

Universidad De San Carlos de Guatemala

Lenguajes Formales y de Programación

Ing. Otto Amílcar Rodríguez Acosta

Aux: Robín Navarro



Manual Técnico

Edgar Orlando Guamuch Zárate

201314632

Sección: A+

Objetivos

Generales:

- Desarrollar un analizador léxico con base en un autómata finito determinístico.

Específicos:

- Desarrollar un analizador léxico, mediante el método del árbol.
- detectar errores léxicos durante la ejecución del programa.
- generar un reporte de los errores detectados durante dicha ejecución.

Introducción:

La finalidad de este manual técnico es dar a conocer al lector la lógica con la que se ha desarrollado la aplicación incluyendo la descripción de todas las clases que se utilizaron para llevar a cabo el desarrollo de este software indicando así el IDE que se utilizó para la creación del mismo, su versión y los requerimientos que debe poseer en su ordenador al momento de realizar alguna edición en el software.

Este software realiza en análisis léxico a un archivo de entrada con extensión .lfp. los cuales permitirán que la aplicación pueda ser manejada por medio del lenguaje de aceptación y mediante estas instrucciones poder darle la funcionalidad a un juego de AHORCADO.

Requisitos del Sistema:

El programa se podrá ejecutarse media vez se cumplan con las siguientes características minimas:

1. Sistema operativo: Windows 7, Windows 8, Windows 8.1 y Windows 10
2. Espacio En Disco Disponible: 100Mb.
3. Memoria Ram: 1Gb.
4. Visual Studio 2013 o superior.

Descripción de la Solución:

Para la Solución de este proyecto lo primero que se realizó fue la creación de expresiones regulares que analizaran el texto de entrada para poder identificar que el texto de entrada no tuviera errores y que perteneciera al lenguaje para posteriormente realizar un AFD, para el análisis léxico se utilizó el siguiente algoritmo:

1. Árbol.
 - Agregar estado de aceptación.
 - Numerar Hojas del árbol.
 - Calcular si es Anulable.
 - Calcular Primera Posición.
 - Calcular Última Posición.
2. Llenar tabla de Siguiendo.
3. Llenar tabla de Estados.
4. Autómata(AFD).

Para el análisis sintáctico se realizaron los siguientes pasos:

1. Generación de una gramática LL(1)
2. Generación de tabla de primeros.
3. Generación de tabla de siguientes.

ANALISIS LEXICO

Expresiones Regulares a Implementar:

APERTURA DE ARCHIVO:

$E1 = /**$

CIERRE DE ARCHIVO:

$E2 = **/$

DIGITOS CON O SIN DECIMAL:

$E4 = D+(. D+)?$

ID Y RUTAS:

$E5 = L((L|D|_)*|:(L|D|_)+(\backslash(L|D|_)+)*.(L|D|_)+)$

Tabla de Siguientes:

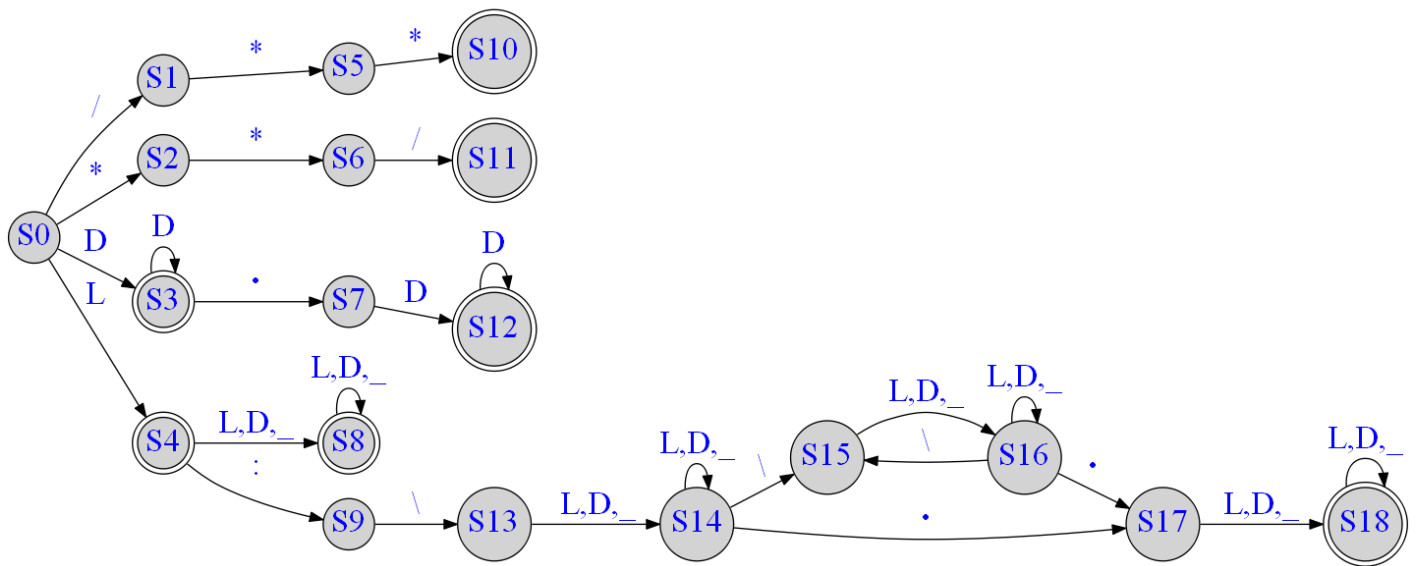
No	Alfabeto	Follow
1	/	2
2	*	3
3	*	4
4	#1	
5	*	6
6	*	7
7	/	8
8	#2	
9	D	9,10,12
10	.	11
11	D	11,12
12	#3	
13	L	14,15,16,17,30
14	L	14,15,16,30
15	D	14,15,16,30
16	_	14,15,16,30
17	:	18
18	\	19,20,21
19	L	19,20,21,22,26
20	D	19,20,21,22,26
21	_	19,20,21,22,26
22	\	23,24,25
23	L	22,23,24,25,26
24	D	22,23,24,25,26
25	_	22,23,24,25,26
26	.	27,28,29
27	L	27,28,29,30
28	D	27,28,29,30
29	_	27,28,29,30
30	#4	

