

COMPLEJIDAD DE LOS ALGORITMOS

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: Complejidad de los algoritmos				Ciclo de formación: Especializado			
				Eje de formación: Básico			
				Semestre: 1 al 4			
Elaborado por: Dr. Nodari Vahkania				Fecha de elaboración: 1 agosto 2016			
Actualizado por: M en E. Ana Linda Pineda Mendez y Dr. Nodari Vahkania				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

Este curso desarrolla el temario de los cursos anteriores Algoritmia y Teoría de la Computación. En la base del conocimiento del análisis y desarrollo de los algoritmos y modelos teóricos de computación, se construye teoría de complejidad computacional de los algoritmos computacionales. Se darán las definiciones formales de los algoritmos y problemas tratables polinomiales e intratables NP-completos y los métodos del establecimiento de complejidad computacional de un problema o algoritmo dado.

Saber la complejidad computacional de un algoritmo, es decir, tiempo que va a necesitar el algoritmo para resolver un problema dado, es drásticamente importante. Un algoritmo on-line que da la solución en tiempo real no necesita este tipo de estimación. Desafortunadamente, para la mayoría de los problemas reales, los algoritmos de este tipo no dan soluciones satisfactorias en términos de aproximación al valor óptimo. Entonces, cuando la calidad de las soluciones es importante, tenemos que desarrollar/aplicar un algoritmo off-line, que puede tardar de unos segundos a unos siglos hasta miles de siglos en las computadoras de hoy. La estimación de complejidad temporal de un algoritmo nos da el tiempo que la computadora tardará en correr este algoritmo para un problema dado. Esta estimación nos facilitara la selección de un algoritmo apropiado para un problema dado (según los requerimientos prácticos dados).

Requisitos mínimos: Algoritmia y Teoría de la Computación.

PROPÓSITOS

El propósito del curso es dotar al alumno de las herramientas necesarias para que sea capaz de estimar la complejidad computacional de un problema (algoritmo) y desarrollar el algoritmo mas adecuado para este problema, en base de esta estimación.

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: La definición formal de complejidad computacional.	1.1. Problemas polinomiales e intratables. 1.2. Problemas de decisión y los esquemas de codificación de los problemas (la longitud de esta codificación). 1.3. Máquinas de Turing deterministas: la definición de la clase P. 1.4. Máquinas de Turing no-deterministas: la definición de la clase NP. 1.5. Complejidad temporal y espacial.
Módulo 2: Enseñanza de NP-completes.	2.1 Transformaciones polinomiales y NP-completes. El primer problema NP-completo: teorema de Cook. 2.2 Las técnicas básicas de enseñanza de NP-completes de un problema usando transformación polinomial. 2.3 Los seis problemas básicos NP-completes (satisfacibilidad, emparejamiento, dimensiones, cubierta de vértices, clique, circuito Hamiltoniano y partición). 2.4 Problemas NP-completes en el sentido rígido (unario) y débil (binario). 2.5 Problemas y algoritmos pseudo-polinomiales. 2.6 Problemas de optimización NP-duros.
Módulo 3: Algoritmos aproximados para problemas NP-duros y NP-completes.	3.1 Aproximación relativa y absoluta. 3.2 Esquemas aproximados polinomiales, esquemas aproximados completamente polinomiales. 3.3 Algoritmos aproximados polinomiales probabilísticos, estimaciones de complejidad de tiempo probabilísticas. 3.4 Heurísticas y meta-heurísticas. Búsqueda local. Recocido simulado.

ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE

Estrategias de aprendizaje sugeridas (Marque con X)			
Aprendizaje basado en problemas	☒	Nemotecnica	☐
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 ideos, 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.

Complementarias:-----

Web:-----

Otros: -----