

IDENTIFICACIÓN DE LA UNIDAD DE APRENDIZAJE

Unidad académica: Instituto de Investigación en Ciencias Básicas y Aplicadas							
Plan de estudios: Licenciatura en Inteligencia Artificial							
Unidad de aprendizaje: INTRODUCCIÓN A LA TEORÍA DE LA COMPUTACIÓN				Ciclo de formación: Profesional Eje general de formación: Teórico-Técnica Área de conocimiento: Fundamentos teóricos de la computación Semestre: 5º			
Elaborada por: Dr. Jorge Hermosillo Valadez Dr. Daniel Rivera López Dra. Elisa Chinos Olivan				Fecha de elaboración: Abril, 2021			
Clave:	Horas teóricas :	Horas prácticas :	Horas totales :	Créditos :	Tipo de unidad de aprendizaje :	Carácter de la unidad de aprendizaje :	Modalidad:
IT37CP030208	03	02	05	08	Obligatoria	Teórico - Práctica	Escolarizada
Plan (es) de estudio en los que se imparte: A partir de todos los programas impartidos por el Instituto de Investigación en Ciencias Básicas y Aplicadas.							

ESTRUCTURA DE LA UNIDAD DE APRENDIZAJE

<p>Presentación:</p> <p>Se estudian los fundamentos teóricos de las ciencias de la computación. Para saber si un problema se puede resolver o no con una computadora, es necesario formalizar el concepto de algoritmo e introducir un modelo teórico de las computadoras. El estudio de estas máquinas nos ayuda a formalizar el proceso y concepto de cómputo que nos permitirá tener más idea de lo que se puede calcular, en qué tiempo y qué no se puede calcular en las computadoras. Clasificamos los problemas/lenguajes en decidibles, no-</p>
--



decidibles, reconocibles y no-reconocibles. Los problemas decidibles son los que se pueden resolver en las computadoras, los reconocibles se pueden resolver en condiciones favorables, pero no son resolubles en general. Y los problemas no-reconocibles son todavía mucho más complicados que los problemas reconocibles. En esta unidad de aprendizaje se estudiará la técnica llamada de reducción. Informalmente, un problema/lenguaje se reduce a otro si el primero se puede resolver teniendo la solución del segundo.

Propósito:

Conozca, aprenda y comprenda los modelos matemáticos de las computadoras, al término de la unidad de aprendizaje, mediante el conocimiento que sustenta el modelo teórico y conceptual de las computadoras y del quehacer computacional, con la finalidad de profundizar en los fundamentos teóricos de las ciencias de la computación aprendiendo a clasificar los problemas/lenguajes en decidibles, no-decidibles, reconocibles y no-reconocibles brindando elementos para el enriquecimiento de la comprensión de la disciplina computacional, con responsabilidad social.

Competencias que contribuyen al perfil de egreso

Competencias genéricas:

- CG15. Capacidad para formular y gestionar proyectos.
- CG18. Capacidad para actuar en nuevas situaciones.
- CG20. Capacidad de expresión y comunicación.

Competencias específicas:

- CE4. Analiza soluciones computacionales mediante la aplicación de fundamentos teóricos del diseño de algoritmos y estructuras de datos adecuadas para resolver problemas con pensamiento crítico.
- CE5. Descubre nuevas tecnologías, herramientas y estándares en su área de especialidad, mediante la consulta continua de las actualizaciones para mantenerse a la vanguardia en su campo con autonomía y creatividad.
- CE6 Contribuye al desarrollo científico y tecnológico, mediante la aplicación de conocimientos y habilidades computacionales adquiridos para la solución innovadora de problemas planteados en el sector público, privado o social con una actitud ética y responsable.

