



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



**PROGRAMA SINTÉTICO**

**UNIDAD ACADÉMICA:** Escuela Superior de Cómputo

**PROGRAMA** Ingeniería en Sistemas Computacionales

**ACADÉMICO:**

**UNIDAD DE APRENDIZAJE:** Compiladores

**NIVEL:** III

**PROPÓSITO DE LA UNIDAD DE APRENDIZAJE:**

Desarrolla un intérprete con base en herramientas generadoras de analizadores léxicos y sintácticos y programación orientada a objetos.

**CONTENIDOS:**

- I. Arquitectura de los Compiladores e Interpretes
- II. Construcción de Analizadores Léxicos
- III. Construcción de Analizadores Sintácticos
- IV. Análisis semántico y Generación de código intermedio
- V. Optimización de código

**ORIENTACIÓN DIDÁCTICA:**

Se utilizará la estrategia de enseñanza POL (aprendizaje orientado a proyectos). El docente conducirá el curso mediante el método de enseñanza deductivo, en donde se requiere la participación activa de los estudiantes, auxiliándose de las técnicas de aprendizaje en las que el alumno sea el actor principal, tales como: Lluvia de ideas, Fichas de trabajo, Indagación documental, Exposición de temas, Análisis y discusión meta cognitiva, Resolución de problemas, Elaboración de mapas conceptuales y desarrollo de prácticas.

**EVALUACIÓN Y ACREDITACIÓN:**

La presente Unidad de Aprendizaje se evaluará a partir del portafolio de evidencias, el cual se conforma de: evaluación formativa, sumativa y rubricas de autoevaluación y coevaluación.

Esta unidad de aprendizaje también se puede acreditar mediante:

- Evaluación de saberes previamente adquiridos.
- Acreditación en otra UA del IPN u otra institución educativa nacional o internacional.

**BIBLIOGRAFÍA:**

- Aho, A. V., Sethi, R., y Ullman, J. D. (2008). *Compiladores: Principios, Técnicas y Herramienta*, (2ª Edición). México: Pearson-Addison Wesley. ISBN: 978-970-26-1133-2.
- Alfonseca, M. (2006). *Compiladores e intérpretes: Teoría y práctica*. (1ª Edición). México: Pearson Alhambra. ISBN: 8420550310.
- Grune, D., E. Bal. H. (2007). *Diseño de Compiladores Modernos*. (1ª Edición). España: Mc Graw Hill. ISBN: 978-84-481-5656-5.
- I. Holub Allen. (1990). *Compiler Design in C*. Prentice Hall. (2ª Edición). E.U.A: Prentice Hall. ISBN: 0131550454.
- Ruiz, J. (2010). *Compiladores: Teoría e Implementación*. México: Alfaomega Grupo Editor. ISBN 978-607-7854-68-5.



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



**UNIDAD ACADÉMICA:** Escuela Superior de  
Cómputo  
**PROGRAMA ACADÉMICO:** Ingeniería en Sistemas  
Computacionales  
**SALIDA LATERAL:** Analista Programador de  
Sistemas de Información  
**ÁREA DE FORMACIÓN:** Profesional  
**MODALIDAD:** Presencial.

**UNIDAD DE APRENDIZAJE:** Compiladores  
**TIPO DE UNIDAD DE APRENDIZAJE:** Teórico - Práctica,  
Obligatoria  
**VIGENCIA:** 2011  
**NIVEL:** III  
**CRÉDITOS:** TEPI. 7.5 - SATCA. 4.39

**INTENCIÓN EDUCATIVA**

Aplicar los conocimientos de algoritmia, programación y teoría computacional en el diseño y desarrollo de algoritmos de análisis léxico y sintáctico; así como integrar estos conocimientos en la construcción de un intérprete para un lenguaje de programación específico.

Las competencias a desarrollar por medio de esta unidad de aprendizaje son :

Diseñar e implementar por medio de la programación orientada a objetos analizadores léxicos, analizadores sintácticos por descenso recursivo, analizadores sintácticos ascendentes y algoritmos para la generación de código intermedio.

