3	RA5: Reporta, de forma clara y coherente, las soluciones a problemas de procesamiento de información, considerando su modelación formal e implementación, cuyos resultados respalda, en gráficos o figuras.

**D. Unidades temáticas:**

Número	RA al que tributa	Nombre de la unidad	Duración en semanas
1	RA4	Conceptos básicos de programación	2 semanas
Contenidos		Indicador de logro	
1.1. Conceptos básicos de programación. 1.2. Correctitud de programas, invariantes. Ejemplo: algoritmos elementales de ordenación.		La/el estudiante: 1. Identifica y utiliza algoritmos elementales de ordenación 2. Escribe programas básicos, razonando sobre su correctitud y eficiencia.	
Bibliografía de la unidad		[12] Capítulo 1.	

Número	RA al que tributa	Nombre de la unidad	Duración en semanas
2	RA2, RA3, RA4, RA5	Diseño y análisis de algoritmos	3 semanas
Contenidos		Indicador de logro	
2.1. Conceptos matemáticos. 2.2. Notación "O". 2.3. Análisis de algoritmos sencillos. 2.4. Inducción simple y reforzada. 2.5. Recursividad, dividir para reinar. 2.6. Programación dinámica. 2.7. Algoritmos avaros ( <i>greedy</i> ) y su complejidad. 2.8. Casos de estudios: subsecuencia de suma máxima, subsecuencia común más larga, multiplicación de matrices.		La/el estudiante: <ol style="list-style-type: none"> <li>1. Efectúa análisis matemáticos de algoritmos, eligiendo el enfoque según la naturaleza del problema a resolver.</li> <li>2. Diseña algoritmos eficientes, utilizando métodos de programación dinámica, dividir para reinar y algoritmos avaros.</li> <li>3. Implementa soluciones en lenguaje de programación a problemas de procesamiento de información.</li> <li>4. Reporta formalmente la solución elegida, evaluándola de acuerdo a criterios de correctitud y eficiencia y considerando aspectos éticos.</li> </ol>	
Bibliografía de la unidad		[1] Capítulos 2 y 3.	

Número	RA al que tributa	Nombre de la unidad	Duración en semanas
3	RA2, RA3, RA4, RA5	Estructuras de datos elementales	1 semana
Contenidos		Indicador de logro	
3.1. Técnicas de estructuras de datos elementales. 3.1.1. Arreglos. 3.1.2. Punteros y listas enlazadas. 3.1.3. Árboles.		La/el estudiante: <ol style="list-style-type: none"> <li>1. Utiliza técnicas básicas de estructuras de datos elementales a partir de ejemplos con el objeto de resolver problemas considerando su descripción formal y el uso de estructuras de datos.</li> <li>2. Evalúa la estructura de datos elementales, seleccionando, según su naturaleza, la más idónea en la resolución de un problema.</li> <li>3. Describe formalmente problemas de procesamiento de datos, según su naturaleza.</li> <li>4. Explica la descripción formal e implementación de la solución elegida para un problema de procesamiento de información.</li> </ol>	
Bibliografía de la unidad		[1] Capítulo 4.	

Número	RA al que tributa	Nombre de la unidad	Duración en semanas
4	RA2, RA3, RA4	Tipos de datos abstractos (TDAs)	1 semana
Contenidos		Indicador de logro	
4.1. Tipos de datos abstractos (TDAs). 4.2. Pilas. 4.3. Colas. 4.4. Colas de Prioridad.		La/el estudiante: <ol style="list-style-type: none"> <li>Resuelve problemas de procesamiento de información, considerando datos abstractos.</li> <li>Utiliza distintos tipos de datos abstractos para la implementación de la solución a problemas de procesamiento de información.</li> <li>Implementa en lenguaje de programación la modelación formal de soluciones a problemas de procesamiento de información.</li> <li>Evalúa la eficiencia de la implementación de la solución, al definir y utilizar un tipo de dato abstracto.</li> </ol>	
Bibliografía de la unidad		[1] Capítulo 5.	

Número	RA al que tributa	Nombre de la unidad	Duración en semanas
5	RA2, RA3, RA4	Diccionarios	3 semanas
Contenidos		Indicador de logro	
5.1. Búsqueda secuencial. 5.2. Búsqueda binaria. 5.3. Árboles de búsqueda binaria. 5.4. Árboles balanceados. 5.5. <i>Skip Lists</i> . 5.6. <i>Hashing</i> . 5.7. Árboles digitales.		La/el estudiante: <ol style="list-style-type: none"> <li>Analiza los principales métodos de búsqueda en memoria central, con respecto a la eficiencia en el uso de memoria y tiempo.</li> <li>Aplica los principales métodos de búsqueda en memoria central para implementar estructuras eficientes en algún problema que se les presente.</li> <li>Compara los distintos modelos de diccionarios, considerando ventajas y desventajas.</li> <li>Analiza la eficiencia de los modelos en cuanto al uso de memoria y velocidad de ejecución.</li> <li>Selecciona la solución a un problema, considerando los distintos modelos de diccionarios.</li> <li>Implementa en lenguaje de programación, soluciones a problemas de procesamiento de información.</li> </ol>	
Bibliografía de la unidad		[1] Capítulo 6.	

Número	RA al que tributa	Nombre de la unidad	Duración en semanas
6	RA4	Ordenación	1,5 semanas
Contenidos		Indicador de logro	
6.1. <i>Quicksort, Quickselect, select lineal.</i> 6.2. <i>Heapsort.</i> 6.3. <i>Mergesort.</i> 6.4. <i>Radix sort.</i>		La/el estudiante: <ol style="list-style-type: none"> <li>Analiza cuáles algoritmos de ordenación son más eficientes, según el tipo de problema que deben resolver.</li> <li>Aplica, por su eficiencia y posibilidad de uso, los principales algoritmos de ordenación, según el tipo de problema que se le presente.</li> <li>Implementa en un lenguaje de programación dado, la descripción formal de una solución a problemas de procesamiento de información.</li> </ol>	
Bibliografía de la unidad		[1] Capítulo 7.	

