

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
COORDINACIÓN DE FORMACIÓN BÁSICA
COORDINACIÓN DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA
DEPARTAMENTO DE ACTUALIZACIÓN CURRICULAR Y FORMACIÓN DOCENTE

DESCRIPCIÓN GENÉRICA DE UNIDADES DE APRENDIZAJE

Descripción Genérica

Nombre: **Compiladores**

Etapa: **Terminal**

Área de conocimiento: **Software de base**

Plan 2008-1

Competencia:

Analizar las diferentes fases que constituyen el proceso de traducción de un lenguaje de programación, con el fin de aplicarlas en el desarrollo de sistemas aplicación real

Evidencia de desempeño:

Desarrollo de una aplicación en la que se utilicen las técnicas vistas en clase

Distribución	HC	HL	HT	HPC	HCL	HE	CR	Requisito
	2	2	2			2	8	

Contenidos Temáticos

1. Visión General de Compiladores
 - 1.1 Introducción
 - 1.2 Fases de un compilador
 - 1.3 Análisis léxico
 - 1.4 Análisis sintáctico o gramatical
 - 1.5 Análisis semántico
 - 1.6 Generador de código intermedio
 - 1.7 Optimización
 - 1.8 Generación de código

2. Análisis Léxico
 - 2.1 Función del analizador léxico
 - 2.2 Componentes léxicos, patrones y lexemas
 - 2.3 Atributos de los componentes léxicos
 - 2.4 Manejo de “buffers” de entrada
 - 2.5 Especificación de componentes léxicos
 - 2.6 Expresiones regulares y autómatas finitos
 - 2.7 Reconocimientos de componentes léxicos
 - 2.8 Manejo de errores léxicos

3. Tabla de símbolos
 - 3.1 Propósito de la tabla de símbolos
 - 3.2 Atributos y estructura de datos para una tabla de símbolos
 - 3.3 Operaciones en la tabla de símbolos

3.4 Eficacia de las tablas de símbolos

4. Análisis de sintaxis

- 4.1 Propósito del analizador sintáctico
- 4.2 Lenguajes y gramáticas
- 4.3 Gramáticas independientes al contexto
- 4.4 Análisis sintáctico descendente
- 4.5 Análisis sintáctico ascendente
- 4.6 Análisis sintáctico predictivo
- 4.7 Análisis sintáctico de precedencia
- 4.8 Problemas del análisis sintáctico
- 4.9 Manejo de errores

5. Traducción dirigida por sintaxis

6. Análisis semántico

- 6.1 Introducción
- 6.2 Verificación estática
- 6.3 Especificación de un comprobador tipos simple

7. Generación de código

- 7.1 Introducción
- 7.2 Preparación para la generación de código
- 7.3 Generador de código simple
- 7.4 Estrategias para la generación de código

Referencias bibliográficas actualizadas

Modern Compiler Implementation in Java

Andrew W. Appel y Jens Palsberg

Cambridge University Press

Compilers: Principles, Techniques, and Tools (2nd Edition)

Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman

Addison Wesley

Construcción de Compiladores: Principios y Prácticas

Kenneth C. Loudon

Thomson International