

UNIVERSIDAD DE GUADALAJARA



CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E INGENIERIAS

Seminario de Solución de Problemas de Traductores de Lenguajes II

Reporte de practica

Nombre del alumno:	Casillas Núñez Fernando
Profesor:	López Franco Michel Emanuel
Título de la practica:	"Analizador Sintáctico"
Fecha:	26/05/2020

Introducción

En esta practica se continua en el proceso de la creación de un compilador. Esta vez, expandamos el uso del analizador léxico para complementarlo

con uno sintáctico, que nos permitirá validar si es una cadena valida propia del lenguaje o no.

Un analizador sintáctico es un programa que es capaz de recibir una cadena de símbolos de acuerdo a las reglas de una gramática formal. El analizador convierte el texto recibido en otro tipo de estructuras (comúnmente arboles, como veremos mas adelante con la implementación del árbol sintáctico); crea tokens de una secuencia de caracteres de entrada y son los mismos tokens los que son procesados por el parser para convertirlos en una estructura.

Metodología

Para la implementación del presente analizador sintáctico, se toma en cuenta el método de tablas de análisis LR (del ingles: Left to Right, de izquierda a derecha), la cual consta de dos partes, las funciones de desplazamiento y acción.

Funciona de la siguiente manera: primero se determina el estado usando el tope de una pila y el símbolo actual de nuestra entrada de datos a analizar. Después, se consulta en la tabla de análisis de función acción con los parámetros anteriores y que puede dar como resultado.

La siguiente parte que se muestra, es una parte del código principal que permite el análisis sintáctico:

```
int k = 0;
int accion = 1;
while(accion != 0)
{
    int fila = (int)pila.peek();

    int column = lista.get(k).getTID();
    acción = SyntaxTable[fila][column];

    if(acción == 0)
    {
        if(!esValida)
            OptionPane.showMessageDialog(this, "Cadena invalida para el lenguaje"); break;
    }
}
else if(accion == -1)
{
    esValida = true;
    if(esValida)
```



```

JOptionPane.showMessageDialog(this, "Cadena valida para el lenguaje"); break;
}
else if(accion > 0)
{
    pila.push(columna);
    pila.push(accion);
    k++;
}

else
{
    int x = ruleslist.get((accion-1+2)*(-1)-1).getCantGenerada(); int cantpop =
    x*2;

    for(int i=0; i<cantpop; i++)
        pila.pop();

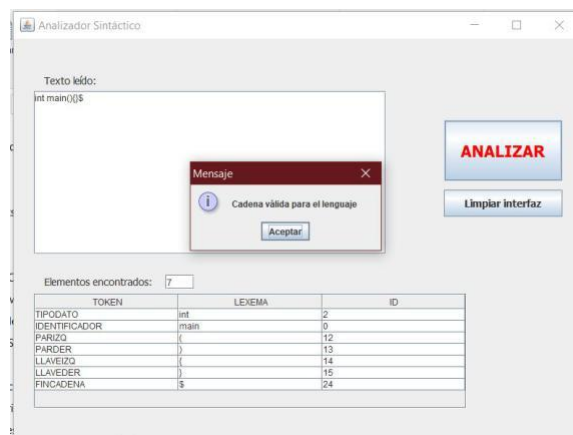
    fila = (int)pila.peek();
    columna = ruleslist.get((accion-1+2)*(-1)-1).getLadolzq();
    pila.push(columna);
    pila.push(SyntaxTable[fila][columna]);
}
}

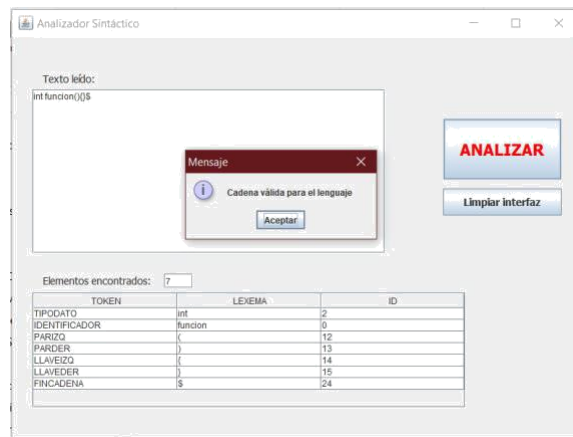
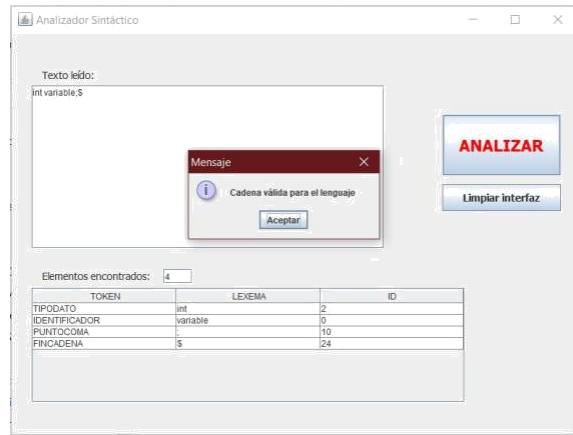
```

Para un mayor entendimiento y profundizaje de los metodos implementados en el programa, sientase bienvenido a revisar los documentos anexos a esta entrega de reporte, donde viene el software completo.

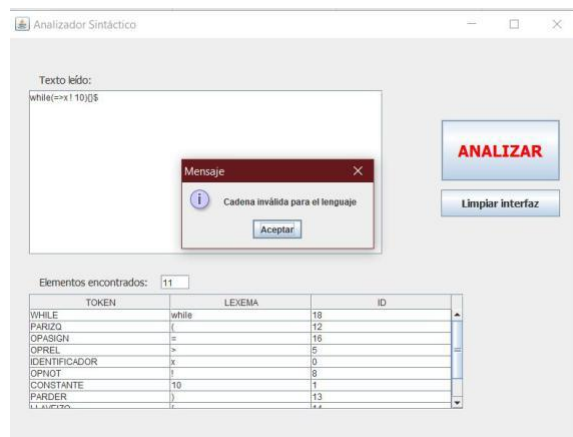
Resultados

Comenzaremos mostrando algunas cadenas que son gramaticalmente validas:





Ahora un ejemplo de una cadena que no es valida:



Referencias

1.- Traductores, Compiladores e Interpretes. TEMA 3: Analisis Sintactico. Mar a del Mar Aguilera Sierra, Sergio Galvez Rojas. Departamento Lenguajes y Ciencias de la Computacion, Universidad de Malaga. www.lcc.uma.es/~galvez/ftp/tci/tictema3.pdf