

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO****Licenciatura en Ciencias de la Computación****Facultad de Ciencias**

Programa de la asignatura

**Denominación de la asignatura:*****Complejidad Computacional***

Clave:	Semestre: 7	Eje temático: Computación Teórica	No. Créditos: 10
Carácter: Obligatoria	Horas		Horas por semana
Tipo: Teórico- Práctica	Teoría: 4	Práctica: 2	Total de Horas 96
Modalidad: Curso	Duración del programa: Semestral		

Asignatura con seriación obligatoria antecedente: Probabilidad I; Autómatas y Lenguajes Formales; Lógica Computacional

Asignatura con seriación obligatoria subsecuente: Ninguna

Asignatura con seriación indicativa antecedente: Análisis de Algoritmos

Asignatura con seriación indicativa subsecuente: Ninguna

Objetivo general:

Comprender y aplicar la Complejidad Computacional, desde la teoría básica de NP-Completez, hasta temas más avanzados, contribuyendo así a profundizar la formación del alumno en computación teórica y dotándolo de herramientas y conocimientos que le serán de utilidad tanto para otras materias teóricas como para el resto de su formación.

Índice temático

Unidad	Temas	Horas	
		Teóricas	Prácticas
I	Introducción y conceptos básicos	8	3
II	La teoría de NP-Completez	16	9
III	Demostraciones de problemas NP-Completos	24	12
IV	Temas selectos	16	8
Total de horas:		64	32
Suma total de horas:		96	

Contenido temático	
Unidad	Tema
I Introducción y conceptos básicos	
I.1	Motivación para estudiar complejidad computacional.
I.2	Problemas, algoritmos y complejidad.
I.3	Notación asintótica, codificación y modelos de cómputo.
II La teoría de NP-Completez	
II.1	Máquinas de Turing y la clase P.
II.2	La clase NP.
II.3	Relación P-NP y transformaciones polinomiales.
II.4	Definición de NP-Completez.
II.5	Teorema de Cook.
II.6	Jerarquía de complejidad.
III Demostraciones de problemas NP-completos	
III.1	Problemas básicos: 3SAT, apareamientos, cubierta de vértices, circuito hamiltoniano, clan y partición.
III.2	Técnicas: restricción, remplazo local y diseño de componentes.
IV Temas selectos	
IV.1	Problemas no computables y máquinas de Turing universales.
IV.2	Utilizando NP-Completez para analizar problemas.
IV.3	Enfrentando problemas NP-Completos: aproximación, heurísticas, etc.
IV.4	Otras clases de complejidad.
IV.5	Otros modelos de cómputo: DNA, paralelos, distribuidos, cuánticos, etc.

Bibliografía básica:

1. Michael Garey y David Johnson, *Computers and Intractability - a guide to the theory of NP-Completeness*. PWS Publishing Company, 1997, 2nd Ed. 1995.
2. Michael Sipser, *Introduction to the Theory of Computation*. PWS Publishing Company, 1997. 2nd Ed 2005.
3. Sanjeev Arora y Boaz Barak, *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.
4. Dexter C. Kozen, *Theory of Computation*. Springer, 2006.

Bibliografía complementaria:

1. John E. Hopcroft, Rajeev Motwani y Jeffrey D. Ullman, *Introduction to Automata Theory, Languages, and Computation*. Addison Wesley, 2006 (3rd edition).
2. Christos H. Papadimitriou, *Computational Complexity*. Addison Wesley, 1993.
3. Oded Goldreich, *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, 2008.
4. Dexter C. Kozen, *Automata and Computability*. Springer, 1997.

Sugerencias didácticas:		Métodos de evaluación:	
Exposición oral	(X)	Exámenes parciales	(X)
Exposición audiovisual	()	Examen final escrito	()
Ejercicios dentro de clase	(X)	Trabajos y tareas fuera del aula	(X)
Ejercicios fuera del aula	()	Exposición de seminarios por los alumnos	()
Seminarios	()	Participación en clase	(X)
Lecturas obligatorias	()	Asistencia	()
Trabajo de investigación	()	Seminario	()
Prácticas de taller o laboratorio	()	Otras: Proyectos de programación.	
Prácticas de campo	()		
Otras: _____			
Perfil profesiográfico:			
Egresado preferentemente de la Licenciatura en Ciencias de la Computación o Matemático con especialidad en Computación. Es conveniente que posea un posgrado en la disciplina. Con experiencia docente.			