Número	RA al que tributa	Nombre de la unidad	Duración en semanas
7	RA1, RA2, RA4	Búsqueda en texto y compresión de datos	1,5 semanas
Contenidos		Indicador de logro	
7.1. <i>Knuth- Morris-Pratt.</i> 7.2. <i>Boyer-Moore.</i> 7.3. Métodos de compresión de datos.		La/el estudiante: <ol style="list-style-type: none"> <li>Aplica el algoritmo de texto, seleccionando el más apropiado según su eficiencia.</li> <li>Resuelve problemas asociados a una situación real utilizando, por su eficacia en el procesamiento de información, métodos de compresión.</li> <li>Implementa soluciones a problemas, en el contexto de búsqueda de textos y compresión de datos.</li> </ol>	
Bibliografía de la unidad		[1] Capítulos 8 y 9.	

Número	RA al que tributa	Nombre de la unidad	Duración en semanas
8	RA1, RA2, RA3, RA4, RA5	Algoritmos para grafos	2 semanas
Contenidos		Indicador de logro	
8.1. Representación de grafos. 8.2. Búsqueda en profundidad. 8.3. Árbol cobertor mínimo: Kruskal, Prim. 8.4. Distancias mínimas: <i>Dijkstra</i> , Floyd. Cerradura transitiva: <i>Warshall</i> .		La/el estudiante: <ol style="list-style-type: none"> <li>1. Identifica los principales algoritmos para la representación de grafos.</li> <li>2. Modela situaciones reales de problemas de procesamiento de información, utilizando grafos.</li> <li>3. Compara las distintas soluciones obtenidas de un problema de procesamiento de información, según su eficiencia, seleccionando aquella que resulta óptima por la naturaleza del problema.</li> <li>4. Implementa, en lenguaje de programación, la descripción formal de una solución, según criterios de corrección y eficiencia.</li> <li>5. Reporta, de forma clara, la implementación de una solución en lenguaje de programación.</li> </ol>	
Bibliografía de la unidad		[1] Capítulo 10.	

### E. Estrategias de enseñanza - aprendizaje:

La metodología de enseñanza y aprendizaje fomenta la participación del estudiante en el aula. Las clases son principalmente:

- Clase expositiva o usando "*flipped classroom*", en donde los y las estudiantes identifican los conceptos básicos de estructuras de datos y de diseño de algoritmos eficientes.
- Trabajo dirigido en clase auxiliar, en donde los y las estudiantes resuelven problemas usando los conceptos vistos en cátedra, con acompañamiento del profesor auxiliar.
- Resolución de problemas, a partir de ejemplos que se le presentan.

A lo anterior se le suman actividades de evaluación formativa que deben ser desarrolladas con el computador, las que son enviadas a través de U-Cursos.

## F. Estrategias de evaluación:

El curso tiene distintas instancias de evaluación de proceso, entre ellas:

Tipo de evaluación	Resultado de aprendizaje (RA) asociado a la evaluación
• Hito 1: Unidades 1, 2, 3, y 4.	Evalúa:RA1,RA2,RA3, RA4,RA5
• Hito 2: Unidades 1, 2, 3, 4, 5 y 6.	Evalúa RA1,RA2,RA3, RA4,RA5
• Evaluación terminal: todas las unidades del programa.	Evalúa:RA1,RA2,RA3, RA4,RA5
• Actividades de evaluación formativa (por ejemplo: tareas, ejercicios, etc.).	Evalúa:RA1,RA2,RA3, RA4,RA5

*Se espera por parte de los estudiantes que resuelvan con honestidad y responsabilidad las tareas y ejercicios que se le asignan. En estas actividades se revisan, discuten y luego cada uno entrega sus resultados, los que deben ser trabajados de manera individual, con análisis propio y justificando con sus propias palabras los resultados correspondientes.*

## G. Recursos bibliográficos:

### Bibliografía obligatoria:

- [1] Apunte de Algoritmos y estructuras de datos (2020). URL: [www.dcc.uchile.cl/ppoblete/AED/](http://www.dcc.uchile.cl/ppoblete/AED/).

### Bibliografía complementaria:

- [2] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. Introduction to Algorithms, Third Edition (3rd. ed.). The MIT Press.
- [3] Donald E. Knuth. 1997. The art of computer programming, volume 1 (3rd ed.): fundamental algorithms. Addison Wesley Longman Publishing Co., Inc., USA.
- [4] Donald E. Knuth. 1998. The art of computer programming, volume 3: (2nd ed.) sorting and searching. Addison Wesley Longman Publishing Co., Inc., USA.
- [5] M.H. Alsuwaiyel, Algorithms – Design Technique and Analysis (Revised Edition), [http://dl.booktolearn.com/ebooks2/computer/algorithms/9789814723640\\_algorithms\\_cf99.pdf](http://dl.booktolearn.com/ebooks2/computer/algorithms/9789814723640_algorithms_cf99.pdf).

## H. Datos generales sobre elaboración y vigencia del programa de curso:

Vigencia desde:	Otoño, 2021
Elaborado por:	Nelson Baloian, Patricio Poblete
Validado por:	Enviado a revisión entre pares Benjamín Bustos y Jeremy Barbay. Validación CTD de Computación. Validación final por Nelson Baloian, Jeremy Barbay, Benjamin Bustos y Patricio Poblete. Retroalimentación 20 de octubre de 2023, con profesor Nelson Baloian.
Revisado por:	Área de Gestión Curricular