1 Seminario de Solución de Problemas de Traductores de Lenguaje II	
27/02/2023	Sección: D02
Seminario de Solución de Problemas de Traductores de Lengua	je II
Tarea 5:	
Analizador Sintáctico Con Tabla	
Nombre del maestro:	
López Franco Michel Emanuel	
Nombre del alumno: Ramos Calderón Christian Daniel	
Código del alumno:	
216577014	

2	Seminario de	Solución de Problema	s de Traductores d	le Lenguaje II
---	--------------	----------------------	--------------------	----------------

Índice

Contents

ndice	2
ntroducción	3
Desarrollo	3
Capturas de pantalla	
Conclusión	7
Bibliografía	7

3 | Seminario de Solución de Problemas de Traductores de Lenguaje II

Introducción

Para esta actividad el maestro nos proporciono una tabla para el uso del analizador Sintáctico y de esta manera tomarla mediante un archivo de texto y tenemos también la matriz que de igual manera nos la dio el, mediante lo explicado en clase generar un programa a partir de lo proporcionado.

Desarrollo

Así que tenemos que desarrollar tanto como el analizador léxico como el analizador Sintáctico siendo posible que el programa pueda leer la siguiente gramática que son las reglas proporcionadas por el maestro.

Básicamente para el desarrollo del programa anteriormente utilizamos la clase EP con la cual manejamos los Terminales, No terminales y el Final de la cadena, de esta manera solo cambiamos el cómo están las reglas porque teníamos otras para una gramática en especial ahora con esto podríamos integrar cualquier tipo de gramática de un lenguaje C.

Las reglas proporcionadas por el maestro son las siguientes:

- R1 programa ::= Definiciones
- R2 Definiciones ::= \e
- R3 Definiciones ::= Definicion Definiciones
- R4 Definicion ::= DefVar
- R5 Definicion ::= DefFunc
- R6 DefVar ::= tipo identificador ListaVar ;
- R7 ListaVar ::= \e
- R8 ListaVar ::= , identificador ListaVar
- R9 DefFunc ::= tipo identificador (Parametros) BlogFunc
- R10 Parametros ::= \e
- R11 Parametros ::= tipo identificador ListaParam
- R12 ListaParam ::= \e
- R13 ListaParam ::= , tipo identificador ListaParam
- R14 BlogFunc ::= { DefLocales }
- R15 DefLocales ::= \e
- R16 DefLocales ::= DefLocal DefLocales
- R17 DefLocal ::= DefVar
- R18 DefLocal ::= Sentencia
- R19 Sentencias ::= \e
- R20 Sentencias ::= Sentencia Sentencias
- R21 Sentencia ::= identificador = Expresion ;
- R22 Sentencia ::= if (Expresion) SentenciaBloque Otro
- R23 Sentencia ::= while (Expresion) Bloque
- R24 Sentencia ::= return ValorRegresa ;
- R25 Sentencia ::= LlamadaFunc :
- R26 Otro ::= \e

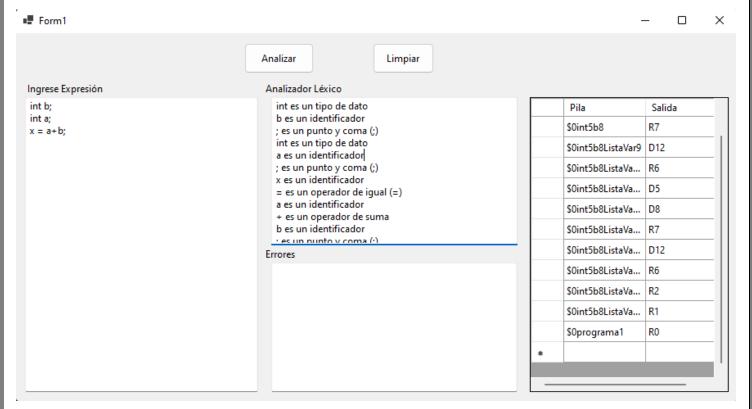
4 | Seminario de Solución de Problemas de Traductores de Lenguaje II