Tabla de Estados:

#	Estados	/	*	D	.	L	_	:	\	Aceptación
S0	1,5,9,13	S1	S2	S3		S4				
S1	2		S5							
S2	6		S6							
S3	9,10,12			S3	S7					Aceptación
S4	14,15,16,17,30			S8		S8	S8	S9		Aceptación
S5	3		S10							
S6	7	S11								
S7	11			S12						
S8	14,15,16,30			S8		S8	S8			Aceptación
S9	18								S13	
S10	4									Aceptación
S11	8									Aceptación
S12	11,12			S12						Aceptación
S13	19,20,21			S14		S14	S14			
S14	19,20,21,22,26			S14	S17	S14	S14		S15	
S15	23,24,25			S16		S16	S16			
S16	22,23,24,25,26			S16	S17	S16	S16		S15	
S17	27,28,29			S18		S18	S18			
S18	27,28,29,30			S18		S18	S18			Aceptación

Autómata(AFD):

Autómata Finito Determinístico



ANALISIS SINTACTICO

ESTRUCTURA DE LA GRAMATICA LL(1):

S -> E

E -> INICIO E2

E2 -> INICIO E2

| Epsilon

INICIO -> /** configuracion L_CONT_CONF configuracion **/

L_CONT_CONF -> CONT_CONF L_CONT_CONF2

L_CONT_CONF2 -> CONT_CONF L_CONT_CONF2

| Epsilon

CONT_CONF -> # SCONT_C #

SCONT_C -> Juego LCONT_JUEGO Juego

| usuario LCONT_USUARIO usuario

| vocabulario LCONT_VOCABULARIO vocabulario

//MODULO VOCABULARIO

LCONT_VOCABULARIO -> CONT_VOCABULARIO LCONT_VOCABULARIO2

LCONT_VOCABULARIO2 -> CONT_VOCABULARIO LCONT_VOCABULARIO2

| Epsilon

CONT_VOCABULARIO -> #idioma ="id"# #Palabra="id"# #Longitud="numero"# #Pista1="id"#
#Pista2="id"#

//MODULO JUEGO

LCONT_JUEGO -> CONT_JUEGO LCONT_JUEGO2

LCONT_JUEGO2 -> CONT_JUEGO LCONT_JUEGO2

| Epsilon

CONT_JUEGO -> # SCONT_JUEGO #

SCONT_JUEGO -> niveles LCONT_NIVELES niveles

| tiempo LCONT_TIEMPO tiempo

| sonido LCONT_SA sonido

| ahorcado LCONT_SA ahorcado

LCONT_NIVELES -> CONT_NIVELES LCONT_NIVELES2

LCONT_NIVELES2 -> CONT_NIVELES LCONT_NIVELES2

| Epsilon

CONT_NIVELES -> # SCONT_NT #

LCONT_TIEMPO -> CONT_TIEMPO LCONT_TIEMPO2

LCONT_TIEMPO2 -> CONT_TIEMPO LCONT_TIEMPO2
| Epsilon

CONT_TIEMPO -> # SCONT_NT #

SCONT_NT -> facil = "numero"
| intermedio = "numero"
| dificil = "numero"

