



## PROGRAMA DE ESTUDIO

1. DATOS DE IDENTIFICACIÓN					
UNIDAD DE APRENDIZAJE O MÓDULO:	SOFTWARE DE SISTEMAS				
Clave:					
Ubicación:	SEMESTRE VII	Área: Profesionalizante			
Horas y créditos:	Teóricas: 40	Prácticas: 40	Estudio Independiente: 48		
	Total de horas: 128		Créditos: 8		
Competencia(s) del perfil de egreso al que aporta:	<p><b>Genéricas:</b></p> <p>CG1. Desarrolla su potencial intelectual para generar el conocimiento necesario en la resolución de problemas y retos, tanto de su vida individual y como parte de una comunidad, con sentido de pertinencia, identidad y empatía.</p> <p>CG10. Asume con responsabilidad y ética el manejo de las Tecnologías del Aprendizaje y el Conocimiento y es capaz de re conducir las Tecnologías de la Información y Comunicación para la adquisición y actualización del conocimiento de manera permanente para su vida y su profesión.</p> <p><b>Específicas:</b></p> <p>CE5. Demuestra conocimiento, comprensión y capacidad de evaluar la estructura y la arquitectura de las computadoras, así como los componentes básicos que los conforman.</p> <p>CE11. Construye las piezas de software que hacen posible el funcionamiento de las computadoras en diferentes niveles operativos.</p>				
Unidades de aprendizaje relacionadas:	Estructura de datos, algoritmos y lógica computacional, teoría de la computación.				
Responsable(s) de elaborar el programa:	L.S.C. ALAN D. RAMIREZ NORIEGA		Fecha: 05 de Julio de 2011		
Responsable(s) de actualizar el programa:			Fecha:		
2. PROPÓSITO					
Identificar los diferentes tipos de software de sistemas existentes, y se enfoque principalmente en la comprensión de las fases de un compilador aplicando conocimientos sobre expresiones regulares y autómatas para potenciar el desarrollo de la lógica de programación.					
3. SABERES					



### PROGRAMA DE ESTUDIO

Teóricos:	El alumno complementará los conocimientos sobre programación de aplicaciones ya adquiridos, comprenderá qué es y cómo funciona un compilador, adquirirá la capacidad para diseñar un compilador en sus primeras fases para un lenguaje especificado. Se analizará el concepto de software de sistemas, comparado con los de aplicación, el alumno identificará los tipos de software de sistemas existentes. El alumno entenderá el funcionamiento genérico de un compilador, identificando cada fase del mismo. El alumno conocerá y aplicará las técnicas para elaborar un analizador léxico. El alumno conocerá las ventajas y beneficios de la memoria dinámica, aplicará las técnicas de recursividad y apuntadores para resolver problemas propios del compilador. El alumno adquirirá los conocimientos básicos para elaborar la fase sintáctica de un compilador. El alumno desarrollara un analizador sintáctico del tipo descendente, aplicando recursividad, apuntadores y árboles.
Prácticos:	Conocimiento y aplicación de expresiones regulares. Conocimiento y aplicación de autómatas. Creación de un compilador en sus 3 primeras fases: léxico, sintáctico y semántico.
Actitudinales:	Resolución de problemas, lógica, trabajo en equipo, planeación y organización del proyecto.

### 4. CONTENIDOS

#### Unidad 1 Introducción a los compiladores.

- 1.1 Qué es y cómo funciona un compilador
- 1.2 Concepto de software de sistemas
- 1.3 Comparativa entre software de sistemas y de aplicación
- 1.4 Tipos de software de sistemas existentes
- 1.5 Funcionamiento genérico de un compilador
- 1.6 Fases del compilador

#### Unidad 2 Analizador léxico.

- 2.1 Analizador léxico
- 2.2 Conocimiento y aplicación de expresiones regulares

#### Unidad 3 Analizador sintáctico

- 3.1 Gramáticas libres de contexto
- 3.2 Analizador sintáctico descendente, LL, LR

#### Unidad 4 Analizador semántico

- 4.1 Creación de un compilador en sus 3 primeras fases: léxico, sintáctico y semántico

#### Unidad 5 Memoria dinámica.

#### Unidad 6 Tabla de símbolos.

### 5. ACTIVIDADES PARA DESARROLLAR LAS COMPETENCIAS

#### Actividades del docente:

- Explicaciones del tema en clase.
- Prácticas en centro de cómputo. trabajo extra clase.



## PROGRAMA DE ESTUDIO

- Revisión de avances del proyecto.

*Actividades del estudiante:*

- ❖ Trabajo en equipo.
- ❖ Investigación.
- ❖ Asistencia.
- ❖ Participación.
- ❖ Cumplimiento de prácticas y tareas.
- ❖ Desarrollo de un proyecto (compilador).

**6. EVALUACIÓN DE LAS COMPETENCIAS**

6.1. Criterios de desempeño	6.2 Portafolio de evidencias
<ul style="list-style-type: none"><li>• Correcta aplicación de los algoritmos de la materia.</li><li>• Calidad de la investigación.</li><li>• Adecuado uso de los contenidos.</li><li>• Avances de acuerdo al cronograma.</li></ul>	<ul style="list-style-type: none"><li>• Tareas.</li><li>• Prácticas.</li><li>• Examen.</li><li>• Avances del proyecto.</li></ul>
<b>6.3. Calificación y acreditación:</b>	
Parcial: <ul style="list-style-type: none"><li>• Prácticas y tareas 20%</li><li>• Examen 30</li><li>• Proyecto 50%</li></ul>	Final: <ul style="list-style-type: none"><li>• Promedio de las evaluaciones parciales.</li></ul>

**7. RECURSOS DIDÁCTICOS**

<b>8. FUENTES DE INFORMACIÓN</b>				
<i>Bibliografía básica</i>				
Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
Sergio Gálvez Rojas	Traductores y Compiladores con Lex/Yacc, JFlex/Cup y JavaCC	ISBN,8468910376, 9788468910376		



UNIVERSIDAD AUTÓNOMA DE SINALOA

FACULTAD DE INGENIERÍA MOCHIS  
LICENCIATURA EN INGENIERÍA DE SOFTWARE



PROGRAMA DE ESTUDIO

Anthony A. Aaby,	Compiler Construction using Flex and Bison	Walla Walla College cs.wwc.edu, aabyan@wwc.edu.	Version of April 22, 2005	

*Bibliografía complementaria*

Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
Aho A., Sethi R., Ullman J	Compiladores: Principios, técnicas y herramientas,	Addisson-Wesley	(1986). (Unidad 2)	
Kenneth C. Louden	Construcción de Compiladores, principios y prácticas	Thomson	2004	

9. PERFIL DEL DOCENTE

LIC. EN INFORMÁTICA. LIC. EN SISTEMAS COMPUTACIONALES. ING. EN SOFTWARE. CARRERA AFIN A LAS ANTERIORES