



PLAN DE ESTUDIOS (PE): Licenciatura en Ciencias de la Computación

AREA: Optativas

ASIGNATURA: Compiladores

CÓDIGO: CCOS-616

CRÉDITOS: 6

FECHA: 18 de Mayo de 2017





1. DATOS GENERALES

Nivel Educativo:	Licenciatura
Nombre del Plan de Estudios:	Licenciatura en Ciencias de la Computación
Modalidad Académica:	Presencial
Nombre de la Asignatura:	Compiladores
Ubicación:	Nivel Optativo
Correlación:	
Asignaturas Precedentes:	Lenguajes Formales y Autómatas
Asignaturas Consecuentes:	NA

2. CARGA HORARIA DEL ESTUDIANTE

Concepto	Horas por semana		Total de horas por periodo	Número de créditos
	Teoría	Práctica		
Horas teoría y práctica (16 horas = 1 crédito)	3	2	90	6





3. REVISIONES Y ACTUALIZACIONES

Autores:	Beatriz Beltrán Martínez Hilda Castillo Zacatelco Laura Cuayahuitl Romero Leticia Mendoza Alonso Mario Anzures García Pedro Bello López Yolanda Moyao Martínez Rafael De la Rosa Flores
Fecha de diseño:	25 de Junio de 2009
Fecha de la última actualización:	18 de Mayo de 2017
Fecha de aprobación por parte de la academia de área	18 de Mayo de 2017
Revisores:	Ana Patricia Cervantes Márquez Hilda Castillo Zacatelco Pedro Bello López Meliza Contreras González Yolanda Moyao Martínez Rafael de la Rosa Flores Leticia Mendoza Alonso Luis Enrique Colmenares Guillen Miguel Rodríguez Hernández Beatriz Beltrán Martínez Mario Rossainz López
Sinopsis de la revisión y/o actualización:	Se pasó al nuevo formato, se colocaron las competencias específicas y la contribución de la asignatura a dichas competencias. Se revisó y actualizó la bibliografía.

4. PERFIL DESEABLE DEL PROFESOR (A) PARA IMPARTIR LA ASIGNATURA:

Disciplina profesional:	Ciencias de la Computación
-------------------------	----------------------------





Nivel académico:	Grado preferente Maestría
Experiencia docente:	2 años
Experiencia profesional:	2 años

5. PROPÓSITO:

Que el alumno diseñe e implemente cada una de las fases involucradas en el diseño y construcción de un compilador utilizando técnicas formales para resolver problemas computacionales con áreas interdisciplinarias.

El alumno será capaz de:

- Identificar los tipos de traductores.
- Identificar los principios básicos para el análisis y diseño de un compilador.
- Describirá la estructura general de un compilador.
- Diseñar cada una de las fases de un compilador, aplicando técnicas formales.
- Utilizar autómatas finitos para expresiones regulares y gramáticas en el diseño de reconocedores léxicos.
- Diseñar un analizador sintáctico descendente.
- Aplicar las técnicas de verificación del ámbito y tipos en las variables.
- Identificar los tipos de representación de código intermedio para la construcción de la máquina virtual.

6. COMPETENCIAS PROFESIONALES:

Este programa de asignatura contribuye con las siguientes Competencias Específicas del

Plan de Licenciatura en Ciencias de la Computación





- Diseña y crea lenguajes de programación con base en los fundamentos teóricos y técnicos de procesamiento léxico, sintáctico y semántico de manera ética y responsable con la finalidad de estar a la vanguardia científica y tecnológica.
- Modela y diseña soluciones computacionales con base en los fundamentos matemáticos, principios algorítmicos y teorías de la Ciencia de la Computación para resolver diversas problemáticas sociales y laborales.

Debido a que al concluir el programa el alumno:

Aplica los conceptos fundamentales del diseño de un compilador para dar solución a problemas de diferentes áreas de la ciencia.

