República Bolivariana de Venezuela

Ministerio del Poder Popular para la Educación Superior

Universidad Nacional Experimental Rómulo Gallegos

Área de Ingeniería en Sistemas

San Juan de los Morros- Guárico

Análisis Grupal

Profesor: Bachiller: Ramón Ramírez Jonathan Hernández

C.I: 23.786.812

Libni Cuiche

C.I: 21.098.542

Juan de los Morros, de junio del 2016

Análisis Léxico

Es la primera fase de un compilador consiste en un programa que requiere como entrada el código fuente de otro programa (secuencia de caracteres) o símbolo. Esto tokens sirven para una posterior etapa del proceso de traducción, siendo la entrada para el analizador sintáctico (en inglés parser).

Etapas o funciones del Análisis Léxico

1. Eliminación de espacio en blanco y comentarios: Si, el análisis léxico elimina el espacio en blanco, el analizador sintáctico tendrá que considerarlo.
2. Reconocimiento de Identificadores y Palabras Claves: Es el encargado de construir los lexemas que constituyen los identificadores de los lenguajes de programación.
3. Reconocimiento de constantes: Su tarea es equipar dígitos para formar enteros.

Modo de Empleo del Análisis Léxico

Se emplea o usa en máquinas de estado finito, que contiene información de las posibles secuencias de caracteres que puede conformar cualquier token o símbolo que sea parte del lenguaje (las instancias individuales de estas secuencias de caracteres son denominados lexemas). También se encarga de verificar que el texto este escrito en un formato aceptado para todo el programa que está escrito en un lenguaje de programación, al igual que se encarga de verificar que tenga coherencia, los analizadores léxicos sirven en gran parte para resolver problemas que pueden surgir a causa de que el programa no tenga coherencia o no esté bien estructurado.

Análisis Sintáctico

Es una de las partes de un compilador que transforma su entrada en un árbol de derivación, es decir ; convierte el texto de entrada en otras estructuras (comúnmente arboles), que son más útiles para el posterior análisis y capturan la jerarquía implícita de la entrada. El análisis sintáctico también es un estado inicial del análisis de frases de lenguaje natural.

Función del Análisis Sintáctico

Es procesar los tokens o símbolos provenientes de un analizador léxico para construir la estructura de datos, como son los arboles de sintaxis abstractas.

El análisis sintáctico es usado para generar diagramas de lenguajes que usan flexión gramatical, como los idiomas romanes o el latín. Los lenguajes habitualmente reconocidos por los analizadores sintácticos son los lenguajes libres de contexto (reconocibles por autómatas de pilas).

Modo de Empleo del Análisis Sintáctico

Se emplea de dos formas:

1. Análisis sintáctico ascendente (Zop – Dowen - Parsen): es empleado con el símbolo inicial e intentan transformar la entrada, que conlleva intuitivamente a dividir la entrada de manera progresiva en parte cada vez más pequeña. Ejemplo: java CC.
2. Análisis sintáctico ascendente (Botton – Up –Parsen): se emplea de forma que se empieza con la entrada e intentan llegar hasta el símbolo inicial, intuitivamente el analizador intenta encontrar los símbolos más pequeños y progresivamente construir la jerarquía de símbolos hasta el inicial. Ejemplo: Yacc.

Análisis Semántico

Se compone de un conjunto de rutinas independientes, llamados por los analizadores morfológico (léxico) y sintáctico. Este tipo de análisis utiliza como entrada el árbol sintáctico detectado por el análisis sintáctico para comprobar restricciones de tipo y otras limitaciones semánticas y preparar la generación de código.

Etapas del Análisis Semántico

\*Dinámico: Posee los aspectos que solo pueden ser conocidos en tiempos de ejecución.

\*Estático: Posee aspectos que solo pueden ser controlados en el tiempo de compilación.

\*Verificación de tipos: Aquí solo el compilador verifica si cada operador tiene operandos permitidos por la especificación del lenguaje fuente.

Modo de Empleo y Aplicación del Análisis Semántico

Se utiliza como entrada el árbol sintáctico detectado por el análisis sintáctico para comprobar las restricciones de tipo y las otras limitaciones semánticas y proponer la generación del código.

En compiladores de un solo paso, las llamadas a las rutinas semánticas se realizan directamente desde el analizador sintáctico y son dichas rutinas las que llaman al generador de código. El instrumento más utilizado para conseguir es la gramática de atributos.

En cualquier caso, las rutinas semánticas suelen hacer uso de una pila (la pila semántica) que contiene la información semántica asociada a las operando (y a veces a los operadores) en forma de registros semánticos.