

**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**  
**CENTRO UNIVERSITARIO DE OCCIDENTE**  
**DIVISIÓN DE CIENCIAS DE LA INGENIERÍA**



**MANUAL DE TECNICO DE PROYECTO #1**

**LABORATORIO DE LENGUAJES FORMALES  
Y DE PROGRAMACIÓN**

**PRESENTADO POR:**

**UMAÑA DE LEÓN, WILLIAM RODRIGO**

**(201931448)**

**DOCENTE:**

**Ing. OLIVER ERNESTO SIERRA PAC**

**AUXILIAR**

**Bach. DANIEL GONZALES GONZALES**

**QUETZALTENANGO – QUETZALTENANGO – GUATEMALA**

**17-09-2020**

## Índice

<b>Índice</b>	<b>2</b>
<b>Introducción</b>	<b>3</b>
<b>Contenido</b>	<b>4</b>
Objetivos	4
Desarrollo de la lógica	4
Desarrollo de la interfaz	7

## **Introducción**

La aplicación consiste en la verificación que conlleva un analizador léxico, por lo que es un inicio de aplicación que a futuro servirá para la creación de un IDE de programación, la aplicación reconoce y pinta las palabras reservadas, al igual que lo hace con los números y símbolos que se requirieron en el documento de redacción del proyecto.

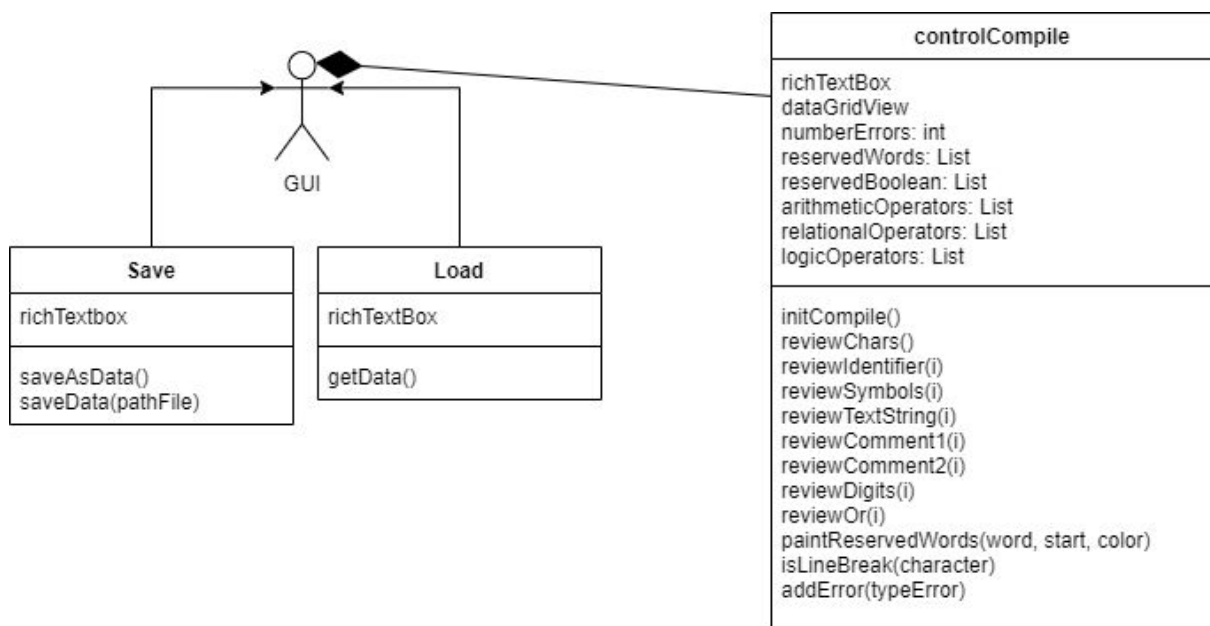
## Contenido

### Objetivos

- Manejo de archivos en C#
- Adaptación de un analizador léxico
- Aprender a programar autómatas

### Desarrollo de la lógica

Primero se realizó el diagrama de clases que se muestra en la siguiente imagen. Aunque cabe mencionar que se modificó en la creación de código ya que se agregaron algunos métodos.



Luego se tomó en cuenta la creación de los distintos autómatas, donde se utilizó el método de Thomson para realizar algunos ya que la combinación de todos era mucho más sencilla de lo que parecía.