Las competencias desarrolladas en ésta unidad de aprendizaje permiten que el alumno tener una formación sólida en el diseño e implementación de algoritmos.

Diseñar gramáticas para un propósito específico y Aplicar el concepto de atributos heredados y sintetizados para la construcción de sistemas que requieran la aplicación de analizadores sintácticos: Evaluadores de expresiones, graficadores, derivadores simbólicos, procesadores de consultas, etc. Comparar alternativas de implementación de algoritmos de análisis léxico, sintáctico y semántico.

Así mismo, se desarrolla el pensamiento estratégico, el pensamiento creativo, el trabajo colaborativo y participativo y la comunicación asertiva.

Esta unidad de aprendizaje tiene como antecedente las unidades de aprendizaje de Algoritmia y Programación Estructurada, Matemáticas Discretas, Teoría Computacional, Programación Orientada a Objetos, Análisis y Diseño Orientado a Objetos. Horizontalmente se relaciona con Análisis de Algoritmos.

**PROPÓSITO DE LA UNIDAD DE APRENDIZAJE**

Desarrolla un intérprete con base en herramientas generadoras de analizadores léxicos y sintácticos y programación orientada a objetos.

**TIEMPOS ASIGNADOS**

**HORAS TEORÍA/SEMANA:** 3.0

**HORAS PRÁCTICA/SEMANA:** 1.5

**HORAS TEORÍA/SEMESTRE:** 54

**HORAS PRÁCTICA/SEMESTRE:** 27

**HORAS DE APRENDIZAJE  
AUTÓNOMO:** 54

**HORAS TOTALES/SEMESTRE:** 81

**UNIDAD DE APRENDIZAJE DISEÑADA**  
Academia de Proyectos Estratégicos y  
toma de decisiones

**REVISADA POR:**

Dr. Flavio A. Sánchez Garfías  
Subdirección Académica

**APROBADA POR:**

Ing. Apolinar Fco. Cruz Lázaro  
presidente del  
Consejo Técnico Consultivo Escolar.

**AUTORIZADO POR:** Comisión de  
Programas Académicos del Consejo  
General Consultivo del IPN.

Ing. Rodrigo de Jesús Serrano  
Domínguez  
Secretario Técnico de la Comisión  
de Programas Académicos

**HOJA: 3 DE 10**

N° UNIDAD TEMÁTICA: I		NOMBRE: Arquitectura de los compiladores e intérpretes				
UNIDAD DE COMPETENCIA						
Determina la estructura general de los compiladores e intérpretes con base en la funcionalidad de sus etapas y fases.						
No.	CONTENIDOS	HORAS AD Con Docente (a)		HORAS Aprendizaje Autónomo (b)		CLAVE BIBLIOGRÁFICA
		T	P	T	P	
1.1	Definición de compiladores e intérpretes	0.5		1.0		B1 y B3
1.2	Arquitectura de los compiladores e intérpretes	0.5		1.0		
1.3	Máquinas virtuales	0.5		1.0		
1.4	Funcionalidad de las fases de un compilador	1.5		3.0		
1.4.1	Análisis léxico					
1.4.2	Análisis Sintáctico					
1.4.3	Análisis Semántico					
1.4.4	Generación de código intermedio					
1.4.5	Optimización de código					
1.4.6	Generación de código objeto					
	Subtotales:	3.0		6.0		
ESTRATEGIAS DE APRENDIZAJE						
Encuadre del curso y formación de equipos.						
La presente unidad se abordará a partir de la estrategia de aprendizaje orientada a proyectos y método deductivo, lo que permitirá la consolidación de las siguientes técnicas de aprendizaje: lluvia de ideas, ficha de trabajo, indagación documental, discusión dirigida, mapas conceptuales, resolución de problemas y exposición en equipo de temas.						
EVALUACIÓN DE LOS APRENDIZAJES						
Evaluación diagnóstica						
Portafolio de evidencias:						
Ficha de trabajo 5%						
Mapa conceptual 5%						
Exposición en equipo 20%						
Propuesta de proyecto 15%						
Rúbricas de autoevaluación 2%						
Rúbrica de coevaluación 3%						
Evidencia de aprendizaje 50%						



