


<p>UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA</p>  <p>UNIDAD ACADÉMICA:</p> <p>FACULTAD DE INGENIERÍA</p> <p>PROGRAMA ANALÍTICO DE LA UNIDAD DE APRENDIZAJE:</p> <p><u>COMPILADORES E INTERPRETES</u></p>	DES:	INGENIERÍA
	Programa académico	Ingeniería en Computación
	Tipo de materia (Obli/Opta):	Optativa
	Clave de la materia:	OPCO703
	Semestre:	Séptimo
	Área en plan de estudios:	Específica
	Total de horas por semana:	6
	<i>Teoría: Presencial o Virtual</i>	0
	<i>Laboratorio o Taller:</i>	4
	<i>Prácticas:</i>	0
	<i>Trabajo extra-clase:</i>	2
	Créditos Totales:	6
	Total de horas semestre (x sem):	64
	Fecha de actualización:	Octubre 2024
<i>Prerrequisito (s):</i>	MC401 Programación Orientada a Objetos	
DESCRIPCIÓN:		
<p>El curso proporciona al estudiante el desarrollo y construcción de sentencias de código creadas mediante la programación de analizadores de texto, haciendo uso de técnicas y lógica de programación, así como conceptos abstractos para la creación de compiladores e intérpretes.</p>		
COMPETENCIAS PARA DESARROLLAR:		
B4. TRANSFORMACIÓN DIGITAL		
<p>Transforma la cultura digital en la sociedad, en las organizaciones e instituciones educativas para aprovechar al máximo el potencial de las tecnologías y herramientas digitales; propiciar su uso responsable y ético que estimule la creatividad, innovación, la comunicación efectiva y el trabajo colaborativo e interdisciplinar en la solución de problemas de la sociedad digital; promoviendo la privacidad y la seguridad, así como el respeto a los derechos de autor y la propiedad intelectual.</p>		
P1. CIENCIAS E INGENIERÍA		
<p>Aplica los conocimientos y metodologías para el planteamiento y resolución de problemas complejos de las ciencias naturales y de la ingeniería, para la toma de decisiones en un contexto de responsabilidad social y del medio ambiente.</p>		

DOMINIOS	OBJETOS DE ESTUDIO	RESULTADOS DE APRENDIZAJE	METODOLOGÍA	EVIDENCIAS
<p>B4.7 Colabora de forma interdisciplinar en el desarrollo de propuestas de innovación y transformación que impulsen el bienestar de las comunidades y la sociedad.</p> <p>P1. CIENCIAS E INGENIERÍA</p> <p>1. Utiliza conceptos, métodos y leyes fundamentales de las ciencias básicas para dar soluciones a problemas complejos de ciencias e ingeniería analizando los resultados para emitir conclusiones acordes a la realidad.</p>	<p>UNIDAD 1: INTRODUCCIÓN</p> <p>1.1 Traductores.</p> <p>1.2 Compiladores.</p> <p>1.3 Intérpretes.</p> <p>1.4 Fases del proceso de traducción.</p> <p>1.4.1 Análisis léxico.</p> <p>1.4.2 Análisis sintáctico.</p> <p>1.4.3 Análisis semántico.</p> <p>1.4.5 Generación de código.</p> <p>1.5 Herramientas para la implementación de traductores.</p>	<p>Distingue los diferentes procesos para el diseño de gramáticas para un lenguaje de programación específico.</p>	<ul style="list-style-type: none"> ○ Clase interactiva Maestro - Alumno. ○ Uso de recursos tecnológicos institucionales ○ Uso de un lenguaje de programación y su interfaz de desarrollo. ○ Ejercicios prácticos del tema desarrollados en clase en forma individual o grupal. 	<p>Resolución de ejercicios de clase a través de programas computacionales desarrollados en un lenguaje de programación</p> <p>Presentación de un programa computacional aplicando los conceptos presentados para la finalización del objeto de estudio.</p>

	<p>UNIDAD 2: ANÁLISIS LÉXICO</p> <p>2.1 Expresiones regulares.</p> <p>2.2 Expresiones regulares y autómatas finitos.</p> <p>2.3 Autómatas finitos y reconocedores.</p> <p>2.4 Implementación de un analizador léxico.</p> <p>2.4.1 Uso de Lex para la implementación.</p>	<p>Desarrolla analizadores léxicos para el tratamiento de texto que identifiquen elementos básicos tipo tokens.</p>	<ul style="list-style-type: none"> ○ Clase interactiva Maestro - Alumno. ○ Uso de recursos tecnológicos institucionales. ○ Uso de un lenguaje de programación y su interfaz de desarrollo. ○ Ejercicios prácticos del tema desarrollados en clase en forma individual o grupal. 	<p>Resolución de ejercicios de clase a través de programas computacionales desarrollados en un lenguaje de programación</p> <p>Presentación de un programa computacional aplicando los conceptos presentados para la finalización del objeto de estudio.</p>
	<p>UNIDAD 3: ANÁLISIS SINTÁCTICO</p> <p>3.1 Sintaxis y gramáticas</p>	<p>Desarrolla analizadores sintácticos para el</p>	<ul style="list-style-type: none"> ○ Clase interactiva Maestro - Alumno. 	<p>Resolución de ejercicios de clase</p>
	<p>libres de contexto.</p> <p>3.2 Análisis sintáctico descendente.</p> <p>3.3 Análisis sintáctico ascendente.</p> <p>3.4 Implementación de un analizador sintáctico.</p> <p>3.4.1 Uso de Bison para la implementación.</p>	<p>tratamiento de texto que comprueban procedencia de operadores como preámbulo a un lenguaje de programación.</p>	<ul style="list-style-type: none"> ○ Uso de recursos tecnológicos institucionales ○ Uso de un lenguaje de programación y su interfaz de desarrollo. ○ Ejercicios prácticos del tema desarrollados en clase en forma individual o grupal. 	<p>a través de programas computacionales desarrollados en un lenguaje de programación</p> <p>Presentación de un programa computacional aplicando los conceptos presentados para la finalización del objeto de estudio.</p>

