

 Institución Universitaria	MICRODISEÑO CURRICULAR	Código	FDE 058
		Versión	03
		Fecha	2011-07-25

SISTEMAS DE INFORMACIÓN

1. IDENTIFICACIÓN

Asignatura				Compiladores					
Área				Ciencias Básicas de Ingeniería					
Código		CPI82		Pensum		3			
Correquisitos				Prerrequisitos		MII-72			
Créditos	2	TPS	2	TIS	4	TPT	32	TIT	64

2. JUSTIFICACIÓN

El uso de los lenguajes de programación es una tarea cotidiana de un Ingeniero de Sistemas, para tener los elementos de juicio en la elección de un lenguaje programación y de las herramientas que permiten su implementación cuando se desea realizar una aplicación específico, es necesario que él posea el conocimientos básicos del funcionamiento de un compilador, las técnicas que se emplean en la construcción, las ventajas y desventajas del mismo, con la finalidad de poseer el criterio de realizar la mejor elección.

La asignatura de Compiladores da continuidad al área de desarrollo de software, que aborda los conocimientos y habilidades propios de la Ingeniería de Sistemas para el diseño, construcción, e implementación de un desarrollo que satisfaga las necesidades del cliente. Busca formar en los conceptos fundamentales de un compilador, que el estudiante apropie los principios básicos sobre los cuales se fundamenta cada una de las fases en los procesos de compilación de un programa, aplique las técnicas empleadas para que cada una de estas fases contribuya a la transformación de un aplicativo de lenguaje de usuarios a lenguajes de máquina, para poder ser procesado por un equipo de cómputo, y conozca las herramientas requeridas en la construcción de compiladores. Estos principios, técnicas y herramienta son de gran utilidad en el diseño y construcción de un compilador, pero también puede ser empleados en el diseño y desarrollo de otro tipo de software, además dotan al alumno del conocimiento necesario para realizar el proceso de compilación de sus aplicativos en forma consciente y da un visión crítica ante la gran cantidad de compiladores existentes en el medio. Es un curso indispensable en la formación integral de todo Ingeniero de Sistemas.

3. COMPETENCIA

Resolver problemas relacionados con las fases de análisis de un compilador: léxicos, sintácticos y semánticos, a partir de máquinas de estado finito, expresiones regulares y reglas de una gramática independiente del contexto semánticas de un lenguaje sencillo, además de uso de herramientas CASE; conocida como metacompiladores.

 Institución Universitaria	MICRODISEÑO CURRICULAR	Código	FDE 058
		Versión	03
		Fecha	2011-07-25

TABLA DE SABERES:

Saber (contenido declarativo)	Saber complementario (contenido declarativo)	Saber hacer (contenido procedimental)	Ser –Ser con Otros (Contenido actitudinal)
<p>Conceptos de Ambientes de compilación (Análisis y síntesis) y fases del proceso de compilación.</p> <p>Análisis de léxico (conceptos básico, Expresiones regulares, Autómatas finitos: determinísticos y no determinísticos, Tabla de símbolos, Optimización, Generación)</p> <p>Máquina de estados finitos.</p> <p>Análisis Gramatical (Gramática independiente del contexto, tipos de analizadores sintácticos y Generador de analizadores)</p> <p>Análisis semántico (Patrones de Diseño, Tabla de Símbolos, Comprobación de tipos, Generación de código, Código intermedio, Traducción dirigida por la sintaxis)</p> <p>Manejo de errores (Clasificación de los errores, Manejo de los errores léxicos, sintácticos,</p>	<p>Manejo de la lógica de programación y la aplicación en un lenguaje.</p> <p>Trabajo independiente de ejercicios de creación de fases de compilación.</p> <p>Diseño e implementación de autómatas finitos y su aplicabilidad a soluciones diferentes a los compiladores.</p>	<p>Reconocer la importancia de los conceptos básicos de la compilación al construir una aplicación.</p> <p>Diferenciar los procesos de compilación modernos de los tradicionales.</p> <p>Apropiar los conceptos de máquinas de estado finito y los en la solución de problemas diversos.</p> <p>Implementar un analizador léxico a partir de expresiones regulares de un lenguaje.</p> <p>Diseñar y construir un analizador sintáctico empleado gramática independiente del contexto.</p> <p>Implementar las reglas semánticas de un lenguaje sencillo</p>	<p>Responsabilidad frente a la lectura de material y el trabajo independiente.</p> <p>Actitud respetuosa frente al conocimiento del docente y de sus compañeros.</p> <p>Dispuesto a trabajar en equipo valorando las potencialidades de sus compañeros.</p> <p>Demuestra interés en adquirirá nuevos conocimientos y técnicas de aprendizajes en un contexto diferente</p> <p>Actitud ética frente a su entorno social.</p>

 Institución Universitaria	MICRODISEÑO CURRICULAR	Código	FDE 058
		Versión	03
		Fecha	2011-07-25

