

Proyecto de Compiladores

Tercera entrega, Análisis semántico

15 de abril de 2008

1. Indiciaciones Generales

En la tercera entrega se deben adicionar atributos a los símbolos no terminales de la gramática y acciones semánticas a las producciones que permitan realizar las siguientes verificaciones semánticas y generar los correspondientes mensajes de error:

- Detectar inconsistencias entre el tipo de un operador y el de sus operandos
- Detectar variables y funciones redefinidas
- Ausencia o duplicidad del main, o que tenga parámetros

La idea es que durante el análisis sintáctico se vaya asignando valores a los atributos y luego, haciendo un recorrido sobre el árbol de análisis sintáctico, que puede ver esa información almacenada, se realice la verificación de tipos.

Adicionalmente, se deben detectar los siguientes errores:

- El llamado al método no coincide con la definición, bien sea en el tipo o en el número de parámetros o en el tipo que retorna el método
- Variables y funciones indefinidas
- Intento de subindicar variable que no es arreglo o de usar como subíndice algo que no es un entero
- División por cero

Para esta entrega será indispensable que el proyecto:

- Reciba el nombre del archivo a compilar como parámetro de entrada
- El análisis léxico debe contemplar todas las posibilidades planteadas en la descripción del lenguaje decaf
- El árbol de análisis sintáctico debe estarse construyendo correctamente y debe haber una forma de escribirlo en forma legible en pantalla
- El árbol de análisis sintáctico debe corresponder exactamente a la gramática de decaf
- Se debe estar almacenando en la tabla de símbolos lexemas, tipo y tamaño por cada identificador encontrado.
- El compilador debe hacer recuperación de errores y debe reportar correctamente la línea donde se encuentra el error.

2. Entrega y sustentación

Se deben entregar los fuentes del programa escrito en C ó en Python.

Las salidas del proyecto serán: el árbol de análisis sintáctico decorado, la tabla de símbolos llena y un archivo texto donde se reporten todas las revisiones semánticas que se realizaron. Al final de la ejecución, en pantalla se verán los errores generados: léxicos, sintácticos y semánticos.

En la evaluación del código se dará mucha importancia a que se hayan realizado todas las verificaciones solicitadas, a que las inconsistencias se detecten oportuna y claramente y que haya una adecuada recuperación de errores.

En la sustentación evaluará el dominio del estudiante sobre su trabajo y sobre las herramientas flex y yacc.

3. Fecha de entrega

Este proyecto se propone el martes 15 de abril y deberá entregarse el miércoles 7 de mayo en sustentación. Las sustentaciones serán individuales y se realizarán el miércoles 2 de abril. En la clase podrán apuntarse para elegir la hora de sustentación.