

INVESTIGACION DE OPERACIONES

IDENTIFICACIÓN DE LA UNIDAD DE APRENDIZAJE

Unidad académica: Facultad de Contaduría Administración e Informática							
Programa educativo: Maestría en Optimización y Cómputo Aplicado							
Unidad de aprendizaje: Investigaciones de Operaciones				Ciclo de formación: Especializado			
				Eje de formación: Básico			
				Semestre: 1 al 4			
Elaborado por: Dr. José Crispín Zavala Díaz				Fecha de elaboración: 1 diciembre 2022			
Actualizado por: M en E. Ana Linda Pineda Mendez y Dr. José Crispín Zavala Díaz				Fecha de revisión y actualización: 1 diciembre 2022			
Clave:	Horas teóricas:	Horas prácticas:	Horas totales:	Créditos:	Tipo de unidad de aprendizaje:	Carácter de la unidad de aprendizaje	Modalidad
No Aplica	2	2	4	6	Optativa	Posgrado	presencial
Programa (s) educativo (s) en los que se imparte: Maestría en Optimización y Cómputo Aplicado							

PRESENTACIÓN

En este curso se van a estudiar los tipos básicos de algoritmos no vistos en el curso de Algoritmia. Estos son los algoritmos de programación dinámica, los métodos de ramas y cortes y los algoritmos para los problemas de programación lineal. Los problemas considerados tienen aplicaciones importantes en la vida real. Se define el problema fundamental de programación lineal (PL). Muchos problemas de vida real se pueden representar como un problema de programación lineal entera. Se verá el método fundamental para resolver el problema PL, es decir, método Simplex.

Los métodos de ramas y cortes al igual que el método Simplex eliminan los subconjuntos de las soluciones no-óptimas. Los algoritmos de ramas y cortes son algoritmos de búsqueda implícita que permite excluir subconjuntos de las soluciones factibles sin perder la solución óptima. Esto se efectúa aplicando los límites superiores e inferiores con la regla del corte. Estos algoritmos se aplican sólo a los problemas intratables.

Los algoritmos de programación dinámica aprovechan la estructura recursiva del problema para eliminar las soluciones que no pueden ser óptimas. Estos algoritmos pueden ser polinomiales o exponenciales para los problemas intratables. El problema de la ruta más corta es recursivo y puede ser resuelto con programación lineal. En el curso de Algoritmia 1 se estudió el algoritmo de Dijkstra para las gráficas cuyos pesos asociados a las aristas son no-negativos. En este curso se verán los casos más generales y el método de aproximación sucesiva de Bellman-Ford para estos casos.

También se verá el algoritmo para encontrar caminos más cortos entre todos los pares de vértices.

Requisitos mínimos: Algoritmia.

PROPÓSITOS

El propósito del curso es dotar al alumno de las herramientas necesarias para que sea capaz, en base a un estudio minucioso del análisis y al diseño de algoritmos para un problema dado, de decidir cuál es el mejor algoritmo y la mejor estructura de datos para solucionar el problema

COMPETENCIAS QUE CONTRIBUYEN AL PERFIL DE EGRESO

Competencias básicas	
a. Lectura, análisis y síntesis b. Comunicación oral y escrita c. Aprendizaje estratégico d. Razonamiento lógico-matemático e. Razonamiento científico	
Competencias genéricas	
Competencias genéricas: a. Cognitivas-metacognitivas • Resolución de problemas • Pensamiento crítico • Creatividad b. Socioemocionales genéricas • Orientación al logro • Apertura a la experiencia c. Digitales genéricas • Búsqueda, valoración y gestión de información • Comunicación y colaboración en línea • Resolución de problemas técnicos d. Socioculturales genéricas • Comunicación en segundo idioma • Responsabilidad social y ciudadana	
Competencias laborales	
Competencias específicas disciplinares	
a) Conocer, comprender y aplicar conceptos básicos de matemática discreta, teorías, análisis y diseño de algoritmos y métodos estadísticos, mediante el estudio de modelos teóricos de la computación para generar soluciones de cómputo e informáticas b) Plantea el modelo matemático de problemas de optimización y cómputo aplicado para economizar recursos en el desarrollo sostenible y sustentable mediante la aplicación del método científico. c) Analiza metodologías para producir nuevo conocimiento a través de proyectos de investigación	
Competencias transferibles para el trabajo	
Digitales para el trabajo • Aplica herramientas de análisis de datos para generar información que auxilie en la toma de decisiones utilizando repositorios de datos públicos y/o privados. Competencias para el trabajo transdisciplinar • Desarrolla investigación en equipo, a fin de identificar puentes de interconexión entre disciplinas que aporten nuevo conocimiento a través de discusiones y búsqueda de información compartida Competencias para el aprendizaje a lo largo de la vida laboral (aprender, reaprender y desaprender)	

- Desarrolla conocimientos, competencias, actitudes y valores necesarios para tener la capacidad de adaptación a nuevas demandas profesionales que surjan en el mercado laboral, mediante un aprendizaje flexible personalizado por cada estudiante definido por el asesor de tesis con apoyo del paradigma de la educación 4.0

