

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

COORDINACIÓN GENERAL DE FORMACIÓN BÁSICA
COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA
PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

- 1. Unidad Académica:** Facultad de Ingeniería, Mexicali; Facultad de Ingeniería, Arquitectura y Diseño, Ensenada; y Facultad de Ciencias Químicas e Ingeniería, Tijuana.
- 2. Programa Educativo:** Ingeniero en Computación
- 3. Plan de Estudios:** 2020-1
- 4. Nombre de la Unidad de Aprendizaje:** Traductores
- 5. Clave:** 36310
- 6. HC:** 02 **HL:** 02 **HT:** 00 **HPC:** 00 **HCL:** 00 **HE:** 02 **CR:** 06
- 7. Etapa de Formación a la que Pertenece:** Disciplinaria
- 8. Carácter de la Unidad de Aprendizaje:** Optativa
- 9. Requisitos para Cursar la Unidad de Aprendizaje:** Ninguno



Equipo de diseño de PUA

Víctor Rafael Nazario Velázquez Mejía
Haydeé Meléndez Guillén
Guillermo Licea Sandoval

Vo.Bo. de Subdirectores de Unidades Académicas

Alejandro Mungaray Moctezuma
Humberto Cervantes de Ávila
Rocío Alejandra Chávez Santoscoy

Fecha: 17 de octubre de 2019

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje tiene como finalidad que el alumno comprenda los principios, técnicas y herramientas para la construcción de traductores (compiladores), para aplicarlos en la construcción de un traductor que dé solución a problemas reales relacionados con las áreas traductores y compiladores.

La unidad de aprendizaje de Traductores pertenece a la etapa terminal, es optativa y forma parte del área de conocimiento Ciencias de la Ingeniería el programa educativo de Ingeniero en Computación y no precisa de requisitos previos para cursarla; aunque es deseable tener conocimientos sobre algoritmos y estructura de datos, así como la habilidad para elaborar reportes técnicos y capacidad para el trabajo en equipo.

III. COMPETENCIA DE LA UNIDAD DE APRENDIZAJE

Diseñar un traductor, aplicando los principios, técnicas, herramientas y el proceso de construcción de traductores, para su implementación en problemas que requieran el proceso del traductor, mostrando una actitud reflexiva y propositiva.

IV. EVIDENCIA(S) DE DESEMPEÑO

Elabora prototipo de un traductor que incluya un reporte técnico donde se muestre la aplicación correcta de un lenguaje y una gramática independiente al contexto, para resolver un problema real.

V. DESARROLLO POR UNIDAD

UNIDAD I. Construcción de compiladores

Competencia:

Identificar los conceptos fundamentales del proceso de compilación y traducción, con base al estudio de la estructura de los traductores y su elaboración, para relacionarlos y aplicarlos en situaciones reales que requieran de una solución del proceso de compilación, con una actitud crítica y reflexiva.

Contenido:**Duración:** 4 horas

- 1.1. Introducción a los sistemas de traducción.
- 1.2. Conceptos fundamentales de traductores.
- 1.3. Estructura de un traductor.
 - 1.3.1. Análisis Léxico
 - 1.3.2. Análisis Sintáctico
 - 1.3.3. Análisis Semántico
 - 1.3.4. Generación de código Intermedio
 - 1.3.5. Optimización de Código
 - 1.3.6. Generación de Código
 - 1.3.7. Manejador de errores
- 1.4. Proceso para elaborar un traductor

UNIDAD II. Análisis léxico

Competencia:

Examinar el procedimiento para analizar textos, mediante el estudio de los componentes que integran al analizador léxico, para su implementación en la traducción de un lenguaje fuente a mínimas de lenguaje, mostrando una actitud analítica y reflexiva.