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



**UNIDAD DE APRENDIZAJE:** Compiladores.

**HOJA:** 4 **DE** 10

N° UNIDAD TEMÁTICA: II		NOMBRE: Construcción de analizadores léxicos.				
UNIDAD DE COMPETENCIA						
Implementa el proceso de análisis léxico, con base en autómatas finitos deterministas y no deterministas.						
No.	CONTENIDOS	HORAS AD Con docente (a)		HORAS Aprendizaje Autónomo (b)		CLAVE BIBLIOGRÁFICA
		T	P	T	P	
2.1	Definición de clases léxicas por medio de expresiones regulares.	0.5		1.0		1B y 4B
2.2	Creación de AFN (Autómatas finitos no deterministas) por medio de la construcción de Thompson	1.0	0.5	1.5	1.0	
2.3	Conversión de AFN a AFD (Autómatas finitos deterministas)	1.0	0.5	1.5	1.0	
2.4	Creación de un AFD a partir una expresión regular	0.5	0.5	1.0	1.0	
2.5	Minimización de estados de un AFD	0.5		1.5		
2.6	Construcción de analizadores léxicos definidos por medio de expresiones regulares	0.5	1.0	1.5	2.0	
2.7	Generadores de analizadores léxicos (FLEX, JFLEX, JACCIE, GPLEX, etc.)			1.0		
	Subtotales:	4.0	2.5	9.0	5.0	

**ESTRATEGIAS DE APRENDIZAJE**

La presente unidad se abordará a partir de la estrategia aprendizaje orientada a proyectos y método deductivo, lo que permitirá la consolidación de las siguientes técnicas de aprendizaje: indagación documental, ficha de trabajo, discusión dirigida, resolución de ejercicios, exposición en equipo de temas complementarios, propuesta del proyecto y realización de prácticas 1, 2, 3 y 4.

**EVALUACIÓN DE LOS APRENDIZAJES**

Portafolio de evidencias:

Ficha de trabajo	5%
Problemario	5%
Exposición en equipo	10%
Reportes de práctica	20%
Avance de proyecto	10%
Rúbricas de autoevaluación	5%
Rúbrica de coevaluación	5%
Evidencia de aprendizaje	40%



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



**UNIDAD DE APRENDIZAJE:** Compiladores

**HOJA:** 5 **DE** 10

N° UNIDAD TEMÁTICA: III		NOMBRE: Construcción de analizadores sintácticos				
UNIDAD DE COMPETENCIA						
Implementa analizadores sintácticos, con base en gramáticas libres de contexto y herramientas automatizadas.						
No.	CONTENIDOS	HORAS Con docente (a)		HORAS Aprendizaje Autónomo (b)		CLAVE BIBLIOGRÁFICA
		T	P	T	P	
3.1	Analizadores sintácticos	1.0		2.0		1B, 2B, 3B y 6C
3.1.1	Gramáticas ambiguas					
3.1.2	Eliminación de recursión izquierda en las gramáticas					
	Árboles de sintaxis					
3.1.3	Definición de lenguajes libres del contexto					
3.1.4	Analizadores Sintácticos Descendentes					
3.2	Construcción de analizadores sintácticos por descenso	1.0	1.0	2.0	2.0	
3.2.1	recursivo					
3.2.2	Construcción de analizadores sintácticos LL(1)					
3.3	Analizadores Sintácticos Ascendentes	2.0	1.5	3.0	3.0	
3.3.1	Analizadores LR(0)					
3.3.2	Analizadores LR(1)					
3.3.3	Analizadores LALR					
3.4	Generadores de analizadores sintácticos (YACC, JACC, JACCIE, GPPG)	1.0	1.0	2.0	2.0	
	Subtotales:	5.0	3.5	9.0	7.0	
ESTRATEGIAS DE APRENDIZAJE						
La presente unidad se abordará a partir de la estrategia aprendizaje orientada a proyectos y método deductivo, lo que permitirá la consolidación de las siguientes técnicas de aprendizaje: indagación documental, ficha de trabajo, discusión dirigida, cuadro de comparaciones, exposición en equipo de temas complementarios, propuesta del proyecto y realización de prácticas 5, 6 y 7.						
EVALUACIÓN DE LOS APRENDIZAJES						
Portafolio de evidencias:						
Ficha de trabajo		5%				
Cuadro comparativo		5%				
Prácticas con reportes		20%				
Exposición en equipo		10%				
Avance de proyecto		20%				
Rúbricas de autoevaluación		5%				
Rúbrica de coevaluación		5%				
Evidencia de aprendizaje		30%				



