



1. Identificación del curso

Teoría de la computación							
Programa educativo				Departamento de adscripción			
Licenciatura en Ingeniería en Computación				Departamento de Ingenierías			
Área de formación				Tipo de Unidad de Aprendizaje			
Básica particular				Curso - Taller			
Carga horaria						Créditos	Clave
Teoría	40	Práctica	40	Total	80	8	IL357
Modalidad de Enseñanza - Aprendizaje				Prerrequisito			
Presencial				No aplica			
Academia				Profesor responsable			
Ciencias computacionales				Fernando Cornejo Gutiérrez			
Elaboró / Modificó				Fecha de elaboración / modificación			
Fernando Cornejo Gutiérrez				31 de octubre de 2022			

2. Competencias que abonan al perfil de egreso

Transversal	Disciplinar	Profesional
<p>Aplica habilidades de investigación para la solución de problemas. Posee habilidades de trabajo en equipo que le permiten desarrollarse como líder de proyectos en su campo profesional o integrarse a un grupo ya establecido. Posee habilidades de aprendizaje autogestivo que le permiten incrementar sus conocimientos en distintas áreas de interés.</p>	<p>Posee capacidad de razonamiento crítico, lógico y matemático para resolver problemas dentro de su área de estudio a través de modelos abstractos que reflejen situaciones reales. Demuestra conocimientos esenciales de principios y teorías relacionadas a las ciencias computacionales y su aplicación para la resolución de problemas propios de la ingeniería.</p>	<p>Aplicar los principios matemáticos y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real, para el desarrollo de un sistema de software.</p>

3. Saberes previos

Matemáticas discretas (funciones, relaciones), teoría de conjuntos, principios de teoría de grafos

4. Presentación de la unidad de aprendizaje

Unidad de aprendizaje que está enfocada en el desarrollo de la capacidad del ingeniero en computación para el diseño y análisis de sistemas formales que permitan representar por medio de las bases teóricas de la computación el funcionamiento de los compiladores, lenguajes formales y máquinas abstractas.

5. Objetivo de aprendizaje

Aplicar las técnicas y metodologías de los principios teóricos de la computación para el diseño y construcción de sistemas formales y máquinas abstractas que representen situaciones reales.

6. Competencia general de la unidad de aprendizaje

A-CC.118. Plantea y resuelve problemas matemáticos aplicados a la computación. Reconoce el contexto y necesidades involucrados en un sistema, empleando técnicas para identificar, obtener, analizar, priorizar, documentar, verificar y validar los requisitos. (ANIEI A-CC.118)



7. Habilidades, valores y actitudes

Liderazgo, respeto a las opiniones de los demás, colaboración y trabajo en equipo. Muestra interés al aprendizaje continuo. Valora la retroalimentación grupal.

8. Elementos de competencia

Bloque No. I: Introducción a los Lenguajes Formales		
Sub-competencia	Comprende los conceptos básicos para el diseño de lenguajes formales a partir de las gramáticas formales que los generan y su relación con los lenguajes de programación.	
Cognitivos (Contenido)		
<ul style="list-style-type: none"> • Teoría de conjuntos • Lenguajes formales <ul style="list-style-type: none"> ○ Simbología, alfabeto y cadenas ○ Operaciones • Gramáticas formales <ul style="list-style-type: none"> ○ Notación BNF ○ Árboles de derivación • Jerarquía de Chomsky • Ambigüedad • Principios de la teoría de autómatas 		
Procedimentales		
Soluciona ejercicios sobre las operaciones con lenguajes y la producción a partir de gramáticas formales, a través de su expresión mediante notación matemática.		
Estrategias didácticas		
Exposición por parte del profesor. Resolución de problemas/ejercicios. Trabajo colaborativo.		
Criterios de desempeño	Producto esperado	Sesiones estimadas
Orden, limpieza, puntualidad en la entrega. Razonamiento lógico-matemático para la generación de lenguajes.	Portafolio de evidencias	18
Área de conocimiento	2.4 Teoría matemática de la computación	

Bloque No. II: Autómatas de estado finito	
Sub-competencia	Construye autómatas finitos para la resolución de problemas que reconozca diferentes lenguajes regulares, así como identifica su relación con las expresiones regulares.
Cognitivos (Contenido)	
<ul style="list-style-type: none"> • Autómatas de estado finito <ul style="list-style-type: none"> ○ Representación (grafo, tabla de transiciones y producciones δ) ○ Tipos (Deterministas, No deterministas y Transición λ) • Relación AF y gramática regular • Expresiones regulares • Teorema de Arden • Minimización de AFND a AFD 	
Procedimentales	



Representa el funcionamiento de un autómata finito por medio de un grafo y tablas de transición, así como la relación entre las gramáticas regulares.

