



5.4 ÁREA DE PROFUNDIZACIÓN II: CIENCIAS DE LA COMPUTACIÓN Y MATEMÁTICAS

 <p>UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO Escuela Nacional de Estudios Superiores Unidad Juriquilla</p> <p>Plan de Estudios de la Licenciatura en Tecnología Sistema Escolarizado: Modalidad Presencial</p>	 <p>ENES JURIQUILLA</p>		
Programa			
Diseño y Análisis de Algoritmos			
Clave	Semestre	Créditos	Duración
	5	10	16 semanas
			Eje de formación
			Profundización
			Área de profundización
			Ciencias de la Computación y Matemáticas
			Etapa de formación
			Intermedia
Modalidad	Curso (X) Taller (X) Lab () Sem ()		Tipo
			T () P () T/P (X)
Carácter	Obligatorio ()	Optativo ()	Horas
	Obligatorio E (X)	Optativo E ()	
			Semana
			Semestre
			Teóricas 4
			Teóricas 64
			Prácticas 2
			Prácticas 32
			Total 6
			Total 96
Seriación			
Ninguna (X)			
Obligatoria ()			
Asignatura antecedente			
Asignatura subsecuente			
Indicativa ()			
Asignatura antecedente			
Asignatura subsecuente			
Objetivos generales:			

Al finalizar el curso el alumnado será capaz de diseñar, analizar y evaluar algoritmos empleando la notación asintótica. Además, el alumnado comprenderá y aplicará los fundamentos de estructuras de datos avanzadas para la resolución de problemas complejos.

Objetivos específicos:

1. El alumnado comprenderá el concepto de complejidad computacional y lo aplicará en el análisis de algoritmos.
2. El alumnado estudiará algoritmos de ordenamiento eficientes y los aplicará para la resolución de problemas en el ámbito de computación.
3. El alumnado abordará el uso de estructuras de datos avanzadas para optimizar el acceso a memoria y reducir tiempo de acceso y búsquedas.
4. El alumnado estudiará nuevas técnicas de análisis de algoritmos que le permitirá abordar problemas y plantear soluciones utilizando esquemas tales como programación dinámica y algoritmos voraces y análisis de complejidad algorítmica amortizada.
5. El alumnado desarrollará habilidades en el área de análisis de grafos y las aplicará para la solución de problemas de distintas disciplinas.

Índice temático

	Tema	Horas Semestre	
		Teóricas	Prácticas
1	Fundamentos y notación asintótica	10	0
2	Algoritmos de ordenamiento	10	8
3	Estructuras de datos avanzadas	16	8
4	Técnicas de análisis y diseño avanzado de algoritmos	16	8
5	Análisis de grafos	12	8
Subtotal		64	32
Total		96	

Contenido Temático

Tema	Subtemas
1	Fundamentos y notación asintótica 1.1 Notación asintótica 1.1.1 Análisis de algoritmos 1.1.2 Notaciones estándar y funciones 1.2 Recurrencias 1.2.1 El método de sustitución 1.2.2 Recursión de árbol 1.2.3 El método maestro 1.3 Análisis probabilístico y algoritmos aleatorios 1.3.1 Variables aleatorias indicadoras 1.3.2 Algoritmos aleatorios 1.3.3 Análisis probabilístico
2	Algoritmos de ordenamiento 2.1 Heapsort

	<ul style="list-style-type: none"> 2.1.1 Construcción de pilas y propiedades 2.1.2 Algoritmo Heapsort 2.1.3 Colas de prioridad 2.2 Quicksort <ul style="list-style-type: none"> 2.2.1 Descripción del algoritmo 2.2.2 Rendimiento del algoritmo 2.2.3 Versión aleatoria de quicksort 2.2.4 Análisis 2.3 Algoritmos de ordenamiento en tiempo lineal <ul style="list-style-type: none"> 2.3.1 Límites inferiores de ordenamiento 2.3.2 Counting sort 2.3.3 Radix sort 2.3.4 Bucket sort
3	<p>Estructuras de datos avanzadas</p> <ul style="list-style-type: none"> 3.1 Estructuras elementales <ul style="list-style-type: none"> 3.1.1 Implementación utilizando apuntadores y objetos 3.1.2 Representación de árboles 3.2 Tablas hash <ul style="list-style-type: none"> 3.2.1 Tablas de direcciones 3.2.2 Funciones hash 3.2.3 Direccionamiento abierto 3.2.4 Hash perfecto 3.3 Árboles binarios de búsqueda <ul style="list-style-type: none"> 3.3.1 Definiciones y conceptos básicos 3.3.2 Consultas 3.3.4 Inserciones y eliminaciones 3.3.5 Construcción aleatoria 3.3.6 Árboles red-black
4	<p>Técnicas de análisis y diseño avanzado de algoritmos</p> <ul style="list-style-type: none"> 4.1 Programación dinámica <ul style="list-style-type: none"> 4.1.1 Elementos de programación dinámica 4.1.2 Problema de subcadena común más grande 4.1.3 Uso de árboles binarios de búsqueda 4.2 Algoritmos voraces <ul style="list-style-type: none"> 4.2.1 Elementos de algoritmos voraces 4.2.2 Códigos Huffman 4.2.3 Fundamentos teóricos 4.2.4 Planificación de tareas 4.3 Análisis amortizado <ul style="list-style-type: none"> 4.3.1 Análisis agregado 4.3.2 Método de conteo