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



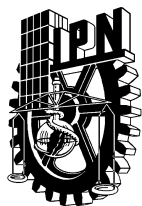
UNIDAD DE APRENDIZAJE: Compiladores

HOJA: 6 DE 10

N° UNIDAD TEMÁTICA: IV		NOMBRE: Análisis semántico y generación de código intermedio					
UNIDAD DE COMPETENCIA							
Implementa una aplicación de análisis semántico con base en atributos sintetizados y heredados.							
No.	CONTENIDOS	HORAS AD Actividades de docencia (a)		HORAS TAA Actividades de Aprendizaje Autónomo (b)		CLAVE BIBLIOGRÁFICA	
		T	P	T	P		
4.1	Atributos	0.5		2.0		1B, 2B, 5B, 4C y 6C	
4.1.1	Atributos heredados						
4.1.2	Atributos sintetizados						
4.2	Diseño de tabla de símbolos para soportar llamadas a funciones y/o procedimientos y ámbito de variables.	0.5		2.0			
4.3	Generación de código de 3 y 4 direcciones	0.5					
4.4	Generación de código intermedio para estructuras iterativas y de control (for, while, if-else, switch)	0.5	1.0	2.0	2.0		
4.5	Generación de código intermedio para el llamado a funciones y procedimientos	0.5	1.0		2.0		
4.6	Manejo del Polimorfismo en un compilador	0.5					
	Subtotales:	3.0	2.0	6.0	4.0		
ESTRATEGIAS DE APRENDIZAJE							
La presente unidad se abordará a partir de la estrategia aprendizaje orientada a proyectos y método deductivo, lo que permitirá la consolidación de las siguientes técnicas de aprendizaje: indagación documental, ficha de trabajo, discusión dirigida, exposición en equipo de temas complementarios, propuesta del proyecto y realización de prácticas 8 y 9.							
EVALUACIÓN DE LOS APRENDIZAJES							
Portafolio de evidencias:							
Ficha de trabajo		5%					
Exposición en equipo		15%					
Prácticas con reportes		20%					
Avance de proyecto		30%					
Rúbricas de autoevaluación		5%					
Rúbrica de coevaluación		5%					
Evidencia de aprendizaje		20%					

UNIDAD DE APRENDIZAJE: Compiladores

HOJA: 7 DE 10



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



**N° UNIDAD TEMÁTICA:** V

**NOMBRE:** Optimización de código

**UNIDAD DE COMPETENCIA**

Implementa los generadores de código en el proceso de construcción de compiladores, con base en algoritmos de optimización.