Saber (contenido declarativo)	Saber complementario (contenido declarativo)	Saber hacer (contenido procedimental)	Ser –Ser con Otros (Contenido actitudinal)
<p>semánticos, Recuperación de errores, Generación de código ejecutable)</p> <p>Optimización de código (Fuentes, Depuración código optimizado y generados de código).</p>			

4. TABLA DE RESULTADOS DEL APRENDIZAJE (CRITERIOS PARA LA EVALUACIÓN – INDICADORES DE COMPETENCIA)

De conocimiento (contenidos declarativos)	De desempeño (contenido procedimental y actitudinal)	Producto (evidencias de aprendizaje)
<p>Identifica las etapas correspondientes al proceso de traducción de un programa fuente a ejecutable.</p> <p>Manipula las herramientas para el análisis léxico.</p> <p>Maneja las herramientas para el análisis sintáctico.</p> <p>Representa y manipula expresiones regulares.</p> <p>Identifica errores y es capaz de depurarlos con la ayuda de las herramientas de optimización.</p>	<p>Implementa, modela, y programa una gramática sencilla, y aplica todas las fases de un compilador en su proyecto.</p> <p>Aplica los metacompiladores para obtener las fases léxica, sintáctica y semántica de un compilador.</p>	<p>Diseño de un proyecto compilador con todas sus fases de análisis, generación de código y optimización.</p> <p>Usa las máquinas de estado finito para dar soluciones a problemáticas diversas en las que se pueden aplicar.</p>

 Institución Universitaria	MICRODISEÑO CURRICULAR	Código	FDE 058
		Versión	03
		Fecha	2011-07-25

5. TABLA DE ESTRATEGIAS METODOLÓGICAS

Actividades de enseñanza-aprendizaje	Actividades de trabajo independiente	Actividades de evaluación		
		Actividad	%	Fecha
Clases magistrales con el apoyo de recursos tecnológicos Talleres con acompañamiento del docente Prácticas en el laboratorio con asesorías del docente Estudio de casos: proyecto de aula. Solución de ejercicios con la guía del docente.	Lectura de temas relacionados con fases de un compilador. Guía de trabajo independiente sobre un analizador léxico. Guía de trabajo independiente sobre un analizador sintáctico. Guía de trabajo independiente sobre un analizador semántico.	Pruebas escritas unidad 1.	15	Transcurso de las clases
		Formulación del proyecto de aula.	15	
		Practica sobre parser de un lenguaje sencillo.	15	
		Diseño y construcción de un analizador léxico usando metacompiladores.	15	
		.		
		Parcial escrito: Unidades 1, 2 y 3	20	
Practica final: Diseño y construcción de un analizador léxico, sintáctico y semántico usando metacompiladores	20			

6. BIBLIOGRAFÍA

- Aho, A.V., Sethi, R., Ullman, J.D. (1990) Compiladores: principios, técnicas y herramientas. Addison-Wesley Iberoamericana, Madrid, ISBN: 0-201-62903-8. Copias: 3 en castellano, 2 en inglés. Lugar: Biblioteca Campus Signatura: 681.3.06 AHO
- Gálvez, S. y Mora, Á. (2005) Java a tope: Traductores y Compiladores. Universidad de Málaga. España.

	MICRODISEÑO CURRICULAR	Código	FDE 058
		Versión	03
		Fecha	2011-07-25

- Moreno, M. y otros. (2006) *Compiladores e Intérpretes: teoría y práctica*. Pearson. España. 2006.
- Louden, K.C. (1997) *Compiler Construction: Principles and Practice*. PWS Publishing Company, Boston, Massachusetts, ISBN: 0-534-93972-4. Copias: 2 en inglés. Lugar: Biblioteca Campus. Signatura: 681.3.06 LOU
- Vivancos, E., Moreno, L., Gisbert, V y Benedí, J.M. (2000) *Compiladores I: una introducción a la fase de análisis*. Servicio de Publicaciones de la Universidad Politécnica de Valencia, ISBN: 84-7721-915-X. Copias: 3, Lugar: Biblioteca Campus. Signatura: CI 681.3 COM
- Iñesta, J.M., García, P. y Gracia, I. (1998) *Técnicas básicas para el diseño de compiladores*. Servicio de Publicaciones de la Universitat Jaume I, ISBN: 84-8021-259-4, 1998). Copias: 1, Lugar: Biblioteca Campus. Signatura: 681.3.06 IÑE
- Bennet, J.B.(1996) *Introduction to Compiling Techniques: a first course using ansi C, Lex and Yacc*. Mcgraw-Hill Publishing Company, England, ISBN: 007709221X. Copias: 2 en inglés. Lugar: Biblioteca Campus. Signatura: 681.3.06 BEN
- Teufel, B., Schmidt, S., Teufel, T. (1995) *Compiladores: conceptos fundamentales*. Addison-Wesley Iberoamericana, Wilmington, Delaware, ISBN: 0-201-65365-6. Copias: 1 castellano. Lugar: Biblioteca Departamento

Elaborado por:	<i>Gabriel E. Taborda</i>
Versión:	<i>3</i>
Fecha:	
Aprobado por:	