

C 2000

10

UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE CIENCIAS EXACTAS Y NATURALES

1. DEPARTAMENTO: Computación
2. CUATRIMESTRE: Segundo de 2000
3. ASIGNATURA: Laboratorio VII (Diseño de compiladores)
4. CARRERA: Licenciatura en Ciencias de la Computación
5. CARACTER DE LA MATERIA: obligatoria
6. NUMERO DE CODIGO DE CARRERA: 18
7. NUMERO DE CODIGO DE MATERIA: C
8. PUNTAJE: ----
9. PLAN DE ESTUDIOS AÑO: (1993)
10. DURACION DE LA MATERIA: cuatrimestral
11. HORAS DE CLASE SEMANAL:
 - a) TEORICAS: 3hs
 - b) LABORATORIO: ---
 - c) PRACTICAS: 3hs
 - d) SEMINARIOS: ---
12. CARGA HORARIA TOTAL: 6hs semanales. 90 hs cuatrimestrales.
13. ASIGNATURAS CORRELATIVAS: Teoría de lenguajes y autómatas – Arquitectura y sistemas operativos
14. FORMA DE EVALUACION: prácticos y final
15. PROGRAMA Y BIBLIOGRAFIA: adjuntos a esta hoja

Fecha: 29 de junio 2000

Prof. Responsable
Dr. JUAN MIGUEL SANTOS

Directora Dra. Patricia Borensztejn

Diseño de Compiladores

Contenido

1. Introducción

Arquitectura tradicional de un compilador

Etapas de análisis. Un compilador - vista rápida

Diseño de un lenguaje. Pros y contras

2. Análisis Léxico.Sintáctico

Definición y traducción dirigida por sintaxis. Atributos heredados y sintetizados. Definiciones S-atributos y L-atributos.Traduccion top-down y bottom-up

Distintas herramientas standard para la automatización del proceso.

(Lex/Yacc)

3. Tablas de Símbolos

Definición. Nombres y bindings. Scopes Construcción de la TS durante el parsing

4. Análisis semántico

Sistemas de tipos. Equivalencia de tipos. Conversiones. Tipos hardware/básicos; tipos construidos por el programador. Semántica de los tipos de los lenguajes standard. Soporte de runtime. Sobrecarga de funciones y operadores. Funciones polimórficas

5. Código Intermedio

Lenguajes intermedios. Maquina de pila abstracta.Generación de una representación intermedia. Evaluación de expresiones. Estructuras de control. Análisis de las distintas construcciones utilizadas en los lenguajes standard.Backpatching.Llamados a procedimientos

6. Código Objeto

Generación de código objeto. Asignación de registros. Generacion de codigos a partir de dags. Generadores de codigos.

7. Optimización

Técnicas de optimización

Bloques basicos y graficos de flujo. Informacion de proximo uso.

Representacion de bloques basicos mediante dags. Optimizacion Peephole.

Fuentes de optimizacion. Loops en grafos de flujo. Analisis global de flujo de datos. Transformaciones que mejoran el codigo. Aliases.

8. Runtime

Disposición en memoria. Acceso a nombres no locales. Manejo del almacenamiento en tiempo de ejecucion Administración de memoria. Garbage collection Pasaje de parámetros

9. Debugging simbólico

10. Entornos integrados de desarrollo

Editores guiados por sintaxis. Compilacion incremental Parsing incremental descendente. Parsing incremental ascendente. Representacion de Ghezzi Mandriolli.

Proyecto

El proyecto realizado durante el transcurso del cuatrimestre consiste en la evaluación y diseño de un nuevo lenguaje, y el diseño e implementación de un compilador para el mismo. El lenguaje, llamado MQR es un subconjunto acotado del lenguaje Java, compilándose a código assembler para una arquitectura 80x86. El compilador, se desarrolla utilizando el lenguaje Java, junto con algunas herramientas y librerías diseñadas específicamente para el proyecto.

Dr. RAJICIA BORENSZTEJN
DIRECTORA
DEPTO. DE COMPUTACION
F. C. E. y N. UBA

Bibliografía

 Aho, Setti, Ullman: Compiladores, principios, técnicas y herramientas Adison Wesley

 Fischer, Grodzinsky: The Anatomy of programming Languajes, Prentice Hall International

 Stroustrup The C++ ProgrammingLenguaje Addison Wesley

 <http://java.sun.com/docs/index.html>

Requisitos

 Teoría de Lenguaje y Automatas

 Arquitectura y Sistemas Operativos



Dra. PATRICIA BORENSZTEIN
DIRECTORA
DEPTO. DE COMPUTACION
F. C. E. y N. UBA



Dra. PATRICIA BORENSZTEIN
DIRECTORA
DEPTO. DE COMPUTACION
F. C. E. y N. UBA