CONTENIDOS

Bloques	Temas
Módulo 1: Programación lineal y entera.	1.1. Formas de problemas de programación lineal. Óptimos locales y globales, funciones convexas y conjuntos convexas. 1.2. Método Simplex (soluciones básicas factibles, cambiar de una solución básica a otra, organización de tabla), aspectos geométricos del algoritmo Simplex 1.3. Dualidad 1.4. Algoritmo “primal-dual” para PL, el problema de transporte 1.5. Simplex no es un algoritmo polinomial. El algoritmo polinomial para PL 1.6. Cascos convexas, poliedros y dimensión, politopos y facetas 1.7. Problemas de optimización formuladas como PE, límites superiores 1.8. Poliedros enteros. El algoritmo de Planos de Corte. 1.9. Flujos en redes. Flujos con el costo máximo y el costo mínimo. Interpretación LP (dos semanas) 1.10. Problemas de emparejamientos (“matching”). Emparejamientos perfectos con el peso mínimo.
Módulo 2: Programación dinámica.	2.1. Definición del método de programación dinámica, el principio de la optimalidad. 2.2. El problema de la ruta crítica, las ecuaciones recurrentes de Bellman. 2.3. Rutas más cortas en gráficas no-cíclicas. 2.4. El método de aproximación sucesiva (Bellman-Ford). 2.5. Interpretación del problema de programación lineal. 2.6. Caminos más cortos entre todos los pares de vértices. 2.7. El algoritmo de programación dinámica para el problema de la mochila entera. 2.8. El algoritmo de programación dinámica para el problema de suma de subconjuntos. 2.9. El algoritmo de programación dinámica para el problema de viajero. 2.10. El algoritmo de programación dinámica para el problema de planeación de trabajos a plazos con los tiempos de liberación.
Módulo 3: Los métodos de Ramas y Cortes - Branch-and-Bound.	3.1. Representación del espacio de las soluciones con un árbol de soluciones. 3.2. El concepto de vuelta atrás (backtracking). 3.3. Definición de los límites superiores e inferiores. 3.3. Definición de las reglas de ramificación y del corte. 3.5. El algoritmo de Ramas y Cortes para el problema de viajero.

	3.6. El algoritmo de Ramas y Cortes para el problema de la mochila. 3.7. El algoritmo de Ramas y Cortes para el problema general de la calendarización multi- etapas (job-shop scheduling).
--	--

ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE

Estrategias de aprendizaje sugeridas (Marque con X)			
Aprendizaje basado en problemas	☒	Nemotecnia	☐
Estudios de caso	☒	Análisis de textos	☒
Trabajo colaborativo	☒	Seminarios	☐
Plenaria	☐	Debate	☐
Ensayo	☐	Taller	☐
Mapas conceptuales	☐	Ponencia científica	☐
Diseño de proyectos	☒	Elaboración de síntesis	☒
Mapa mental	☐	Monografía	☐
Práctica reflexiva	☐	Reporte de lectura	☒
Trípticos	☐	Explosión oral	☐
Otros			
Estrategias de enseñanza sugeridas (Marque X)			
Presentación oral (conferencia o exposición) por parte del docente	☒	Experimentación (prácticas)	☒
Debate o Panel	☐	Trabajos de investigación documental	☐
Lectura comentada	☐	Anteproyectos de investigación	☐
Seminario de investigación	☐	Discusión guiada	☐
Estudio de Casos	☒	Organizadores gráficos (Diagramas, etc.)	☐
Foro	☐	Actividad focal	☐
Demostraciones	☐	Analogías	☐
Ejercicios prácticos (series de problemas)	☒	Métodos de proyectos	☒
Interacción con la realidad (a través de videos, fotografías, dibujos software especialmente diseñado)	☒	Exploración de la web	☒
Archivo	☒	Portafolio de evidencias	☐
Ambiente virtual (foros, chat, correos, ligas a otros sitios web, otros)	☐	Enunciado de objetivo o intenciones	☐
Otra, especifique (lluvia de ideas, mesa redonda, textos programados, cine, teatro, juego de roles, experiencia estructurada, diario reflexivo, entre otras): mesa redonda			

CRITERIOS DE EVALUACIÓN

Criterios	Porcentaje
- Examen por módulo Nota: Se llevará a cabo tres exámenes parciales con un valor de 26.6%, cada uno.	80%

- Examen o proyecto final	20%
Total	100%

PERFIL DEL PROFESOR

Doctor en matemáticas o en computación teórica, activo en investigación, preferentemente que haya impartido esta asignatura o similares al menos en 4 ocasiones.

REFERENCIAS BIBLIOGRÁFICAS

Básicas:

- Aho, A., Hopcroft, J. and Ullman, J. The design and analysis of computer algorithms. Addison- Wesley, 1974.
- Horowitz, E. and Sahni, S. Fundamentals of computer algorithms. Computer Science Press 1978.
- Garey, M. and Johnson, D. Computers and intractability: A guide to theory of NP-completeness. Freeman and Company, 1979.
- Aho, A., Hopcroft, J. and Ullman, J. Estructuras de datos y algoritmos. Addison-Wesley, 1988.
- Lawler, E. Combinatorial optimization: Networks and Matroids. Dover Publications, 1976.
- Cook, W., Cunningham, W., Pulleyblank, W. and Schrijver, A. Combinatorial optimization. Wiley, 1998.
- Papadimitriou, C. and Steiglitz, K. Combinatorial optimization: algorithms and complexity. Dover Publications, 1998.
- Mehlhorn, K. and Sanders, P. Algorithms and Data Structures. The Basic Toolbox. Springer, 2007.
- Dasgupta, S., Papadimitriou, C. and Vazirani, U. Algorithms. McGraw-Hill, 2006.
- Levitin, A. Introduction to the Design & Analysis of Algorithms. Addison Wesley 2 edition, 2006.
- Mehta, D. P. (Series Editor: Sahni, S.). Handbook of Data Structures and Applications, 2004.
- Kocay, W. Graphs, Algorithms, and Optimization, 2004.
- Dobrushkin, V. A. Methods in Algorithmic Analysis, 2009.

Complementarias:-----

Web:-----

Otros: -----