



**UNIVERSIDAD AUTÓNOMA DE
CHIHUAHUA**

Clave:



Clave:

FACULTAD DE INGENIERÍA

COMPUTACIÓN

DES:	Ingeniería
Programa(s) Educativo(s):	Maestría en Ciencias Básicas y Aplicadas
Tipo de materia:	Obligatoria
Clave de la materia:	MCBAOP101
Semestre:	Primero
Área en plan de estudios:	Específica
Créditos	5
Total de horas por semana:	5
<i>Teoría:</i>	5
<i>Práctica</i>	
<i>Taller:</i>	
<i>Laboratorio:</i>	
<i>Prácticas complementarias:</i>	
<i>Trabajo extra clase:</i>	1
Total de horas semestre:	80
Fecha de actualización:	Febrero 2016
Materia requisito:	Ninguna

PROPÓSITO DEL CURSO

Que el alumno defina y analice el tipo de problemas que pueden o no pueden ser resueltos por una computadora bajo términos formales. Asocia el concepto de máquina abstracta y codificación de los problemas. Conoce y describe los problemas indecidibles. Simula el comportamiento de las máquinas abstractas básicas inmersas en la teoría de la computación. Manifiesta valores y actitudes, como el respeto a las ideas de otros, responsabilidad de su aprendizaje, disciplina en el aula, cooperación, criticidad y creatividad con carácter científico a través del trabajo en equipo. Comunica en forma oral y escrita sus ideas e interpretaciones, respecto a las máquinas abstractas estudiadas, así como expone juicios de valor respecto a la relación que estos guardan con su vida y el mundo que le rodea.

COMPETENCIAS (Tipo y nombre de la competencias que nutre la materia y a las que contribuye)	DOMINIOS COGNITIVOS (Objetos de estudio, temas y subtemas)	RESULTADOS DE APRENDIZAJE. (Por objeto de estudio).
<p>GENÉRICAS: Gestión del conocimiento: Demuestra conocimientos y habilidades para la búsqueda, análisis crítico, síntesis y procesamiento de información para su transformación conocimiento con actitud ética.</p> <p><i>Identifica y articula sus necesidades de conocimiento</i></p>	<p>I. LENGUAJES REGULARES Y AUTÓMATAS FINITOS</p> <p>1.1 Expresiones regulares. 1.2 Autómatas finitos deterministas. 1.3 Autómatas finitos no deterministas. 1.4 Variantes de los autómatas finitos. 1.5 Propiedades de los lenguajes regulares.</p>	<p>Analiza y compara los diferentes tipos de autómatas finitos.</p> <p>Construye modelos teóricos de autómatas finitos como representaciones de ciertos fenómenos.</p>

<p><i>a partir de definir problemas de información relevante.</i></p> <p>ESPECÍFICAS: Modelación y simulación matemática</p> <p>Modela sistemas dinámicos mediante simulaciones matemáticas para generar predicciones de comportamiento que contribuyen a la solución de problemas de contexto considerando distintos escenarios de forma honesta y responsable</p> <p><i>Clasifica problemas de sistemas complejos que pueden ser modelados matemáticamente.</i></p> <p><i>Plantea métodos matemáticos y computacionales de solución de forma responsable y ética.</i></p> <p><i>Determina soluciones particulares del sistema complejo en tiempos razonables.</i></p> <p><i>Deduce comportamientos del sistema complejo por medio de simulaciones computacionales y matemáticas de forma honesta y responsable.</i></p>		
	<p>II. LENGUAJES LIBRES DE CONTEXTO Y AUTÓMATAS DE PILA</p> <p>2.1 Gramáticas independientes del contexto. 2.2 Árboles de análisis y ambigüedad. 2.3 Autómatas de pila. 2.4 Propiedades de los lenguajes independientes del contexto.</p>	<p>Analiza y diseña gramáticas independientes del contexto para construir lenguajes.</p> <p>Construye modelos de autómatas de pila deterministas y no deterministas.</p> <p>Contrasta las diferencias entre los autómatas finitos y los autómatas de pila.</p>
	<p>III. DECIDIBILIDAD</p>	