CONTENIDOS



Bloques	Temas
<p>1. Lenguajes regulares y autómatas finitos.</p>	<p>1.1. Introducción a la computabilidad. 1.2. Gramáticas y lenguajes. Gramáticas regulares. 1.3. Autómatas finitos deterministas, con y sin salidas, reconocedores. Autómatas finitos no-deterministas y la equivalencia de autómatas deterministas y no-deterministas. 1.4. Expresiones regulares, cerradura sobre expresiones regulares. La equivalencia entre los autómatas de estado finito y expresiones regulares. 1.5 Lenguajes no-regulares.</p>
<p>2. Lenguajes libres del contexto y autómatas con pila.</p>	<p>2.1. La definición de un lenguaje libre del contexto y ejemplos de estos lenguajes. Construcción de lenguajes libres del contexto. 2.2 La reducción a la forma normal de Chomsky. 2.3 La definición formal de autómatas con pila (pushdown automation) y ejemplos. 2.4 Equivalencia entre los lenguajes de libres del contexto y autómatas con pila. 2.5 Lenguajes que no son de libres del contexto, lema de bombeo para los lenguajes de libres del contexto.</p>
<p>3. Modelo general de computación y la hipótesis de Church-Turing</p>	<p>3.1. Tesis de Church y Turing. 3.2. Máquinas de Turing. 3.2.1. Lenguajes decidibles y reconocibles 3.2.2. Variaciones de máquinas de Turing con multi-cinta 3.2.3. Máquinas de Turing no deterministas 3.2.4. Equivalencia de las máquinas de Turing deterministas y no-deterministas</p>



	3.2.5. Equivalencia de las máquinas de Turing con otros modelos de computación 3.3. Noción de algoritmo.
4. Decibilidad	4.1. Lenguajes decidibles: lenguajes decidibles del campo de conjuntos regulares, autómatas finitos, lenguajes libres de contexto y autómatas con pila. 4.2 Problemas no-decidibles. Problema de parada de una máquina de Turing, el método de diagonalización. Lenguajes no-reconocibles.
5. Reducibilidad	5.1. El método de reducción para demostrar que un lenguaje es o no decidible. 5.2. Problemas no-decidibles de teoría de lenguajes. 5.3. El problema de post-correspondencia. 5.4. Definición formal de reducibilidad. Reducibilidad por mapeo, funciones computables.

ESTRATEGIAS DE ENSEÑANZA - APRENDIZAJE

Estrategias de aprendizaje sugeridas (Marque X)			
Aprendizaje basado en problemas	(X)	Nemotecnia	()
Estudios de caso	()	Análisis de textos	()
Trabajo colaborativo	()	Seminarios	()
Plenaria	()	Debate	()
Ensayo	()	Taller	()
Mapas conceptuales	()	Ponencia científica	()
Diseño de proyectos	(X)	Elaboración de síntesis	()
Mapa mental	()	Monografía	()
Práctica reflexiva	()	Reporte de lectura	()
Trípticos	()	Exposición oral	(X)



Otros			
Estrategias de enseñanza sugeridas (Marque X)			
Presentación oral (conferencia o exposición) por parte del docente	(X)	Experimentación (prácticas)	(X)
Debate o Panel	()	Trabajos de investigación documental	()
Lectura comentada	(X)	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)	(X)	Método de proyectos	(X)
Interacción la realidad (a través de videos, fotografías, dibujos y software especialmente diseñado).	()	Actividades generadoras de información previa	()
Organizadores previos	()	Exploración de la web	()
Archivo	()	Portafolio de evidencias	()
Ambiente virtual (foros, chat, correos, ligas a otros sitios web, otros)	()	Enunciado de objetivo o intenciones	(X)

CRITERIOS DE EVALUACIÓN

Criterios	Porcentaje
<ul style="list-style-type: none"> • Resolución de problemas 	30%
<ul style="list-style-type: none"> • Proyecto final 	40%
<ul style="list-style-type: none"> • Prácticas 	30%



Total	100 %
--------------	-------

PERFIL DEL PROFESORADO

Licenciatura, Maestría o Doctorado en ciencias computacionales, matemáticas o ingeniería en áreas afines a las ciencias computacionales, con experiencia docente en el área.

REFERENCIAS

Básicas:

- Sipser, M. (1997). *Introduction to theory of computation*. Ed. PWS Publishing Company.
- Kenneth H. Rosen.(2004). *Matemática Discreta y sus aplicaciones*. Quinta edición.
- Susanna S. Epp. *Matemáticas Discretas con sus aplicaciones*. Cuarta edición.
- Cooper, S. (2004). *Computability theory*. Ed. Chapman & Hall/CRC.

Complementarias:

- Hopcroft, J., Motwani, R. y Ullman, J. (2000). *Introducción a la teoría de autómatas, lenguajes y computación*. Ed. Prentice-Hall.
- Hopcroft, J., Motwani, R. y Ullman, J. *Teoría de autómatas, lenguajes y computación*. 3a edición. Ed. Pearson.
- Garey, M. y Johnson, D. (1979). *Computers and intractability: a guide to theory of NP-completeness*. Ed. Freeman and Company.

