

**ANALIZADOR LÉXICO,
PRIMERA ENTREGA COMPILADORES**

Realizado por:

Luis Miguel Marulanda Torres

Angel David Santa Giraldo

Presentado a:

Fernando Galindres

Compiladores

Ingeniería de sistemas y computación

2019-1

UTP

Descripción del analizador Léxico

El analizador léxico realizado sigue la siguiente estructura. atravesando las siguientes etapas:

Importe de librerías: Se importan las librerías a las que se le encuentra utilidad en el desarrollo del analizador léxico, tales como: re (para expresiones regulares), codecs (para la gestión del fichero en el cual se pondrán las pruebas). Adicionalmente, se descargará desde ply la librería con el fin de que funcione el lexer.

Definición de Tokens: Tras leer la documentación proporcionada, se llegó a la conclusión de que los tokens necesarios para la entrega son los siguientes:

'ID': Identificador,

'NUMERO': Dato Numérico,

'PLUS': Más (+),

'MINUS': Menos (-),

'POR': Multiplicador (*),

'DIVIDE': División (/),

'MODULO': Porcentaje (%),

'NEGLOGICA': Admiración (i),

'CORCHDER': Corchete Derecho (]),

'CORCHIZQ': Corchete Izquierdo ([),

'PARENTDER': Paréntesis Derecho ()),

'PARENTIZQ': Paréntesis Izquierdo ((),

'YAND': Andersen Doble (&&),

'OOR': Doble Barra (| |),

'ASIGNAR': Asignación (=),

'MAYORQUE': Mayor que (>),

'MENORQUE': Menor que (<),

'ACCESO': Acceder a información de un objeto (.),

'MENORIGUAL': Menor o igual (<=),

'MAYORIGUAL': Mayor o igual (>=),

'COMPIGUAL': Doble igual (==),

'COMPDIFERENTE' Diferencia (i=),

'FINALIZADOR': Punto y coma (;),

'LLAVEDER': Llave derecha (}),

'CADENA': String,

'BIN': Número Binario,

'COMILLASIMPLE': Comilla Simple ('),

'COMA': (,),

'LLAVEIZQ': Llave izquierda ({),

'COMMENT': Comentario (//text, /*text*/),

'NUMEROC': Número en notación científica 3e27, -3E28, 3.66E-9,

'COMILLADOBLE': Comillas dobles (").

PALABRAS RESERVADAS: En cuanto a las palabras reservadas, es de aclarar que se definieron en un diccionario, de tal manera que por ejemplo la palabra reservada else, implícitamente fuera un ELSE en cuanto a su estructura como Token. Las palabras reservadas con su respectivo significado implícito son las siguientes:

'class': 'CLASS', 'extends': 'EXTENDS', 'void': 'VOID', 'int': 'INT', 'boolean': 'BOOLEAN', 'string': 'STRING', 'return': 'RETURN', 'if': 'IF', 'else': 'ELSE', 'while': 'WHILE', 'break': 'BREAK', 'continue': 'CONTINUE', 'this': 'THIS', 'new': 'NEW', 'length': 'LENGTH', 'true': 'TRUE', 'false': 'FALSE', 'null': 'NULL'

Y tienen el formato: (Palabra reservada): Significado.

Finalmente, en esta etapa para lograr tener las palabras reservadas implícitamente como token, se concatenaron los valores implícitos de las palabras reservadas a los tokens de la siguiente manera:

else: ELSE, para el analizador léxico else sería una palabra reservada y ELSE un token más.

Implementación con variables: En esta etapa se implementaron los tokens mediante la expresión regular, tratando de implementar los reconocidos de manera atómica, como aquellos representados con un símbolo o máximo dos (=, ==, &&,}).

Implementación con funciones: Para aquellos tokens con definiciones más complejas, o con diferentes casos dependiendo su estructura se utilizó una implementación mediante funciones, por ejemplo, para el caso de los identificadores (ID) o comentarios simples y dobles.

Refiriéndose a las funciones requeridas, como por ejemplo reemplazar caracteres como ñ, Ñ o tildados por guion bajo, resultó necesaria la implementación de la función Reemplazar, la cual recibe el elemento a evaluar cuya característica sea que cuente con ñ, Ñ o caracteres tildados.

En cuanto a otros elementos necesarios, fue fundamental la gestión de archivos mediante la librería codecs, la cual interpreta el directorio para lectura y obtiene los elementos que tiene el fichero.

Errores Conocidos

1. No separa correctamente los tokens (Identificador) cuando su estructura es errónea:
Ñempanada: Ñ como error, empanada como ID.
2. No separa correctamente con un espacio los identificadores de números u otros tokens:
Pato23: Pato como ID, 23 como CADENA.
3. No detecta correctamente cuando un comentario o string no es cerrado:
"Cadena : Reconoce " Comilla Doble, Cadena como ID.
/*String : Reconoce / Como Divisor, * como multiplicación y String como ID.