Contenido:

Duración: 10 horas

- 2.1. Fundamentos del análisis léxico.
- 2.2. Función del análisis léxico.
- 2.3. Componentes léxicos, patrones y lexema.
- 2.4. Expresiones regulares.
- 2.5. Autómatas finitos.
- 2.6. Patrones para el análisis lexicográfico.
- 2.7. Manejo de errores.
- 2.8. Herramientas para la generación de analizadores lexicográficos.

UNIDAD III. Análisis sintáctico y semántico

Competencia:

Aplicar las técnicas para modelar la sintaxis de una gramática libre de contexto, mediante el uso adecuado de los componentes de los analizadores sintáctico y semántico, para construir un traductor que reconozca frases y recupere errores en ambas fases, con actitud analítica y propositiva.

Contenido:**Duración:** 10 horas

- 3.1. Fundamentos del análisis sintáctico y semántico.
- 3.2. Función del análisis sintáctico y semántico.
- 3.3. Gramáticas libres de contexto.
- 3.4. Árboles de sintaxis.
- 3.5. Parser recursive descendente.
- 3.6. Patrones para el análisis sintáctico.
- 3.7. Manejo de errores sintácticos.
- 3.8. Herramientas para la generación de analizadores sintácticos.
- 3.9 Análisis semántico.
- 3.10. Verificación de errores semánticos.

UNIDAD IV. Traductor

Competencia:

Aplicar las técnicas para modelar la sintaxis de la gramática, a través del análisis de los tipos de alcances de las tablas de símbolos y sus patrones de construcción, para construir un analizador sintáctico de reconocimiento de frases, con una actitud reflexiva y propositiva.

Contenido:**Duración:** 4 horas

- 4.1. Definición y conceptos.
- 4.2. Tipos de alcances de las tablas de símbolos.
- 4.3. Patrones para la construcción de tablas de símbolos.

UNIDAD V. Interpretación.

Competencia:

Implementar las formas en que se puede representar explícitamente un programa fuente, mediante el estudio de los diferentes intérpretes, para generar de forma óptima un código objeto, con una actitud reflexiva y propositiva.

Contenido:

Duración: 4 horas

- 5.1. Definición y conceptos.
- 5.2. Intérpretes dirigidos por sintaxis.
- 5.3. Intérpretes basados en árboles.
- 5.3. Intérpretes de Código de bytes (bytecode).

VI. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

No. de Práctica	Competencia	Descripción	Material de Apoyo	Duración
UNIDAD I				
1	<p>Analizar una problemática real que requiera la aplicación de un sistema de traducción, mediante investigación documental o presencial, para proponer el prototipo que dará solución a esta, con creatividad y actitud propositiva.</p>	<p>Se elabora una propuesta de proyecto con la descripción de un problema real a resolver y que requiere el uso de un sistema de traducción. La propuesta debe contener antecedentes, justificación, objetivos generales y específicos, metodología y plan de trabajo. Se otorga retroalimentación sobre las oportunidades de mejora.</p>	<p>Equipo de cómputo con conexión a internet y procesador de textos.</p>	2 horas
	<p>Esquematizar un prototipo que solucione una problemática que requiera de un sistema de traducción, mediante la creación de las interfaces del software, para validar que se ha comprendido el problema, con creatividad.</p>	<p>Se realiza un reporte que contenga el diseño de las interfaces gráfica del traductor y las interacciones entre estas y el usuario. El diseño de las interfaces propuesta debe cumplir con el proceso de traducción establecido para el proyecto:</p> <ul style="list-style-type: none"> -Analizador léxico. -Analizador sintáctico. -Analizador semántico. -Traductor. 	<p>Equipo de cómputo con conexión a internet, procesador de textos y software para diseño gráfico.</p>	2 horas
UNIDAD II				
2	<p>Diseñar un analizador léxico, mediante la identificación de los componentes del nuevo lenguaje y la gramática libre de contexto, para procesar un archivo</p>	<p>Se realiza un reporte que contenga los componentes de símbolos, unidades léxicas y los tokens, para posteriormente</p>	<p>Equipo de cómputo con conexión a internet y procesador de textos.</p>	2 horas

