



Licenciatura en Ciencias de la Computación
Licenciatura en Ciencias de Datos
FCEN UBA

Técnicas de Diseño de Algoritmos (TDA)

Programa
Febrero 2024

Facultad de Cs. Exactas y Naturales. UNIVERSIDAD DE BUENOS AIRES
Cuatr - 1ro 2024 - Turno TARDE
Técnicas de Diseño de Algoritmos (TDA)

Docentes

Profesores

Bonomo Flavia (*)
Lin Min Chih
Platzer Emilio
Soulignac Francisco Juan (*)

JTP

Iglesias Matias
Terlisky Pablo Ezequiel (*)

Aytes 1ra

Braier Julián (*)
Brandwein Eric

Aytes 2da

Amster Martín (*)
Comaneetz Ezequiel
Dinkel Ayelen
Frassia Fernando
Laks Joaquín (*)
Nores Manuel
Pages Julieta Belen (*)
Raffo Leandro Javier
Umfurer Alberto

Aula

Magna [a confirmar]

Email

Les pedimos no enviar mails a docentes.

Horarios de consulta:

Miercoles 21:00 hs en clase

Clases

Manana (*)

Lunes (T), Miércoles (P) 9:00 a 13:00

Tarde

Lunes (T) 18:00 a 22:00 Miércoles (P) 17:00 a 21:00

[Link Campus, pagina, etc]

() **TM.** Este programa es para el **turno TARDE**. El turno MANANA es “otra materia”. En el sentido de que pueden llevar distinto ritmo y diferir ligeramente en régimen de aprobación.*

Descripción del curso

El curso de Técnicas de Diseño de Algoritmos (TDA, ex Algo 3) es una exploración de los principios fundamentales y las metodologías avanzadas empleadas en el diseño y análisis de algoritmos para resolver problemas computacionales complejos. A lo largo del curso, los estudiantes se sumergirán en una variedad de técnicas algorítmicas claves, incluyendo búsqueda iterativa, divide y vencerás, backtracking, y programación dinámica. Se hará énfasis en la aplicación práctica de estas técnicas en la resolución de problemas clásicos asociados con estructuras de datos, grafos, cadenas y problemas geométricos, permitiendo a los estudiantes no solo comprender las teorías subyacentes sino también cómo implementar soluciones eficientes en situaciones del mundo real.

Cada unidad del curso está diseñada para construir una base sólida en teoría de algoritmos y al mismo tiempo fomentar la aplicación práctica mediante ejercicios de programación, análisis de complejidad y optimización de soluciones. Se pondrá especial énfasis en la teoría de grafos, explorando técnicas para el recorrido de grafos, y el manejo eficiente de datos no estructurados y estructurados, preparando a los estudiantes para aplicar estas habilidades en la resolución heurística de problemas de optimización y en el diseño de algoritmos que abordan la frontera de lo computacionalmente tratable.

Mediante una combinación de clases teóricas, prácticas y proyectos prácticos, los estudiantes desarrollarán habilidades críticas en el diseño de algoritmos, análisis de complejidad, y la aplicación de estructuras de datos apropiadas para optimizar el rendimiento. Al finalizar el curso, los participantes estarán equipados con un profundo conocimiento teórico y práctico, listos para aplicar técnicas algorítmicas avanzadas en la solución de problemas computacionales complejos.

Es la misma materia que Algoritmos y Estructura de Datos 3, para quienes tengan plan de estudios con dicha materia.

Correlatividades

Previas:

- CBC
- Introd. a la Programación (IP), Algebra
- Algoritmos y Estructuras de Datos (AED)

Posteriores:

- Complejidad Computacional (CC)
- Redes de Comunicaciones y Cómputo

Requerimientos

Asistencia y participación:

El curso incluye prácticas de programación, y tiene una carga horaria semanal de sólo 10 horas. Por eso les recomendamos:

- Leer materiales teóricos o introductorios antes de llegar a la clase.
- Presencia en la clase.
- Dedicar suficiente tiempo a resolver ejercicios durante la semana.

