

ANALIZADOR LEXICO

OMAR ALEJANDRO QUIROZ TRUJILLO

- **DESARROLLO**

Su principal función consiste en leer los caracteres de entrada y elaborar como salida una secuencia de componentes léxicos que utiliza el analizador sintáctico para hacer el análisis. Esta interacción, suele aplicarse convirtiendo al analizador léxico en una subrutina o corrutina del analizador sintáctico.

FUNCIÓN SECUNDARIA

Como el analizador léxico es la parte del compilador que lee el texto fuente. También puede realizar ciertas funciones secundarias en la interfaz del usuario, como eliminar del programa fuente comentarios y espacios en blanco en forma de caracteres de espacio en blanco, caracteres TAB y de línea nueva. Otra función es relacionar los mensajes de error del compilador con el programa fuente.

En algunos compiladores, el analizador léxico se encarga de hacer una copia del programa fuente en el que están marcados los mensajes de error. Si el lenguaje fuente es la base de algunas funciones de pre procesamiento de macros, entonces esas funciones del preprocesador también se pueden aplicar al hacer el análisis léxico.

FUENTE:

http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro32/21_funcin_del_analizador_lexico.html

El analizador léxico. Se encarga de buscar los componentes léxicos o palabras que componen el programa fuente, según unas reglas o patrones. La entrada del analizador léxico podemos definirla como una secuencia de caracteres.

CARACTERISTICAS

El analizador léxico tiene que dividir la secuencia de caracteres en palabras con significado propio y después convertirlo a una secuencia de terminales desde el punto de vista del analizador sintáctico, ya que es su entrada. El analizador léxico reconoce las palabras en función de una gramática regular de manera que sus no terminales se convierten en los elementos de entrada de fases posteriores.

FUNCIÓN

Su principal función consiste en leer los caracteres de entrada y elaborar como salida una secuencia de componentes léxicos que utiliza el analizador sintáctico para hacer el análisis. Esta interacción, suele aplicarse convirtiendo al analizador léxico en una subrutina o corrutina del analizador sintáctico

- Reconocer los identificadores de usuario, números, palabras reservadas del lenguaje y tratarlos correctamente con respecto a la tabla de símbolos (solo en los casos que debe de tratar con la tabla de símbolos).
- Llevar la cuenta del número de línea por la que va leyendo, por si se produce algún error, dar información sobre donde se ha producido.
- Puede hacer funciones de preprocesador.

FUENTE:

<http://materiacompiladores3110.blogspot.com/2017/02/analizador-lexico.html/>

- **PRUEBAS DE QUE EL PROGRAMA FUNCIONA**

```

PROYECTO.py X  archivo
PROYECTO.py > ...
1 # Importa el módulo de expresiones regulares
2 import re
3
4 # función para realizar el análisis léxico de una cadena de entrada
5 def lexico(input_string):
6     # Inicializa listas y diccionario para almacenar tokens y la tabla de símbolos
7     tokens = []
8     symbol_table = {}
9
10    # Define patrones de expresiones regulares para diferentes tipos de tokens
11    patterns = [
12        (r'([a-zA-Z][a-zA-Z0-9_]*)', 'ID'), # Identificador
13        (r'(\d+)', 'ENTERO'), # Entero
14        (r'(\d+\.\d+)', 'REAL'), # Real
15        (r'([+|-|*|/])', 'OP_ADICION'), # Operador de adición
16        (r'([*|/|/])', 'OP_MULTIPlicACION'), # Operador de multiplicación
17        (r'=', 'OP_ASIGNACION'), # Operador de asignación
18        (r'(<|>|<=|>=|!=)', 'OP_RELACIONAL'), # Operador relacional
19        (r'&&', 'OP_AND'), # operador And
20        (r'\|\|', 'OP_OR'), # operador Or
21        (r'!', 'OP_NOT'), # Operador not
22        (r'(', 'PARENTESIS_ABIERTO'), # Paréntesis abierto
23        (r')', 'PARENTESIS_CERRADO'), # Paréntesis cerrado
24        (r'//', 'OP_DIVISION') # Operador división
25    ]
26
27    # Tabla de Símbolos:
28    hola
29    ID
30    Ingrese el nombre del archivo de salida para la tabla de símbolos: archivo
31    PS C:\Users\Omar\OneDrive\Documentos\Escritorio\Septimo Semestre\SEM TRADUCTORES II\PROYECTO & C:\Python\python.exe "C:\Users\Omar\OneDrive\Documentos\Escritorio\Septimo Semestre\SEM TRADUCTORES II\PROYECTO\PROYECTO.py"
32    Ingrese la cadena a analizar: +
33
34    Tokens: [('OP_ADICION', '+')]
35
36    Tabla de Símbolos:
37    + OP_ADICION
38    Ingrese el nombre del archivo de salida para la tabla de símbolos: archivo
39    PS C:\Users\Omar\OneDrive\Documentos\Escritorio\Septimo Semestre\SEM TRADUCTORES II\PROYECTO >

```

PROYECTO.py archivo

archivo

1	Símbolo	Tipo
2	-----	
3	+	OP_ADICION
4		

• CONCLUSIÓN

Los analizadores léxicos son fundamentales en el desarrollo de compiladores y lenguajes de programación, ya que proporcionan la base para el análisis posterior del código fuente. Su eficiencia, capacidad de manejo de errores y su papel en la generación de tokens son aspectos críticos para el rendimiento general del sistema.

Al crear este programa se me hizo algo complicado porque estoy comenzando aprender mejor el lenguaje de Python y me gustaría darle una mejor vista a este tipo de archivo, donde tenga una interfaz bonita para que el analizador se vea estético

y demasiado entendible. Poco a poco comencare a darle unas mejores para la hora de presentar el proyecto quede de la manera como estoy planeando.

