

**UNIVERSIDAD AUTÓNOMA DE CHIAPAS
LICENCIATURA EN SISTEMAS COMPUTACIONALES**

Área de formación: Disciplinaria
Unidad académica: Compiladores
Ubicación: Quinto semestre
Clave: _____
Horas semana-mes: 5
Horas Teoría: 3
Horas Práctica: 2
Créditos: 8

PRESENTACIÓN

El conocimiento de la teoría y las técnicas que se utilizan en la construcción de compiladores le permitirá al estudiante aplicar los conocimientos en lenguajes de programación, arquitectura de computadoras, teoría matemática de la computación, algoritmos e ingeniería de software. Así también conocer las diferentes herramientas automáticas que permiten la construcción de compiladores e intérpretes.

OBJETIVO GENERAL

Al terminar el curso el estudiante aplicará los principios, técnicas, estructuras y herramientas para la construcción de compiladores.

UNIDAD I.- INTRODUCCIÓN A LA TEORIA DE COMPILADORES

TIEMPO APROXIMADO: 5 Horas

OBJETIVO DE LA UNIDAD: Conocer los fundamentos y herramientas para el diseño de un compilador.

CONTENIDO

- 1.1 Conceptos generales
- 1.2 Etapas de un compilador
- 1.3 Compiladores e Intérpretes
- 1.4 Herramientas automáticas para generar compiladores

UNIDAD II.- ANÁLISIS LÉXICO

TIEMPO APROXIMADO: 15 Horas

OBJETIVO DE LA UNIDAD: Construir un analizador léxico.

CONTENIDO

- Definición de análisis léxico
- 2.1 Expresiones regulares
 - 2.2 Autómatas.

- 2.2.1 Autómatas no determinísticos
- 2.2.2 Autómatas determinísticos
- 2.3 Matrices de transición
- 2.4 Tabla de símbolos
- 2.5 Programación de un analizador léxico básico
- 2.7 Diferentes herramientas automáticas para generar analizadores léxicos

UNIDAD III.- ANÁLISIS SINTÁCTICO

TIEMPO APROXIMADO: 20 Horas

OBJETIVO DE LA UNIDAD: Aplicar las técnicas de desarrollo y construcción de analizadores sintácticos.

CONTENIDO

- 3.1 Diagramas de sintaxis
- 3.2 Gramáticas
 - 3.2.1 Métodos descendentes
 - 3.2.1.1 Retroceso
 - 3.2.1.2 Predictivos
 - 3.2.1.2.1 Recursivo
 - 3.2.1.2.2 No recursivo
 - 3.2.2 Métodos ascendentes
 - 3.2.2.1 Desplazamiento/reducción
- 3.3 Algoritmos para el análisis sintáctico
- 3.4 Analizadores LL
- 3.5 Analizadores LR y LALR
- 3.6 Manejo de errores
- 3.7 Programación de un analizador sintáctico
- 3.8 Diferentes herramientas automáticas para generar analizadores sintácticos

UNIDAD IV.- ANÁLISIS SEMÁNTICO

TIEMPO APROXIMADO: 20 Horas

OBJETIVO DE LA UNIDAD. Aplicar las técnicas de notaciones para asociar reglas semánticas en la construcción de gramáticas.

CONTENIDO

- 4.1 Traducción dirigida por sintaxis
- 4.2 Gramáticas de atributo
- 4.3 Desarrollo de intérpretes
- 4.4 Analizadores LL y LR con símbolos de acción
- 4.5 Generación de código intermedio
- 4.6 Manejo de la tabla de símbolos
- 4.7 Programación de un analizador semántico

UNIDAD V.- GENERACIÓN DE CÓDIGO

TIEMPO APROXIMADO: 20 Horas

OBJETIVO DE LA UNIDAD. Aplicar las técnicas y herramientas automáticas para la generación de código.

CONTENIDO

- 5.1 Aspectos del diseño de un generador de código
- 5.2 La máquina objeto
- 5.3 Generación de directivas
- 5.4 Generación de código de los árboles de sintaxis
- 5.5 Optimización de código
- 5.6 Generación de código estándar
- 5.7 Administración de la memoria durante la ejecución
- 5.8 Herramientas automáticas para generación de código
- 5.9 Programación de un compilador básico

EXPERIENCIAS DE APRENDIZAJE

- Trabajar en equipo para la solución de problemas
- Investigación en distintos medios como Internet de las diferentes herramientas que generan analizadores léxicos, sintácticos y semánticos.
- Integrar conocimientos previos de programación, estructura de datos y otros

PRÁCTICAS SUGERIDAS

- Prácticas de laboratorio para realizar analizadores léxicos, sintácticos y semánticos utilizando un lenguaje de programación.
- Uso de herramientas automáticas para generar analizadores léxicos, sintácticos y semánticos.
- Construcción de un pequeño compilador que genere código objeto en lenguaje ensamblador.

CRITERIOS DE EVALUACIÓN

Exámenes escritos	30%
Prácticas	30%
Proyecto final	<u>40%</u>
Total	100%

BIBLIOGRAFÍA BÁSICA

Alfred V. Aho, Ravi Sethi, Jeffrey Ullman. (1998). Compiladores. Principios, técnicas y herramientas. Prentice Hall.

Lemone, Karen. (1996). Fundamentos de compiladores. CECSA.

Kelley, Dean. (2000). Teoría de Autómatas y Lenguajes Formales. Prentice Hall, 2ª Edición.

BIBLIOGRAFÍA COMPLEMENTARIA

Guerra Crespo, Héctor. (2005). Compiladores. Tecnología Didáctica.

Hopcroft, John, Jeffrey Ullman. (1995). Introducción a la teoría de autómatas, lenguajes y computación. CECSA.

Tucker, Allen. (1999). Lenguajes de programación, principios y paradigmas. Prentice Hall,

Página que contiene el manual de flex. Generador de analizadores léxicos.
<http://www.hispafuentes.com/hf-doc/man/man1/flex++.1.html>

Página que contiene el manual de bison. Generador de analizadores sintácticos y semánticos.
<http://www.es.tldp.org/Manuales-LuCAS/BISON/bison-es-1.27.html>

Página tutorial de lex. Generador de analizador léxico y yacc generador de analizador sintáctico y semántico.
<http://mailweb.udlap.mx/~is111936/oto02/documentos/tutoriales/lex&yacc/>

Página que contiene todo lo relacionado con la construcción de compiladores e intérpretes.
<http://www.ucse.edu.ar/fma/compiladores/>