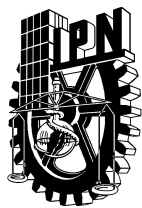
No.	CONTENIDOS	HORAS AD Con docente (a)		HORAS Aprendizaje Autónomo (b)		CLAVE BIBLIOGRÁFICA
		T	P	T	P	
5.1	Introducción a la optimización de código	0.5	1.0	1.0	2.0	1B y 2B
5.2	Análisis de flujo de datos	0.5		1.0		
5.3	Propagación de constantes	0.5		1.0		
5.4	Eliminación de redundancia parcial	0.5		1.0		
5.5	Ciclos en grafos de flujos	0.5		1.0		
5.6	Análisis basado en regiones	0.5		1.0		
Subtotales:		3.0	1.0	6.0	2.0	

**ESTRATEGIAS DE APRENDIZAJE**

La presente unidad se abordará a partir de la estrategia aprendizaje orientada a proyectos y método deductivo, lo que permitirá la consolidación de las siguientes técnicas de aprendizaje: discusión dirigida, exposición por equipos, implementación del proyecto y realización de la práctica 10.

**EVALUACIÓN DE LOS APRENDIZAJES**

Portafolio de evidencias:	
Prácticas con reportes	20%
Exposición en equipo	10%
Implementación del proyecto	60%
Rúbricas de autoevaluación	5%
Rúbrica de coevaluación	5%



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



**UNIDAD DE APRENDIZAJE:** Compiladores

**HOJA:** 8 **DE** 10

PRÁCTICA No.	NOMBRE DE LA PRÁCTICA	UNIDADES TEMÁTICAS	DURACIÓN	LUGAR DE REALIZACIÓN
1	Diseño e implementación de las clases AFN, AFD.	II	1.5	Laboratorio de cómputo
2	Diseño y desarrollo de un sistema para obtener el AFN asociado a una Expresión Regular por construcción de Thompson	II	1.5	
3	Implementación del algoritmo para convertir un AFN a AFD	II	1.5	
4	Desarrollo de un analizador léxico especificado por medio de expresiones regulares.	II	3.0	
5	Desarrollo de un analizador sintáctico por descenso recursivo.	III	3.0	
6	Diseño e implementación de la clase para obtener la tabla LL(1) de una gramática libre del contexto sin recursión izquierda.	III	4.5	
7	Diseño e implementación de la clase para obtener las tablas LR(0), LR(1), LALR.	III	3.0	
8	Generación de un analizador sintáctico ascendente para la gramática de un lenguaje de programación que soporte: Declaración de variables, estructuras de control, estructuras iterativas, expresiones relacionales y aritméticas, llamadas a funciones y procedimientos (Sugerencia HOC).	IV	3.0	
9	Generación de código intermedio para la gramática del problema anterior.	IV	3.0	
10	Diseño e implementación de algoritmos para optimización de código aplicados al código generado en la práctica 9	V	3.0	
		<b>TOTAL DE HORAS</b>	<b>27</b>	

**EVALUACIÓN Y ACREDITACIÓN:**

Las prácticas aportan el 20% de la calificación de las unidades temáticas II a la V.

Las prácticas se consideran requisito indispensable para acreditar esta unidad de aprendizaje.





**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



**UNIDAD DE APRENDIZAJE:** Compiladores

**HOJA:** 9 **DE** 10

PERÍODO	UNIDAD	PROCEDIMIENTO DE EVALUACIÓN	
1	I	Evaluación continua	50%
		Evidencia de aprendizaje	50%
	II	Evaluación continua	60%
		Evidencia de aprendizaje	40%
2	III	Evaluación continua	70%
		Evidencia de aprendizaje	30%
3	IV	Evaluación continua	80%
		Evidencia de aprendizaje	20%
	V	Evaluación continua	100%
Las unidades I y II aportan el 30% de la calificación final. La unidad III aporta el 30% de la calificación final. Las unidades IV y V aportan el 40% de la calificación final.			
Esta unidad de aprendizaje también puede ser acreditada mediante:			
<ul style="list-style-type: none"><li>• Evaluación de saberes previamente adquiridos.</li><li>• Acreditación en otra UA del IPN u otra institución educativa nacional o internacional.</li></ul>			

