

**Universidad de San Carlos de Guatemala**  
**Facultad de Ingeniería**  
**Escuela de Ciencias y Sistemas**  
**Lenguajes Formales y de Programación**



**MANUAL TÉCNICO**

**Nombre: Christopher Alexander Acajabon Gudiel**  
**Carnet: 201404278**

## **Métodos:**

- **Public Sub** metodo\_abrir()

Nos abre un archivo de texto para hacer analizado.

- **Public Sub** metodo\_guardar()

Nos guarda los cambios hechos en el archivo que abrimos.

- **Public Sub** metodo\_guardarcomo()

Nos guarda el archivo en una ruta especifica.

- **Public Sub** metodo\_salir()

Nos saca del analizador léxico.

- **Public Sub** metodo\_analizar()

En este método se encuentra la lectura de los componentes cargados o escritos que se introduzcan para luego operarlos.

- **Public Sub** Metodo2\_tokens(ByVal caracter\_1 As String, ByVal fila\_1 As Integer, ByVal columna\_1 As Integer, ByVal caracterAUX\_1 As String, ByVal caracterAUX\_2 As String)

En el siguiente método se envían todos los componentes léxicos agrupados para ser inspeccionados, comparando en busca de un patrón establecido.

- **Public Sub** Metodo\_Errores(ByVal examinar\_1 As String, ByVal fila\_1 As Integer, ByVal columna\_1 As Integer)

Método encargado del manejo de errores de caracteres no aceptados otorgando posición de fila y columna asignados y el carácter error.

- **Public Sub** MetodoHTML\_bueno()

En este método se manejan el listado de tokens para así mostrarlos en una pagina html.

- **Public Sub** MetodoHTML\_Errores()

En este método se manejan el listado de errores para así mostrarlos en una pagina html.

- **Public Sub** iniciarAnalisis()

Método que inicia todo el análisis léxico.

- `Public Sub` análisis\_sintactico()

Me analiza si las estructuras están bien formadas.

- `Public Sub` graficar()

Método que grafica en graphiz mostrando las clases.

- `Public Sub` llenarCola()

Método para ir verificando que atributos y métodos pertenecen a una clase.

- `Public Sub` llenarCola2(ByVal clase As Queue(Of String), ByVal atributos As Queue(Of String), ByVal metodos As Queue(Of String))

Método que recibe colas con los datos de una clase para poder generar el grafo.

- `Public Function` generarGrafo() As String

Método recursivo de tipo string, que me retorna la cadena de caracteres para generar el grafo.

- `Public Sub` abrirGrafo()

Método que me abre el grafo.

- `Public Sub` contarLlaves()

Método que me cuenta las llaves de cierre y de apertura, para que todos los bloques tengan cierre

## **Descripción de las clases:**

En la clase Form1 En la que se trabaja directamente la interfaz gráfica se manejan también los diferentes métodos léxicos y también el análisis sintáctico los cuales son utilizados para la agrupación de caracteres.

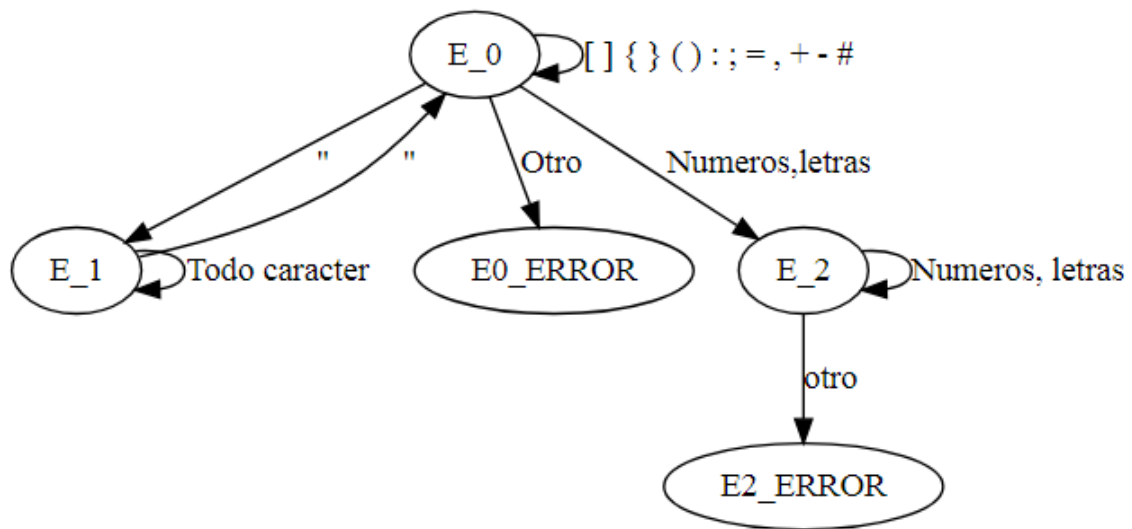
## **Descripción de métodos y funciones principales**

Los métodos principales en el programa son los métodos “Public Sub metodo\_analizar()” el método “Public Sub metodo2\_tokens()” y el meotdo “Public sun análisis\_sintactivo()” en estos tres métodos se encuentra el corazón del programa pues son los encargados de tomar cada carácter del texto de uno a uno y manejarlos, para luego ya agrupados compararlos si coinciden con algún patrón léxico establecido, eso a su vez se encarga de los procesos de almacenaje y operación siendo los principales métodos, para luego verificar si las estructuras están bien formadas.