	<p>de entrada escrito en el nuevo lenguaje y establecer los componentes de entrada del analizador sintáctico, con innovación y una actitud reflexiva.</p>	<p>especificar el lenguaje a ser creado.</p> <p>Se revisa que los símbolos, las unidades léxicas y los tokens sean acordes con el contexto de la problemática. Se retroalimenta respecto a las oportunidades de mejora.</p>		
	<p>Se especifica el lenguaje y la gramática libre de contexto para la solución propuesta.</p> <p>Se determinan los elementos terminales y no terminales para la gramática.</p> <p>Se revisa que el lenguaje y la gramática libre de contexto cumplan con los componentes necesarios para alcanzar los elementos terminales y no terminales.</p>	<p>Equipo de cómputo con conexión a internet y procesador de textos.</p>		4 horas
	<p>Se diseña un analizador léxico que considere como entrada un programa escrito con el nuevo lenguaje creado para el proyecto y la tabla de símbolos que lo representa; que proporcione como salida una los tokens y la tabla de símbolos con base al análisis léxico. El analizador léxico deberá ser capaz de reportar los errores encontrados, así como el tipo de error encontrado.</p> <p>Se revisa que el analizador léxico funcione correctamente,</p>	<p>Equipo de cómputo con conexión a internet, procesador de textos y un IDE para desarrollo de software.</p>		4 horas

		introduciendo un programa escrito con el nuevo lenguaje para el proyecto; verifica que el reporte con la lista de mensajes tanto de éxitos como de errores sean correctos con base la gramática establecida.		
UNIDAD III				
3	Diseñar un analizador sintáctico, mediante la investigación documental de técnicas para el análisis sintáctico, de la aplicación de técnicas de modelado de sintaxis a los componentes resultantes del analizador léxico, para validar la correcta escritura del archivo de entrada y establecer los componentes de entrada del analizador semántico, con creatividad.	Se realiza un reporte que contenga el análisis de los componentes de verificación de la sintaxis del lenguaje a través de la propuesta de autómatas. Se revisa que los autómatas cumplan con la sintaxis para cada gramática y a través de una retroalimentación al alumno presenta oportunidades de mejora.	Equipo de cómputo con conexión a internet y procesador de textos.	2 horas
		Se construye un programa de cómputo que evalúa la sintaxis de los autómatas diseñados para el nuevo lenguaje, tomando como base los tokens generados en la etapa de análisis léxico, y genera la salida para el traductor. El analizador léxico deberá ser capaz de reportar los errores encontrados, así como el tipo de error encontrado. Se revisa que el programa de cómputo evalúe correctamente la sintaxis para cada gramática.	Equipo de cómputo con conexión a internet, procesador de textos y un IDE para desarrollo de software.	4 horas
	Diseñar un analizador semántico,	Se diseña un programa de	Equipo de cómputo con	4 horas

	mediante investigación documental de técnicas del análisis semántico, para validar el correcto significado de los tipos de datos y establecer los componentes de entrada del traductor, con creatividad.	cómputo para realizar el análisis semántico con el fin de verificar los tipos de datos, como entrada tendrá los tokens generados, aceptados y validados en la etapa del analizador sintáctico, como salida deberá producir los mensajes de error en el uso de tipos de datos o éxito del proceso. Se revisa que el programa de cómputo evalúe correctamente la semántica del texto de entrada.	conexión a internet, procesador de textos y un IDE para desarrollo de software.	
UNIDAD IV				
4	Diseñar un traductor dirigido por sintaxis, mediante investigación documental de técnicas de traducción, para implementar el procesamiento de transformación de textos de un lenguaje a otro lenguaje, con creatividad.	Se diseñará un traductor que reciba un lenguaje de entrada y lo traduzca a otro lenguaje. Se revisa que el programa de cómputo traduzca correctamente el lenguaje fuente al lenguaje destino.	Equipo de cómputo con conexión a internet, procesador de textos y un IDE para desarrollo de software.	4 horas
UNIDAD V				
5	Realizar un reporte técnico escrito, mediante el desarrollo de un planteamiento del problema, objetivos generales y particulares, metodología para describir el desarrollo del proyecto, pruebas, resultados y conclusiones, con creatividad y responsabilidad.	Se elabora un reporte técnico que contenga los resultados obtenidos en la realización del proyecto. Se realiza una presentación con una duración de 30 minutos que describa los resultados del proyecto. Se revisa el reporte del proyecto y se evalúa la presentación, para realizar las retroalimentaciones finales.	Equipo de cómputo con conexión a internet y procesador de textos.	4 horas