7. CONTENIDOS TEMÁTICOS

Unidad de Aprendizaje	Contenido Temático	Bibliografía
-----------------------	--------------------	--------------





<p>1. Introducción a los traductores</p>	<p>1.1 Conceptos generales. 1.2 Tipos de traductores. 1.3 Estructura del proceso de compilación.</p>	<p>Aho, Lam Sethi Ullman. (2014), Compilers Principles, Techniques. and Tools, 2nd Edition, Pearson Education Limited.</p> <p>Ruiz, C.J. (2010), Compiladores, Teoría e Implementación. Mexico: Alfaomega.</p> <p>Martínez López F.J., Ramallo Martinez A. (2014). Teoría, Diseño E Implementación De Compiladores De Lenguajes. RaMa Editorial.</p> <p>Francisco Giro J. (2015). Lenguajes Formales Y Teoría De Automatas. Alfaomega.</p> <p>Modern Compiler Implementation in Java 2nd Edition, Andrew W. Appel, Cambridge University Press, 2nd Edition, 2009</p>
--	--	---

Unidad de Aprendizaje	Contenido Temático	Bibliografía
<p>2. Análisis Léxico.</p>	<p>2.1 Definición 2.2 Conceptos de token, patrón, lexema y atributo. 2.3 Especificación de token y expresiones regulares. 2.4 El autómata finito 2.5 La tabla de transición. 2.6 Tratamiento de errores</p>	<p>Aho, Lam Sethi Ullman. (2014), Compilers Principles, Techniques. and Tools, 2nd Edition, Pearson Education Limited.</p> <p>Ruiz, C.J. (2010), Compiladores, Teoría e Implementación. Mexico: Alfaomega.</p>





		<p>Keith Cooper, Linda Torczon (2012), <i>Engineering a Compiler</i>, Elsevier, Second Edition 2012.</p> <p>Dick Grune, Kees van Reeuwijk, Henri E. Bal, Criel J.H. Jacobs, Koen Langendoen (2012), <i>Modern Compiler Design</i> 2nd ed. 2012 Springer Edition.</p>
--	--	--

Unidad de Aprendizaje	Contenido Temático	Bibliografía
3. Análisis Sintáctico.	<p>3.1 Gramáticas Libres de Contexto.</p> <p>3.2 Derivación.</p> <p>3.3 Ambigüedad.</p> <p>3.4 Tipos de Análisis Sintáctico</p> <p>3.5 Analizador Sintáctico Descendente Predictivo.</p> <p>3.6 Gramáticas LL(1).</p> <p>3.7 Tratamiento de errores sintácticos.</p>	<p>Aho, Lam Sethi Ullman. (2014), <i>Compilers Principles, Techniques. and Tools</i>, 2nd Edition, Pearson Education Limited.</p> <p>Martínez López F.J., Ramallo Martinez A. (2014). <i>Teoría, Diseño E Implementación De Compiladores De Lenguajes</i>. RaMa Editorial.</p> <p>Anthony J. Dos Reis (2012) <i>Compiler Construction Using Java, JavaCC, and Yacc</i>, Copyright © 2012 the IEEE Computer Society, Inc.Wiley.</p> <p>Michael L. Scott, (2016), <i>Programming Language Pragmatics</i>, Fourth Edition 4th Edition Elsevier.</p>





Unidad de Aprendizaje	Contenido Temático	Bibliografía
4.Análisis Semántico.	4.1 Introducción. 4.2 Determinación de los tipos de comprobaciones semánticas a realizar 4.3 Diseño, implantación y administración de la tabla de símbolos. 4.3.1 Análisis semántico de tipos. 4.3.2 Análisis semántico en declaraciones. 4.3.3 Análisis semántico de expresiones. 4.3.4 Análisis semántico de instrucciones.	Aho, Lam Sethi Ullman. (2014), Compilers Principles, Techniques. and Tools, 2nd Edition, Pearson Education Limited. Ruiz, C.J. (2010), Compiladores, Teoría e Implementación. Mexico: Alfaomega. Martínez López F.J., Ramallo Martinez A. (2014). Teoría, Diseño E Implementación De Compiladores De Lenguajes. RaMa Editorial Terence Parr (2010), Language Implementation Patterns: Create Your Own Domain-Specific and General Programming Languages, Pragmatic Programmers, 1st Edition.