Estrategias didácticas

Exposición por parte del profesor.
Resolución de problemas/ejercicios.
Trabajo colaborativo.

Criterios de desempeño	Producto esperado	Sesiones estimadas
Orden, limpieza, puntualidad en la entrega. Razonamiento lógico-matemático para el diseño de AF.	<ul style="list-style-type: none"> Portafolio de evidencias. Producto integrador: diseño de un AF a partir de un problema real 	24
Área de conocimiento	2.4 Teoría matemática de la computación	

Bloque No. III: Autómatas de pila

Sub-competencia Diseña autómatas de Pila para la resolución de problemas que reconozcan diferentes lenguajes generados por una gramática independiente del contexto.

Cognitivos (Contenido)

- Autómatas de Pila
 - Representación (grafo, tabla de transiciones y producciones δ)
 - Tipos (Deterministas, No deterministas)
 - Tabla de operación, Notación Turnstile, Árbol de descripciones
- Relación AP y gramática independiente del contexto
- Forma Normal de Greibach (GNF)

Procedimentales

Representa el funcionamiento de un autómata de pila por medio de un grafo y tablas de transición, así como la relación entre las gramáticas independiente del contexto.

Estrategias didácticas

Exposición por parte del profesor.
Resolución de problemas/ejercicios.
Trabajo colaborativo.

Criterios de desempeño	Producto esperado	Sesiones estimadas
Orden, limpieza, puntualidad en la entrega. Razonamiento lógico-matemático para el diseño de AP.	<ul style="list-style-type: none"> Portafolio de evidencias. Producto integrador: diseño de un AP a partir de un problema real 	20
Área de conocimiento	2.4 Teoría matemática de la computación	

Bloque No. IV: Máquinas de Turing y Complejidad Computacional

Sub-competencia Diseña y construye autómatas de Máquina Turing para la resolución de problemas que reconozcan diferentes lenguajes a partir de su relación con la complejidad computacional.

Cognitivos (Contenido)

- Máquinas de Turing
 - Definición formal
 - Operación y diseño
 - Máquinas multipista y problema de la parada
- Funciones computables
- Complejidad computacional



- Problemas P, NP y NP-Completo

Procedimentales

Representa el funcionamiento de una máquina de Turing por medio de un grafo y tablas de transición, así como la relación con la complejidad computacional.

Estrategias didácticas

Exposición por parte del profesor.
Resolución de problemas/ejercicios.
Trabajo colaborativo.

Criterios de desempeño	Producto esperado	Sesiones estimadas
Orden, limpieza, puntualidad en la entrega. Razonamiento lógico-matemático para el diseño de MT.	Portafolio de evidencia	18
Área de conocimiento	2.4 Teoría matemática de la computación	

Nota: 1 sesión = 1 hora;

9. Recursos requeridos

Videoprojector, computadora, paquetería de Ofimática, *software* JFLAP, Plataforma Moodle

10. Evaluación y acreditación de la unidad de aprendizaje

- Exámenes Parciales 30%
- Examen Final 20%
- Actividades de Investigación 10%
- Resolución de Ejercicios 15%
- Actividades integradoras 20%
- Participación 5%

11. Referencias (APA)

Básica

Giró, J., Vázquez, J., Meloni, B. & Constable, L. (2015). *Lenguajes formales y teoría de autómatas*. Alfaomega.

Hopcroft, J. (2007) *Teoría de autómatas, lenguajes y computación*. Addison Wesley.

Aldonseca E., Alfonseca M. (2007) *Teoría de Autómatas y lenguajes formales*. Madrid: McGraw-Hill.

Complementaria

Xavier, E. (2005). *Theory of Automata, Formal Languages and Computation*. New Age International (P) Ltd., Publishers.

Jiménez, J. A. (2010). *Matemáticas para la computación* (3a ed.). Alfaomega.

Epp, S. (2011). *Matemáticas discretas con aplicaciones* (4a ed.). Cengage Learning.

Sitios web



12. Campo de aplicación profesional

El estudiante es capaz de identificar las bases teóricas de las ciencias computacionales que rigen la construcción de lenguajes de programación y su costo computacional, así como la aplicación de estas para el diseño del comportamiento de máquinas y sistemas formales.

13. Perfil docente

El docente de esta materia deberá ser un profesionista con formación en el área de las ciencias computacionales; capaz de motivar a la investigación y creación de conocimiento, con habilidades para transmitir sus conocimientos y enseñar de forma interactiva propiciando en los alumnos el autoaprendizaje.



Dr. Alejandro Pérez Larios
Jefe de departamento de ingenierías

CENTRO UNIVERSITARIO DE LOS ALTOS
DIVISIÓN DE CIENCIAS AGROPECUARIAS E INGENIERÍAS
DEPARTAMENTO DE INGENIERÍAS

Dra. Claudia Islas Torres
Presidente de academia