VII. MÉTODO DE TRABAJO

Encuadre: El primer día de clase el docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-alumno.

Estrategia de enseñanza (docente)

- Exposición conceptual, con lenguaje claro y breve sustentando la información con las fuentes de consulta.
- Explicación de cada una de las fases mediante un caso de estudio práctico donde se aplique un sistema de traducción.
- Coordinar grupos de discusión sobre el caso práctico.
- Asesoría constante durante la elaboración del diseño del traductor.

Estrategia de aprendizaje (alumno)

- Constante investigación documental.
- Participación en grupos de discusión y estudio de caso propuesto por el docente.
- Al finalizar la primera unidad, presentará la propuesta del proyecto en que se especifique la aplicación de un sistema de traducción.
- Durante el semestre tanto en las clases como en la elaboración de las prácticas se deberá ir desarrollando e implementando la propuesta del proyecto.
- Al finalizar el curso se deberá presentar el prototipo con la funcionalidad del traductor y su reporte técnico.
- Resolverá al menos dos exámenes durante el semestre.

VIII. CRITERIOS DE EVALUACIÓN

La evaluación será llevada a cabo de forma permanente durante el desarrollo de la unidad de aprendizaje de la siguiente manera:

Criterios de acreditación

- Para tener derecho a examen ordinario y extraordinario, el estudiante debe cumplir los porcentajes de asistencia que establece el Estatuto Escolar vigente.
- Calificación en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Exámenes escritos	20%
- Prácticas de laboratorio	30%
- Proyecto.....	45%
- Presentación del proyecto	5%
Total	100%

Nota: Es requisito para aprobar la asignatura cumplir en tiempo y forma con lo siguiente: entrega y presentación del proyecto y las prácticas de laboratorio

IX. REFERENCIAS

Básicas

Aho, A., Lam, M., Sethi, R., Ullman, J. (2013). *Compilers: Principles, Techniques, and Tools* (2ª ed.). Estados Unidos de América: Pearson [Clásica].

Dos Reis, A. J. (2018). *Writing Interpreters and Compilers for the Raspberry Pi Using Python*. CreateSpace Independent Publishing Platform.

Flores, J. y Samano, E. (2016). *Construcción de compiladores básicos*. Flex, Bison & MinGw. España: Editorial Académica Española.

Terence, P. (2010). *Language implementations patterns*. Estados Unidos: The pragmatic bookshelf. [Clásica]

Complementarias

Campbell, B., Iyer, S., y Akbal-Delibas, B. (2012). *Introduction to compiler construction in a Java world*. Estados Unidos: CRC Press. [Clásica]

Bal H., Jacobs C., y Langendoen, K. (2012). *Modern Compiler Design*. (2a ed.) Estados Unidos: Springer. [Clásica]

Mogensen, T. (2007). *Basics of compiler design*. Dinamarca: DIKU University of Copenhagen. [Clásica]

X. PERFIL DEL DOCENTE

El docente que imparta esta asignatura deberá poseer título de licenciatura en Ingeniero en Computación o afín, preferentemente con el grado de doctor en Computación o afín. Conocimiento en el desarrollo de compiladores y traductores, prioritariamente con dos años en docencia. Que desempeñe su labor con profesionalismo y tolerancia.