Examen

- Se planean dos evaluaciones en forma de parciales multiple choice, utilizando el sistema del Campus. Las fechas de estas evaluaciones se establecerán al inicio del cuatrimestre para evitar inconvenientes de programación.
 - TBD: Se explorará la posibilidad de incluir preguntas que requieran análisis conceptual y resolución manual en papel.
- La evaluación también puede involucrar, aunque en menor prioridad, la asistencia, participación en clase, realización de trabajos prácticos y proyectos.

Nota

La aprobación del curso se da por una ponderación de los siguientes conceptos:

- **Trabajos Prácticos**
- **Exámenes teóricos**

Materiales

- Los materiales del curso estarán disponibles en el repositorio del curso o en la página de la materia, excepto en casos de conflicto por copyright. Se recomienda que las lecturas para cada clase se realicen por adelantado.

Temario del Curso

- Basado en el libro "Algorithm Design" por Jon Kleinberg y Éva Tardos, el temario incluirá:
 - Análisis básico de algoritmos
 - Grafos
 - Divide & Conquer
 - Greedy
 - Programación Dinámica
 - Flujo en Redes

Guías de Ejercicios

Los estudiantes tendrán disponibles guías de ejercicios diseñadas para ilustrar, reforzar y evaluar los conceptos y casos discutidos, o que se discutirán en clase. Se recomienda que los estudiantes realicen el análisis y la discusión de los ejercicios en pequeños grupos.

Las prácticas son las mismas que se usaron en Algo 3, con la guía de 'Divide and Conquer' de Algo 2, y sin la práctica de complejidad computacional:

- 0 Introducción y Repaso** [a definir]
- 1 Técnicas Algorítmicas**
- 2 Divide y Venceras**
- 3 Grafos y Algoritmos**
- 4 Recorridos y Árboles**
- 5 Camino mínimo**
- 6 Flujo en Redes**

Trabajos Prácticos (TPs)

- Se propone la realización de cuatro TPs individuales, cada uno centrado en una técnica algorítmica específica. Estos TPs serán evaluados utilizando un sistema de juez automático, con posibilidad de revisión del código para asegurar la coherencia con las soluciones esperadas.
 - **Entrega de TPs:** Se permitirá la entrega de los TPs hasta el final del cuatrimestre, con el objetivo de gestionar de manera eficiente las consultas y revisiones.

Lenguaje de Programación

- Los estudiantes tendrán la libertad de elegir el lenguaje de programación para realizar sus TPs, sujeto a la familiaridad de los instructores con el lenguaje escogido y la factibilidad de evaluación de las soluciones propuestas.

Libros

1. **"Diseño de Algoritmos y su Análisis" por Eva Tardos y Jon Kleinberg:** Una obra que combina la teoría y la práctica del diseño de algoritmos, y que acompaña el programa de la materia.
2. **"Introducción a los Algoritmos" por Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest y Clifford Stein:** Este libro es una referencia esencial que abarca un amplio rango de algoritmos en profundidad, presentando técnicas de diseño y análisis de algoritmos de manera rigurosa y accesible. Recomendado para mayor desarrollo y profundidad del contenido.

Cronograma de Clases

El siguiente cronograma es solo orientativo y los docentes decidirán los temas específicos o ejercicios a trabajar según su criterio y adaptándose al desarrollo de la cursada. Los ejercicios indicados en cada práctica son solo sugerencias, y los docentes a cargo pueden decidir qué ejercicios tratar en cada clase independientemente de que estén o no listados en el cronograma.

Semana 1: Introduccion. Backtracking.

- **Lunes 18/3:** Repaso de Complejidad Computacional, introducción a los algoritmos de fuerza bruta y backtracking.
- **Miércoles 20/3:** Práctica de Backtracking. Ejercicios seleccionados entre: conjunto de soluciones válidas, dibujo de árboles de backtracking, implementación recursiva de algoritmos de suma de subconjuntos.