LCONT_SA -> # SCONT_SA #

SCONT_SA -> nombre= "id"# #ruta= "ruta"
| ruta= "ruta"# #nombre= "id"

//MODULO USUARIO

LCONT_USUARIO -> CONT_USUARIO LCONT_USUARIO2

LCONT_USUARIO2 -> CONT_USUARIO LCONT_USUARIO2
| Epsilon

CONT_USUARIO -> #nombre="id"#

Tabla de Primeros y Siguientes:

No	Primeros:	Columna2	Columna	Siguientes:
1	E	/**		\$
2	E2	/**, Epsilon		\$
3	INICIO	/**		/**, \$
4	L_CONT_CONF	#		configuracion **/
5	L_CONT_CONF2	#, Epsilon		configuracion **/
6	CONT_CONF	#		#, configuracion **/
7	SCONT_C	juego, usuario, vocabulario		#
8	LCONT_JUEGO	#		juego
9	LCONT_JUEGO2	#, Epsilon		juego
10	CONT_JUEGO	#		#, juego
11	SCONT_JUEGO	niveles, tiempo, sonido, ahorcado		#
12	LCONT_NIVELES	#		niveles
13	LCONT_NIVELES2	#, Epsilon		niveles
14	CONT_NIVELES	#		#, niveles
15	LCONT_TIEMPO	#		tiempo
16	LCONT_TIEMPO2	#, Epsilon		tiempo
17	CONT_TIEMPO	#		#, tiempo
18	SCONT_NT	facil, intermedio, dificil		#
19	LCONT_SA	#		sonido, ahorcado
20	SCONT_SA	nombre, ruta		#
21	LCONT_USUARIO	#		usuario
22	LCONT_USUARIO2	#, Epsilon		usuario
23	CONT_USUARIO	#		#, usuario
24	LCONT_VOCABULARI	#		vocabulario
25	LCONT_VOCABULARI	#, Epsilon		vocabulario
26	CONT_VOCABULARI	#		#, vocabulario

Tabla de Pre análisis

PRODUCCIONES	/**
E	INICIO E2
E2	INICIO E2
INICIO	/** configuracion L_CONT_CONF configuracion **/
L_CONT_CONF	
L_CONT_CONF2	
CONT_CONF	
SCONT_C	
LCONT_JUEGO	
LCONT_JUEGO2	
CONT_JUEGO	
SCONT_JUEGO	
LCONT_NIVELES	
LCONT_NIVELES2	
CONT_NIVELES	
LCONT_TIEMPO	
LCONT_TIEMPO2	
CONT_TIEMPO	

SCONT_NT	
LCONT_SA	
SCONT_SA	
LCONT_USUARIO	
LCONT_USUARIO2	
CONT_USUARIO	
LCONT_VOCABULARIO	
LCONT_VOCABULARIO2	
CONT_VOCABULARIO	

#
CONT_CONF L_CONT_CONF2
CONT_CONF L_CONT_CONF2
SCONT_C
CONT_JUEGO LCONT_JUEGO2
CONT_JUEGO LCONT_JUEGO2
SCONT_JUEGO
CONT_NIVELES LCONT_NIVELES2
CONT_NIVELES LCONT_NIVELES2
SCONT_NT
CONT_TIEMPO LCONT_TIEMPO2
CONT_TIEMPO LCONT_TIEMPO2
SCONT_NT
SCONT_SA
CONT_USUARIO LCONT_USUARIO2
CONT_USUARIO LCONT_USUARIO2
#nombre="id" #
CONT_VOCABULARIO LCONT_VOCABULARIO2
CONT_VOCABULARIO LCONT_VOCABULARIO2
#idioma ="id" # #Palabra="id" # #Longitud="numero" # #Pista1="id" # #Pista2="id" #

juego	usuario

--

--

juego LCONT_JUEGO juego	usuario LCONT_USUARIO usuario
-------------------------	-------------------------------