Unidad de Aprendizaje	Contenido Temático	Bibliografía
-----------------------	--------------------	--------------





<p>5. Generación de código intermedio.</p>	<p>5.1 Máquina virtual. 5.2 Administración de la memoria durante la ejecución. 5.3 Códigos de 3 y 4 direcciones. 5.4 Estrategias para la generación de código estándar.</p>	<p>Aho, Lam Sethi Ullman. (2014), Compilers Principles, Techniques, and Tools, 2nd Edition, Pearson Education Limited.</p> <p>Martínez López F.J., Ramallo Martinez A. (2014). Teoría, Diseño E Implementación De Compiladores De Lenguajes. RaMa Editorial</p> <p>Ronald Mak (2011), Writing Compilers and Interpreters: A Software Engineering Approach 3rd Edition.</p> <p>Reinhard W., Helmut S. (2010). Compiler Design: Virtual Machines. Springer Science & Business Media.</p>

8. ESTRATEGIAS, TÉCNICAS Y RECURSOS DIDÁCTICOS

Estrategias y Técnicas didácticas	Recursos didácticos
-----------------------------------	---------------------





<p>Estrategias de aprendizaje:</p> <ul style="list-style-type: none"> • Lectura y comprensión • Reflexión • Comparación <p>Estrategias de enseñanza:</p> <ul style="list-style-type: none"> • Aprendizaje activo • Aprendizaje cooperativo • Aprendizaje colaborativo <p>Ambientes de aprendizaje:</p> <ul style="list-style-type: none"> • Aula • Laboratorio • Simuladores <p>Actividades y experiencias de aprendizaje:</p> <ul style="list-style-type: none"> • Asistencia a congresos <p>Técnicas</p> <ul style="list-style-type: none"> • Grupales • De problemas • De estudio de casos • Mapas conceptuales • Para el análisis • Comparación • Síntesis • Lluvia de ideas • Portafolio • Exposición 	<p>Materiales:</p> <ul style="list-style-type: none"> • Proyector • TICs • Plumón y Pizarrón • Ejercicios • Prácticas de Laboratorio • Libros, fotocopias • Artículos científicos • Antologías • Materiales audiovisuales • Programas informáticos (CD u online) educativos. • Aplicaciones Multimedia • Páginas Web • Correo electrónico • Chats • Foros • Links
---	--

9. EJES TRANSVERSALES

Eje (s) transversales	Contribución con la asignatura
Formación Humana y Social	Promueve el trabajo en equipo, el respeto a los demás y al medio ambiente, responsabilidad respecto a la entrega oportuna de trabajos.



Desarrollo de Habilidades en el uso de las Tecnologías de la Información y la Comunicación	Uso de la tecnología para el desarrollo de sus proyectos durante el curso.
Desarrollo de Habilidades del Pensamiento Complejo	Promueve la reflexión y la toma de decisiones para generar alternativas de solución en el diseño de traductores.
Lengua Extranjera	Uso de textos y/o artículos en una lengua extranjera para el desarrollo de los temas del curso.
Innovación y Talento Universitario	Uso de los conocimientos adquiridos durante el curso para el diseño de un nuevo lenguaje de programación imperativo y la construcción del compilador correspondiente.
Educación para la Investigación	Promueve la investigación en cada uno de los temas para el desarrollo de sus proyectos del curso.

10. CRITERIOS DE EVALUACIÓN

Crterios	Porcentaje
<input type="checkbox"/> Tareas	15 %
<input type="checkbox"/> Proyecto final	30 %
<input type="checkbox"/> Pruebas objetivas	30 %
<input type="checkbox"/> Participación en clase	10 %
<input type="checkbox"/> Asistencia	5 %
<input type="checkbox"/> Entregas puntuales	5 %
<input type="checkbox"/> Presentación de trabajos	5 %
Total	100%





11. REQUISITOS DE ACREDITACIÓN

Estar inscrito como alumno en la Unidad Académica en la BUAP
Asistir como mínimo al 80% de las sesiones
La calificación mínima para considerar un curso acreditado será de 6
Cumplir con las actividades académicas y cargas de estudio asignadas que señale el PE

Notas:

- a) La entrega del programa de asignatura con sus respectivas actas de aprobación, deberá realizarse en formato electrónico, vía oficio emitido por la Dirección o Secretaría Académica a la Dirección General de Educación Superior.
- b) La planeación didáctica deberá ser entregada a la coordinación de la licenciatura en los tiempos y formas acordados por la Unidad Académica.