CLAVE	B	C	BIBLIOGRAFÍA
1	X		Aho, A. V., Sethi, R., y Ullman, J. D. (2008) <i>Compiladores: Principios, Técnicas y Herramienta</i> , (2ª Edición). México: Pearson-Addison Wesley. ISBN: 978-970-26-1133-2
2	X		Alfonseca, Manuel. (2006). <i>Compiladores e intérpretes: Teoría y práctica</i> . (1ª Edición) Pearson Alhambra. ISBN: 8420550310
3	X		Grune, Dick., E. Bal. Henri. (2007). <i>Diseño de Compiladores Modernos</i> . (1ª Edición). España: Mc Graw Hill. ISBN: 978-84-481-5656-5
4		X	I. Holub Allen. (1990). <i>Compiler Design in C</i> . Prentice Hall (2ª Edición). ISBN: 0131550454
5	X		Ruiz, Jacinto. (2010). <i>Compiladores: Teoría e Implementación</i> . México: Alfaomega Grupo Editor. ISBN 978-607-7854-68-5
6		X	W. Kernighan Brian, Pike Rob. (1987.) <i>El Entorno de Programación Unix</i> . Editorial Prentice Hall. ISBN: 958-880-067-8



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA ACADÉMICA**  
**DIRECCIÓN DE EDUCACIÓN SUPERIOR**



**PERFIL DOCENTE POR UNIDAD DE APRENDIZAJE**

**1. DATOS GENERALES**

**UNIDAD ACADÉMICA:** Escuela Superior de Computo

**PROGRAMA**

**ACADÉMICO:** Ingeniería en Sistemas Computacionales **NIVEL** III

**ÁREA DE FORMACIÓN:**

<b>Institucional</b>	<b>Científica Básica</b>	<b>Profesional</b>	<b>Terminal y de Integración</b>
----------------------	------------------------------	--------------------	--------------------------------------

**ACADEMIA:** Ciencias de la computación

**UNIDAD DE APRENDIZAJE:** Compiladores

**ESPECIALIDAD Y NIVEL ACADÉMICO REQUERIDO:** Maestría en Ciencias de la Computación

**2. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE:**

Desarrolla un intérprete con base en herramientas generadoras de analizadores léxicos y sintácticos y programación orientada a objetos.

**3. PERFIL DOCENTE:**

CONOCIMIENTOS	EXPERIENCIA PROFESIONAL	HABILIDADES	ACTITUDES
<ul style="list-style-type: none"><li>Conocer el Modelo Educativo Institucional del IPN.</li><li>Teoría computacional, teoría de lenguajes de programación y de compiladores.</li></ul>	<ul style="list-style-type: none"><li>Experiencia de un año implementando estructuras de datos y en el área de teoría de lenguajes de programación.</li><li>Experiencia de un año en implementación programas de cómputo.</li><li>Experiencia mínima de un año en manejo de grupos y trabajo colaborativo.</li><li>Experiencia mínima de un año como Docente de Nivel Superior.</li></ul>	<ul style="list-style-type: none"><li>Expresarse correctamente de forma oral y escrita.</li><li>Revisar programas en un lenguaje de Programación Orientado a Objetos.</li><li>Integrar la ingeniería de software en el desarrollo de sistemas computacionales.</li><li>Aplicar el Modelo Educativo de la Institucional del IPN.</li></ul>	<ul style="list-style-type: none"><li>Paciente</li><li>Creativo</li><li>Responsable</li><li>Tolerante</li><li>Honesto</li><li>Respetuoso</li><li>Comprometido socialmente e institucionalmente</li></ul>

**ELABORÓ**

**REVISÓ**

**AUTORIZÓ**

Lic. Andrés Ortigoza Campos  
M. en C. Luz María Sánchez García  
Profesores Coordinadores

Dr. Flavio Arturo Sánchez Garfías  
Subdirector académico

Ing. Apolinar Francisco Cruz Lázaro  
Director

Fecha: 2011