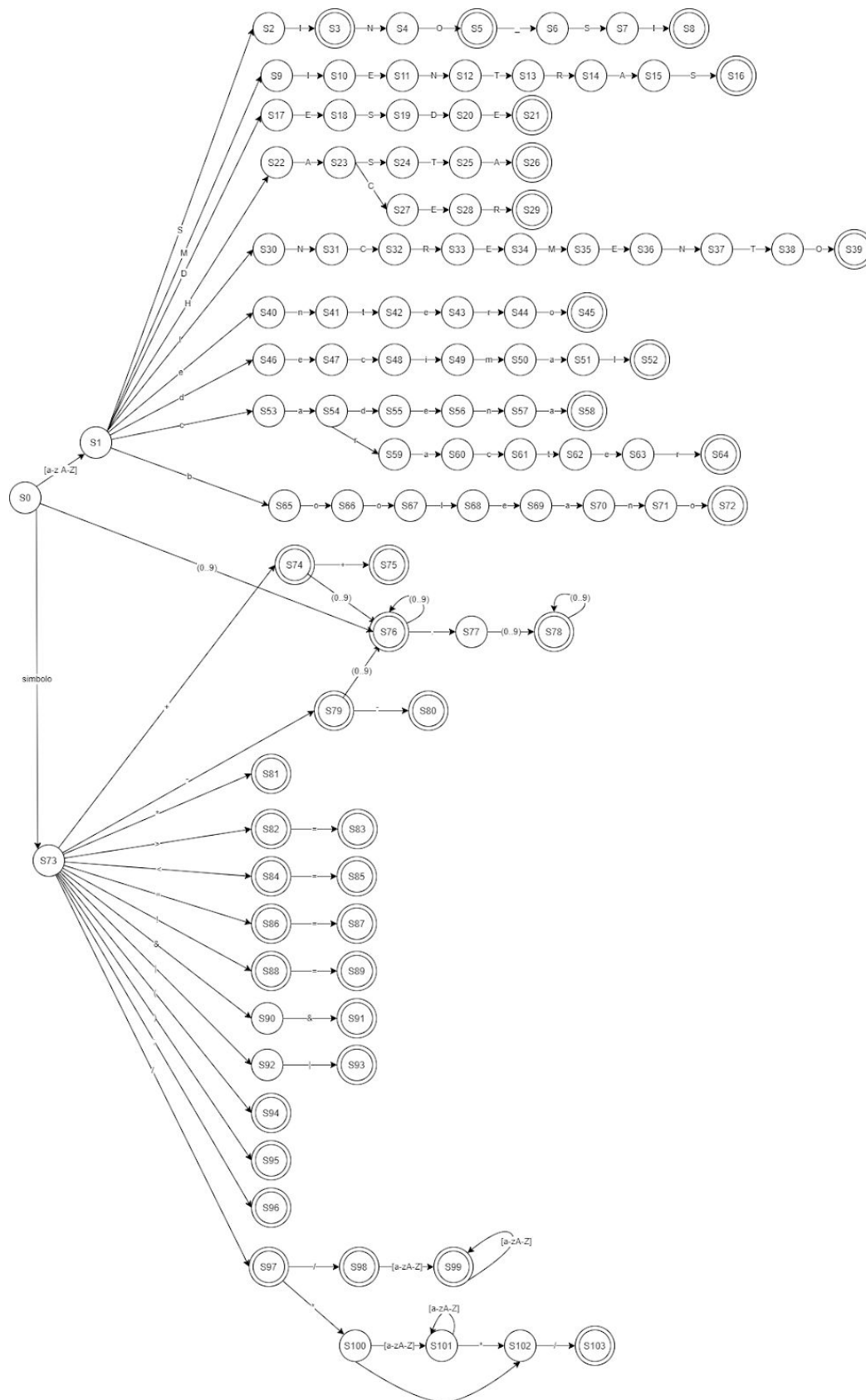
El lenguaje quedó descrito de la siguiente manera.

$\{[a-zA-Z],[1-9], |, !, ", \&, (, ), /, =, ;, +, -, *\}$

Cualquier otro símbolo está fuera del lenguaje

Los autómatas como se mencionó anteriormente se realizaron algunos con el método de Thomson ya que se realizó la unión del autómata final sin utilizar ningún método porque era mucho más sencillo.

El autómata final quedó de la siguiente manera:



[https://drive.google.com/file/d/1Ke1QOCUMEEvYlOMmIE1gAA\\_TNpLtxFuW/view?usp=sharing](https://drive.google.com/file/d/1Ke1QOCUMEEvYlOMmIE1gAA_TNpLtxFuW/view?usp=sharing)

El único autómata que se desarrolló con thomson fue el último autómata donde se puede observar que son 2 tipos de comentarios.