Semana 2: Programación Dinámica

- **Lunes 25/3:** Fundamentos de Programación Dinámica. Diferencias entre enfoques top-down y bottom-up. Aplicaciones y ejemplos.
- **Miércoles 27/3:** Práctica de Programación Dinámica: Top Down. Ejercicios seleccionados entre: optimización de la función de suma de subconjuntos con memoización, generación de cuadrados mágicos, cálculo de matrices simétricas para maximización de sumas.

Semana 3: Avanzando en Programación Dinámica

- **Lunes 1/4:** FERIADO: Con fines turísticos.
- **Miércoles 3/4:** Programación Dinámica: Reconstrucción de soluciones y enfoque Bottom-Up. Ejercicios seleccionados entre: reconstrucción de soluciones para suma de subconjuntos, algoritmos bottom-up para suma de subconjuntos, estrategias de backtracking para problemas de selección de actividades.

Semana 4: Divide & Conquer

- **Lunes 8/4:** Introducción a la técnica de Divide & Conquer. Conceptos teóricos, ejemplos y desarrollo de optimizaciones para alcanzar una mayor eficiencia.
- **Miércoles 10/4:** Práctica de Divide & Conquer. Ejercicios seleccionados entre: arreglo "más a la izquierda", búsqueda binaria para encontrar una posición i tal que $a_i = i$, aplicación del teorema maestro en distintos casos, cálculo de $A^1 + A^2 + \dots + A^n$ para matrices cuadradas.

Semana 5: Algoritmos Greedy

- **Lunes 15/4:** Fundamentos de los algoritmos Greedy. Exploración de estrategias para alcanzar soluciones óptimas locales que conducen a una solución global óptima.

- **Miércoles 17/4:** Algoritmos Greedy. Ejercicios seleccionados entre: encontrar el i -ésimo elemento en la mezcla ordenada de dos arreglos, minimización de la diferencia absoluta entre valores de dos arreglos ordenados, búsqueda de un elemento en un arreglo con una función de aparición compleja, y determinación de la posición única en una matriz booleana con restricciones.

Semana 6: Introducción a los Grafos y Representación

- **Lunes 22/4:** Introducción a los Grafos Parte I. Conceptos básicos de grafos, tipos de grafos, y propiedades iniciales.
- **Miércoles 24/4:** (festividad judía: inicio Pesaj) Grafos: Representación. Ejercicios seleccionados entre: demostración de igualdad de sumas de grados de entrada y salida en digrafos, y existencia de grafos orientados con grados de salida distintos.

Semana 7: Continuación de Grafos

- **Lunes 29/4:** (festividad judía: fin de Pesaj) Introducción a los Grafos Parte II. Profundización en propiedades y teoremas fundamentales de grafos.
- **Miércoles 1/5:** FERIADO: Día del Trabajador.

Semana 8: Recorridos en Grafos

- **Lunes 6/5: (PRACTICA)** Grafos: Recorridos. Estudio de recorridos DFS y BFS, aplicaciones y propiedades derivadas de estos recorridos.
- **Miércoles 8/5:** Consultas y práctica integradora. Discusión sobre ejercicios obligatorios y opcionales, resolución de dudas y consultas específicas.