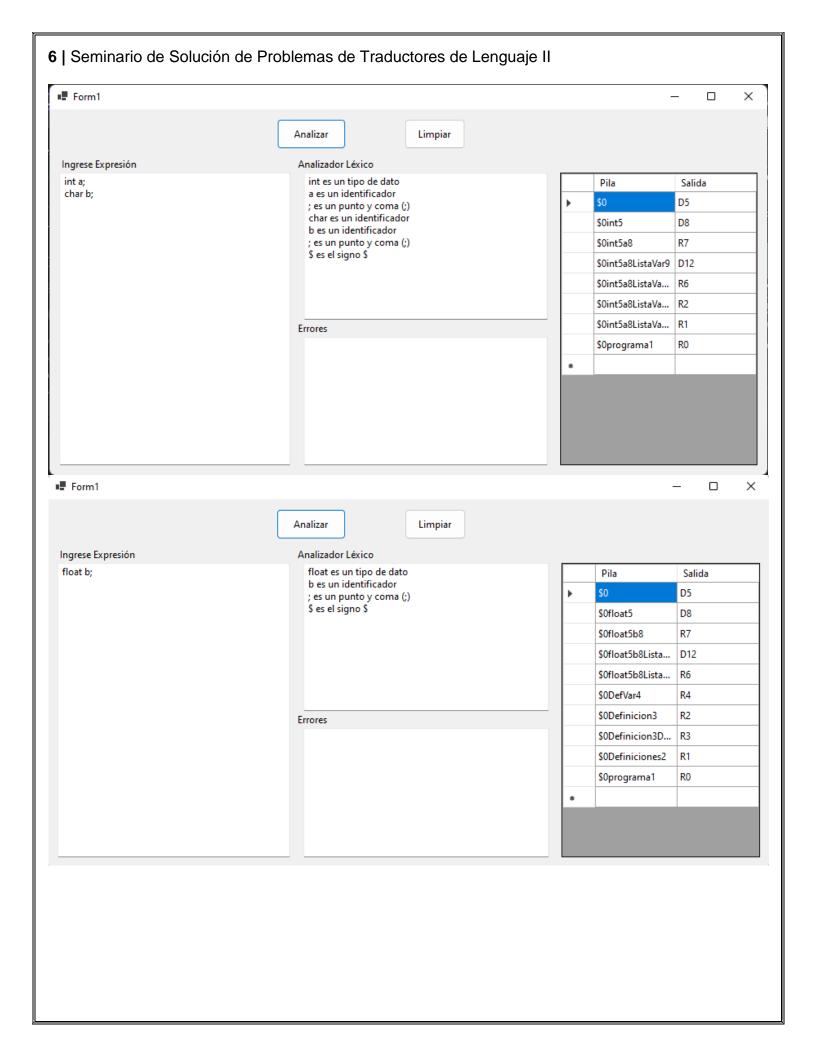
- R27 Otro ::= else SentenciaBloque
- R28 Bloque ::= { Sentencias }
- R29 ValorRegresa ::= \e
- R30 ValorRegresa ::= Expresion
- R31 Argumentos ::= \e
- R32 Argumentos ::= Expresion ListaArgumentos
- R33 ListaArgumentos ::= \e
- R34 ListaArgumentos ::= , Expresion ListaArgumentos
- R35 Termino ::= LlamadaFunc
- R36 Termino ::= identificador
- R37 Termino ::= entero
- R38 Termino ::= real
- R39 Termino ::= cadena
- R40 LlamadaFunc ::= identificador (Argumentos)
- R41 SentenciaBloque ::= Sentencia
- R42 SentenciaBloque ::= Bloque
- R43 Expresion ::= (Expresion)
- R44 Expresion ::= opSuma Expresion
- R45 Expresion ::= opNot Expresion
- R46 Expresion ::= Expresion opMul Expresion
- R47 Expresion ::= Expresion opSuma Expresion
- R48 Expresion ::= Expresion opRelac Expresion
- R49 Expresion ::= Expresion oplqualdad Expresion
- R50 Expresion ::= Expresion opAnd Expresion
- R51 Expresion ::= Expresion opOr Expresion
- R52 Expresion ::= Termino

5 | Seminario de Solución de Problemas de Traductores de Lenguaje II

Capturas de pantalla

```
# Analisador_Sintactico
                                                                     → 省 Analisador_Sintactico.Form1.AnalizadorLexico
                                                                                                                                               using System.Text;
                   amespace Analisador Sintactico
                        eterencias
iblic partial class Form1 : Form
                                InitializeComponent();
                           3 referencias
class Pila...
                           3 referencias
class AnalizadorLexico...
        54 ®
                           5 referencias
class Reglas...
                           12 referencias
class ElementoPila...
                           d referencias
 101
                           class NoTerminal ...
                           class Estado ...
       788
796
                           1 referencia
private void btnLimpiar_Click(object sender, EventArgs e)...
                           1 referencia
private void <a href="https://private.ncb/bt/background-referencia">btnAnalizar_Click(object sender, EventArgs e)</a>...
                           private void mostrarSintactico(List<String> lista, List<String> salida)...
       821
828
829
```





7 Seminario de Solución de Problemas de Traductores de Lenguaje II
Conclusión
La realidad con esta actividad es que fue un poco compleja al momento de como tomar las reglas y la matriz pero una vez resolviendo eso la actividad fluyo por si sola, desde mi punto de vista aun que realmente es un poco tediosa pero sirve para darnos cuenta de todo el proceso que hace un compilador.
Bibliografía
No valida en este trabajo