	<p>3.1 Modelo básico de las máquinas de Turing.</p> <p>3.2 Problemas de decisión.</p> <p>3.3 Esquemas de codificación para los problemas.</p> <p>3.4 Problemas indecidibles.</p> <p>3.5 Teorema de Rice.</p>	<p>Asocia los modelos de máquinas de Turing con problemas codificados.</p> <p>Analiza problemas indecidibles.</p>
--	--	---

OBJETO DE ESTUDIO	METODOLOGÍA (Estrategias, secuencias, recursos didácticos)	EVIDENCIAS DE APRENDIZAJE.
I. Lenguajes regulares y autómatas finitos.	<ol style="list-style-type: none"> 1. Aprendizaje interactivo (exposición del profesor) 2. Investigación de tópicos. 3. Se divide el grupo para búsqueda y análisis de la información. 4. Demostraciones formales. 	<ul style="list-style-type: none"> ● Pseudo-códigos ● Problemas resueltos ● Demostraciones formales.
II. Lenguajes libres de contexto y autómatas de pila.	<ol style="list-style-type: none"> 1. Aprendizaje interactivo (exposición del profesor) 2. Investigación de tópicos. 3. Se divide el grupo para búsqueda y análisis de la información. 4. Demostraciones formales. 	<ul style="list-style-type: none"> ● Pseudo-códigos ● Problemas resueltos ● Demostraciones formales.
III. Decidibilidad.	<ol style="list-style-type: none"> 1. Aprendizaje interactivo (exposición del profesor) 2. Investigación de tópicos. 3. Se divide el grupo para búsqueda y análisis de la información. 4. Demostraciones formales. <p>Material de Apoyo didáctico:</p> <p>Recursos</p> <ul style="list-style-type: none"> ● Literatura citada en el programa del curso. ● Materiales gráficos: artículos y libros, entre otros 	<ul style="list-style-type: none"> ● Pseudo-códigos ● Problemas resueltos ● Demostraciones formales.

	<ul style="list-style-type: none"> • Cañón. • Pizarrón, pintarrones. 	
--	--	--

FUENTES DE INFORMACIÓN (Bibliografía, Direcciones electrónicas)	EVALUACIÓN DE LOS APRENDIZAJES (Criterios e instrumentos)
<ol style="list-style-type: none"> 1. Kelley, D., Aguilar, L. J., & Platas, M. L. D. (1995). <i>Teoría de autómatas y lenguajes formales</i> (Vol. 22). Prentice Hall. 2. Sipser, M. (2012). <i>Introduction to the Theory of Computation</i>. Cengage Learning.. 3. Hopcroft, J. E., Ullman, J. D., & Motwani, R. (2002). <i>Introducción a la teoría de autómatas, lenguajes y computación</i>. 4. Garey, M. R., & Johnson, D. S. (1979). <i>Computers and intractability: a guide to NP-completeness</i>. 	<p>INSTRUMENTOS:</p> <p>Examen escrito Solución de problemas Programas computacionales(Pseudo-código)</p> <p>CRITERIOS DE DESEMPEÑO:</p> <p>Los exámenes por escrito: valoran el nivel de argumentación en relación al hecho que se quiere demostrar. Manejo de lenguaje técnico, coherencia entre párrafos y global, redacción, ortografía y presentación. Se utiliza una rúbrica para autoevaluación y heteroevaluación.</p> <p>Los problemas y las demostraciones: valoran el conocimiento teórico aplicado a la resolución de un ejercicio, debe contener el procedimiento y el resultado correcto. Se utiliza lista de cotejo para autoevaluación y heteroevaluación.</p> <p>Los programas computacionales: valoran la eficiencia del código y el conocimiento teórico aplicado para la programación del problema. Se utiliza una rúbrica para autoevaluación y heteroevaluación.</p> <p>La acreditación del curso:</p> <ul style="list-style-type: none"> • Examen intermedio: 30% • Examen final: 40% • Proyecto final: 30% <p>LAS ACTIVIDADES NO REALIZADAS EN TIEMPO Y FORMA SE CALIFICAN CON CERO. Nota: la calificación mínima aprobatoria es de 80.</p>

Cronograma del Avance Programático

S e m a n a s

Unidades de aprendizaje	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
I. Lenguajes regulares y autómatas finitos.	■	■	■	■	■											
II. Lenguajes independientes del contexto y autómatas de pila.						■	■	■	■	■						
III. Decidibilidad.											■	■	■	■	■	■