Epsilon

--

--

--

--

--

--

Epsilon

--

--

vocabulario	niveles
-------------	---------

--

--

vocabulario LCONT_VOCABULARIO vocabulario

--

niveles LCONT_NIVELES niveles

Epsilon
Epsilon

tiempo	sonido
tiempo LCONT_TIEMPO tiempo	sonido LCONT_SA sonido
Epsilon	

ahorcado	facil	intermedio

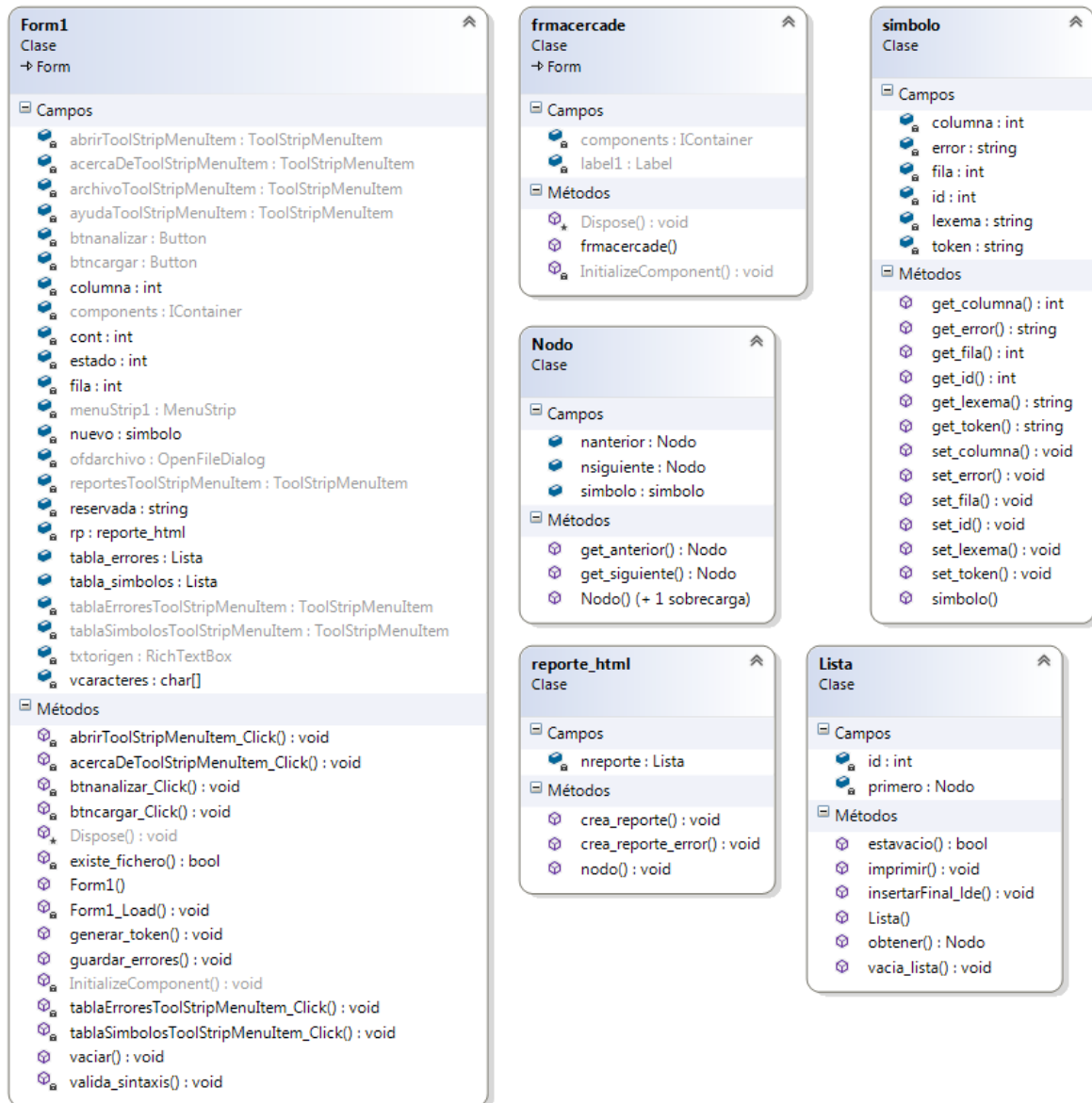
ahorcado LCONT_SA ahorcado

facil = "numero" intermedio = "numero"

dificil	nombre

ruta = "ruta"# #nombre= "id"

Diagrama de Clases:



Clases Utilizadas:

Clase	Descripción
simbolo.cs	Esta Clase crea un objeto símbolo, el cual contiene todos los campos que se guardan del análisis léxico.
Nodo.cs	Esta Clase crea los nodos con su respectivo apuntador en memoria
Lista.cs	Esta Clase genera una lista la cual es utilizada para almacenar la tabla de símbolos y de errores del texto analizado
reporte_html	Esta Clase genera un archivo .html para mostrar el contenido de la tabla de símbolos y de errores.
Form.cs	Clase Principal que contiene la programación del autómata para analizar lexicalmente el texto ingresado en la aplicación.
frmacercade.cs	Clase que contiene la información del desarrollador.

Librerías Adicionales Utilizadas:

Nombre	Descripción
System.Diagnostics	Proporciona acceso a procesos locales y remotos y le permite iniciar y detener procesos del sistema local.
System.IO;	Proporciona métodos estáticos para la creación, copia, eliminación, movimiento y apertura de un solo archivo, y ayuda en la creación de objetos FileStream .