Semana 9: Primer Parcial

- **Lunes 13/5:** Consultas y práctica integradora. Discusión sobre ejercicios obligatorios y opcionales, resolución de dudas y consultas específicas.
- **Miércoles 15/5:** PRIMER PARCIAL

Semana 10: Árbol Generador Mínimo

- **Lunes 20/5:** Teoría de Árboles Generadores Mínimos. Se discutirán los conceptos fundamentales, propiedades y la importancia de los árboles generadores mínimos en la teoría de grafos y aplicaciones prácticas.
- **Miércoles 22/5:** Árbol Generador Mínimo: Algoritmos de Prim y Kruskal + Modelado. Ejercicios seleccionados entre: diseño de un algoritmo de tiempo $O(n+m)$ para encontrar el árbol generador mínimo v -geodésico en grafos con pesos, análisis de propiedades para determinar si un grafo es un cactus y diseño de algoritmos para encontrar árboles generadores mínimos en grafos cactus, y aplicación de los algoritmos de Prim y Kruskal para modelar y resolver problemas específicos.

Semana 11: Camino Mínimo Parte I

- **Lunes 27/5:** Introducción a los problemas de Camino Mínimo. Conceptos básicos, formulación de problemas y ejemplos introductorios.
- **Miércoles 29/5:** Camino Mínimo: Algoritmos de Dijkstra y Bellman-Ford. Ejercicios seleccionados entre: diseño de un algoritmo eficiente que encuentre caminos mínimos evitando aristas específicas, modelado y solución de problemas de tráfico con peajes inversos usando algoritmos de camino mínimo, y algoritmos para determinar la existencia de ciclos beneficiosos en configuraciones de peajes.

Semana 12: Camino Mínimo Parte II

- **Lunes 3/6:** Continuación de Camino Mínimo: Algoritmos de Floyd-Warshall y tratamiento especial de DAGs para caminos mínimos. Profundización en propiedades y técnicas específicas.
- **Miércoles 5/6:** Camino Mínimo: Aplicaciones integradoras y casos especiales. Ejercicios seleccionados entre: diseño de algoritmos de programación dinámica top-down para problemas de camino mínimo en DAGs, modelado del problema del vuelto y gestión de proyectos como problemas de camino mínimo, y selección de algoritmos de camino mínimo adecuados para grafos densos y ralos en distintos contextos.

Semana 13: Camino Mínimo Parte III + Introducción al Flujo

- **Lunes 10/6:** Conclusión sobre Caminos Mínimos y transición hacia el concepto de Flujo en Redes. Exploración inicial de las propiedades de los flujos y presentación de problemas básicos de flujo.
- **Miércoles 12/6:** Flujo: Conceptos Generales. Introducción detallada a los problemas de flujo en redes, incluyendo la definición de redes de flujo, flujo máximo, y ejercicios introductorios sobre propiedades de los flujos.

Semana 14: Feriado y Teoría de Flujo Parte II

- **Lunes 17/6:** FERIADO: Paso a la Inmortalidad del General Martín Miguel de Güemes.
- **Miércoles 19/6 (TEORICA):** Flujo Parte II. Clase teórica enfocada en la profundización de conceptos de flujo, incluyendo algoritmos para encontrar flujos máximos y aplicaciones teóricas.

Semana 15: Flujo: Aplicaciones Prácticas

- **Lunes 24/6 (PRACTICA):** Flujo: Problemas de Asignación y Transporte. Ejercicios seleccionados entre: modelado de problemas de asignación de solteros a mesas en bodas como problemas de flujo, modelado de problemas de asignación de recursos como problemas de flujo, y determinación de la complejidad de resolver estos modelos con el algoritmo de Edmonds y Karp.

- **Miércoles 26/6:** Consultas. Sesión dedicada a resolver dudas, consultas sobre ejercicios y preparación para exámenes.

Semana 16: Segundo Parcial

- **Lunes 1/7:** Consultas y práctica integradora. Discusión sobre ejercicios obligatorios y opcionales, resolución de dudas y consultas específicas.
- **Miércoles 3/7:** SEGUNDO PARCIAL

Recuperatorios

- **Miércoles 10/7:** Consultas primer recuperatorio.
- **Miércoles 17/7:** PRIMER RECUPERATORIO
- **Miércoles 24/7:** Consultas segundo recuperatorio.
- **Miércoles 31/7:** SEGUNDO RECUPERATORIO