## **Tabla de tokens**

NO.	LEXEMA	TIPO
1	{	Signo }
2	}	Signo }
3	Clase	Palabra reservada
4	Nombre	Palabra reservada
5	;	Signo ;
6	=	Signo =
7	Atributos	Palabra reservada
8	:	Signo :
9	(+)	Visibilidad
10	(-)	Visibilidad
11	(#)	Visibilidad
12	Metodos	Palabra reservada
13	Asociacion	Palabra reservada
14	Agregacion	Palabra reservada
15	Composicion	Palabra reservada
16	AsociacionSimple	Palabra reservada
17	[	Signo [
18	]	Signo ]
19	Color	Palabra reservada
20	Comentario	Palabra reservada
21	Texto	Palabra reservada
22	“	Signo comilla
23	,	Signo coma
24	—	Guion bajo

## Automata:



### EXPLICACION:

El estado E\_0 acepta los caracteres: [ ] { } ( ) : ; = , + - # , “ números, letras.

Si en el estado E\_0 viene cualquier otro carácter es Error.

Si en el estado estado E\_0 viene “ (abre comentario) pasa al estado E\_1.

El estado E\_1 acepta cualquier tipo de carácter.

Si en el estado E\_1 viene comilla (cierre de comentario) regresa al estado E\_0.

Si en el estado estado E\_0 vienen números o letras pasa al estado E\_2

El estado E\_2 acepta números y letras

Si en el estado estado E\_2 viene otro carácter es Error.

## Gramática:

$S_0 \rightarrow [ S_1$

$S_1 \rightarrow \text{clase } ] \{ S_3 \} S_0$   
 $\quad \quad \quad | \text{comentario } ] \{ S_4 \} S_0$   
 $\quad \quad \quad | \text{Asociacion } ] \{ S_5 \} S_0$

$S_3 \rightarrow [ S_6$

$S_6 \rightarrow \text{Nombre } ] S_7 S_3$   
 $\quad \quad \quad | \text{Atributos } ] \{ S_8 \} S_3$   
 $\quad \quad \quad | \text{Metodos } ] \{ S_9 \} S_3$

$S_8 \rightarrow (\text{visibilidad}) \text{IDE } S_{10} S_8$

$S_9 = S_8$

$S_5 \rightarrow \text{IDE} : S_{11} : \text{IDE} ; S_5$   
 $\quad \quad \quad | \text{IDE} : \text{IDE} ; S_5$

$\text{visibilidad} \rightarrow +$   
 $\quad \quad \quad | -$   
 $\quad \quad \quad | \#$

$S_7 \rightarrow = \text{IDE} ;$

$S_{10} \rightarrow ;$   
 $\quad \quad \quad | : \text{IDE} ;$

$S_4 \rightarrow [ S_{12}$

$S_{12} \rightarrow \text{Nombre } ] = \text{IDE} ; S_4$   
 $\quad \quad \quad | \text{Texto } ] = " \text{IDE} " ; S_4$

$S_{11} \rightarrow \text{Asociacion}$

$\quad \quad \quad | \text{Herencia}$   
 $\quad \quad \quad | \text{Asociacion Simple}$   
 $\quad \quad \quad | \text{Agregacion}$   
 $\quad \quad \quad | \text{Composicion}$

## **EXPLICACION:**

<b>s0</b>	<b>Abre la estructura con una llave seguido de un s1</b>
<b>s1</b>	Es el tipo del bloque que se va a definir, puede ser clase, comentario, asociacion, seguido de corchete que cierra y llave que abre
<b>s3</b>	son los bloques que se definen dentro de una clase, esta solo abre la estructura del bloque seguido de s6
<b>s4</b>	Abre lo que se va a definir dentro del bloque comentario, puede ser nombre o texto, seguido de s12
<b>s5</b>	Es la estructura de como se definiran las relaciones entre clases, y relacion de clase comentario.
<b>s6</b>	Es la estructura del bloque se definira dentro de la clase, puede ser nombre, atributos o metodos.
<b>s7</b>	Es como se va definir el nombre de la clase.
<b>s8</b>	Es la estructura de como se definiera la visibilidad con su nombre y tipo de dato, en este caso los atributos.
<b>s9</b>	Es la estructura de como se definiera la visibilidad con su nombre y tipo de dato, en este caso los metodos.
<b>s10</b>	Es por si un atributo o metodo no tiene tipo de dato.
<b>s11</b>	Son los tipos de asociaciones que hay entre las clases.
<b>s12</b>	Es lo que va dentro del bloque comentario, este solo abre lo que se va a definir, puede ser nombre o texto.