	<p>UNIDAD 4: TRADUCCIÓN DIRIGIDA POR SINTAXIS</p> <p>4.1 Atributos. 4.2 Tipos de atributos. 4.3 Definiciones dirigidas por sintaxis. 4.4 Grafo de dependencias. 4.5 Traducción ascendente y traducción descendente.</p>	<p>Aplica la información a una construcción del lenguaje proporcionando atributos a los símbolos de la gramática.</p>	<ul style="list-style-type: none"> ○ Clase interactiva Maestro - Alumno. ○ Uso de recursos tecnológicos institucionales ○ Uso de un lenguaje de programación y su interfaz de desarrollo. ○ Ejercicios prácticos del tema desarrollados en clase en forma individual o grupal. 	<p>Resolución de ejercicios de clase a través de programas computacionales desarrollados en un lenguaje de programación</p> <p>Presentación de un programa computacional aplicando los conceptos presentados para la finalización del objeto de estudio</p>
--	--	---	--	---

	<p>UNIDAD 5: GENERACIÓN DE CÓDIGO</p> <p>5.1 Generación de código intermedio. 5.1.1 Código de tres accesos. 5.2 Selección de instrucciones. 5.3 Calendarización de instrucciones. 5.4 Alojamiento de registros. 5.5 Generación de código en tiempo de ejecución.</p>	<p>Emplea el proceso de generación de código intermedio durante el proceso de desarrollo de un compilador.</p>	<ul style="list-style-type: none"> ○ Resolución de ejercicios de clase a través de programas computacionales desarrollados en un lenguaje de programación ○ Presentación de un programa computacional aplicando los conceptos presentados para la finalización del objeto de estudio. ○ Ejercicios prácticos del tema desarrollados en clase en forma individual o grupal 	<p>Presentación de un programa computacional aplicando los conceptos presentados para la finalización del objeto de estudio</p>
--	---	--	--	---

FUENTES DE INFORMACIÓN (Bibliografía, direcciones electrónicas)	EVALUACIÓN DE LOS APRENDIZAJES (Criterios, ponderación e instrumentos)
<p>Martín Villalba, C., Urquía Moraleda, A., & Rubio González, M. Á. (2021). <i>Lenguajes de programación</i>. Editorial UNED. ISBN 978-8436277494.</p> <p>Cooper, K. D., & Torczon, L. (2022). <i>Engineering a compiler</i> (3rd ed.). Morgan Kaufmann. ISBN 978-0128154120.</p> <p>Nystrom, R. (2021). <i>Crafting interpreters</i>. Genever Benning. ISBN 978-0990582946.</p> <p>Aho, Lam, Sethi Ullman (2008). <i>Compiladores, principios, técnicas y herramientas</i>. Pearson Adisson- Wesley</p> <p>Jacinto Ruiz Catalán (2012). <i>Compiladores. Teoría e Implementación</i>. RC Libros.</p> <p>Oviedo Regino, Efraín M.(2015). <i>Lógica De programación Orientada A Objetos</i>. Ecoe Ediciones.</p>	<p>Tres evaluaciones parciales durante el semestre.</p> <p>Ponderaciones:</p> <p>Primer parcial 30%</p> <ul style="list-style-type: none"> ● Desarrollo y revisión de ejercicios en clase 20% ● Tareas diversas 20% ● Presentación de programa computacional 60% <p>Segundo parcial: 30%</p> <ul style="list-style-type: none"> ● Desarrollo y revisión de ejercicios en clase 20% ● Tareas diversas 20% ● Presentación de programa computacional 60% <p>Tercer Parcial: 40%</p> <ul style="list-style-type: none"> ● Desarrollo y revisión de ejercicios en clase 20% ● Tareas diversas 20% ● Presentación de programa computacional 60% <p>Se usará rúbrica para la entrega y presentación de programa computacional realizar.</p> <p>El lenguaje de programación a utilizar se indicará por medio de la academia correspondiente a la materia.</p> <p>Calificación mínima 7.0 Cada actividad y/o tareas debe incluir rubricas y/o listas de cotejo.</p>

CRONOGRAMA

Objetos de estudio	Seman as																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
UNIDAD 1: INTRODUCCIÓN																	
UNIDAD 2: ANÁLISIS LÉXICO																	
UNIDAD 3: ANÁLISIS SINTÁCTICO																	
UNIDAD 4: TRADUCCIÓN DIRIGIDA POR SINTAXIS																	
UNIDAD 5: GENERACIÓN DE CÓDIGO																	