	4.3.3 Método de potencial 4.3.4 Tablas dinámicas
5	Análisis de grafos 5.1 Algoritmos elementales de grafos 5.1.1 Representación de grafos 5.1.2 Búsqueda en anchura 5.1.3 Búsqueda en profundidad 5.1.4 Ordenamiento topológico 5.1.5 Componentes fuertemente conectadas 5.2 Árboles de expansión mínimos 5.2.1 Fundamentos y expansión de Árboles 5.2.2 Algoritmo de Kruskal y Prim 5.3 Camino más corto (fuente única) 5.3.1 Algoritmo Bellman-Ford 5.3.2 Camino más corto en grafos acíclicos y dirigidos 5.3.3 Algoritmo de Dijkstra
Estrategias didácticas	
Exposición	(X)
Trabajo en equipo	(X)
Lecturas	(X)
Trabajo de investigación	(X)
Prácticas (taller o laboratorio)	()
Prácticas de campo	()
Aprendizaje por proyectos	(X)
Aprendizaje basado en problemas	(X)
Casos de enseñanza	(X)
Otras (especificar)	()
Evaluación del aprendizaje	
Exámenes parciales	(X)
Examen final	(X)
Trabajos y tareas	(X)
Presentación de tema	()
Participación en clases	(X)
Asistencia	(X)
Rúbricas	()
Portafolios	()
Listas de cotejo	()
Otras (especificar)	()
Código de conducta	
<p>La conducta del profesorado y alumnado del curso será acorde con los principios y valores especificados en el Código de Ética de la Universidad Nacional Autónoma de México aprobado el 1 de julio del 2015 por el Consejo Universitario, en especial en lo referente a la integridad y honestidad académica. "La integridad y la honestidad académica implican: Citar las fuentes de ideas, textos, imágenes, gráficos u obras artísticas que se empleen en el trabajo universitario, y no sustraer o tomar la información generada por otros o por sí mismo sin señalar la cita correspondiente u obtener su consentimiento y acuerdo. No falsificar, alterar, manipular, fabricar, inventar o fingir la autenticidad de datos, resultados, imágenes o información en los trabajos académicos, proyectos de investigación, exámenes, ensayos, informes, reportes, tesis, audiencias, procedimientos de orden disciplinario o en cualquier documento inherente a la vida académica universitaria" (Gaceta UNAM, 30 de julio 2015).</p>	
Perfil Profesiográfico	
Título o Grado	Deberá contar con licenciatura o posgrado en ciencias o ingeniería de la computación o bien alguna otra afín del área de las Ciencias Físico Matemáticas y las Ingenierías.

Experiencia docente	Con experiencia docente en licenciatura y/o en posgrado, preferentemente de tres años impartiendo la asignatura u otra relacionada en el nivel superior.
Otra característica	El curso-taller deberá ser impartido por académicas/os de tiempo completo de la UNAM, o equivalente, que estén activos en investigación sobre temas directamente relacionados con el curso-taller.
Bibliografía básica <ul style="list-style-type: none"> • Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to algorithms. MIT press. • Sandeep, S. & Amit, K. (2019). Design and Analysis of Algorithms: A contemporary Perspective. Cambridge University Press. • Vrajitoru, D. & Knight, W. (2014). Practical Analysis of Algorithms. Springer. 	
Bibliografía complementaria <ul style="list-style-type: none"> • Vermani, L. R. & Vermani, S. (2019). An Elementary Approach To Design and Analysis of Algorithms. Word Scientific. 	