Para el desarrollo de este autómata primero se definió una expresión regular por separado de cada uno de los comentarios y quedaron de la siguiente manera:

`(/)([a-zA-Z])*`

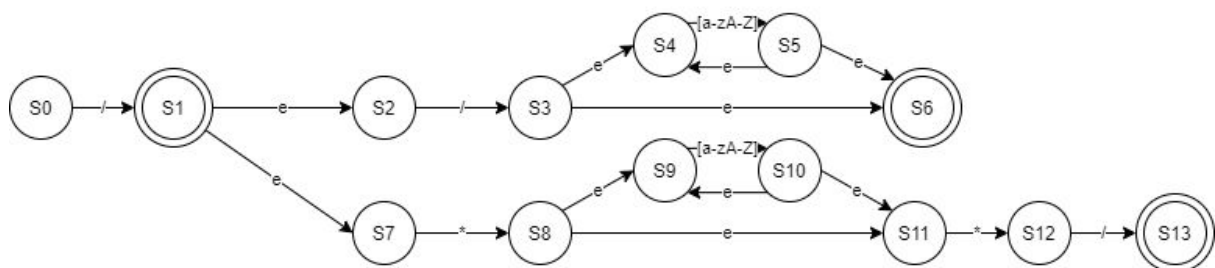
`(*)([a-zA-Z])*(*/)`

Luego se revisó y se compuso para que solo se obtuviera una expresión regular que define a los 2 comentarios en una, quedo de la siguiente manera:

`(/)((/)([a-zA-Z])*(*)([a-zA-Z])*(*/))`

Luego se realizó el modelo según la expresión regular anterior; El modelo quedó de la siguiente manera:

`(/)((/)([a-zA-Z])*(*)([a-zA-Z])*(*/))`

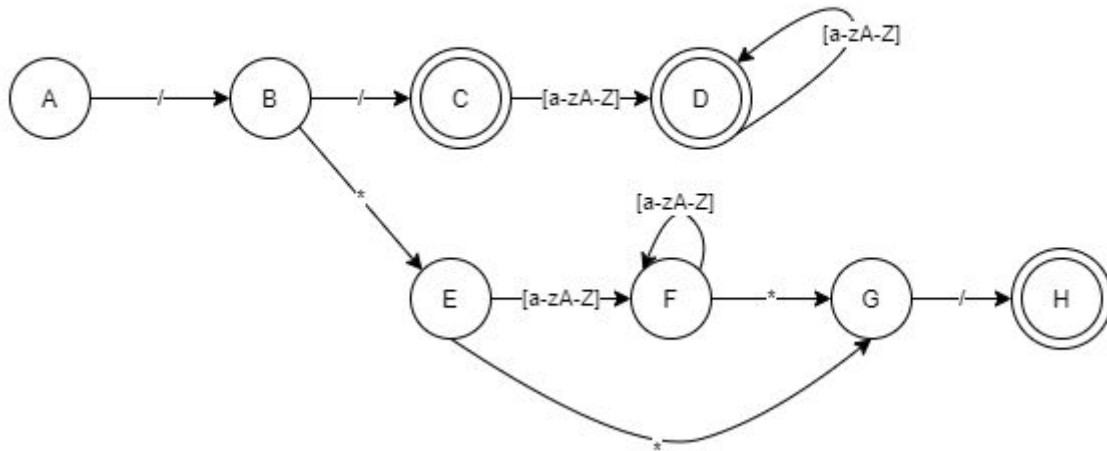


La tabla quedó de la siguiente manera:

F	e	/	*	[a-zA-Z]	
S0		S1			A
S1	S2,S7	S3	S8		B
S3	S4,S6			S5	C
S5	S4,S6			S5	D
S8	S9,S11		S12	S10	E
S10	S9,S11		S12	S10	F
S12		S13			G
S13					H

Y por último el autómata quedó de la siguiente manera:

$(/)((/)([a-zA-Z])^*(/)(([a-zA-Z])^*(/))$



## Desarrollo de la interfaz

Se tomó la idea de utilizar wpf para la creación de la interfaz pero surgieron algunos problemas con las cajas de texto y mejor se tomó la decisión de traspasar la interfaz gráfica a windows forms, por facilidad de uso en los botones y en las cajas de texto.

La interfaz consta únicamente de una página o ventana en donde el usuario puede interactuar completamente con las especificaciones requeridas en el documento.

Ejemplo de Ventana principal:



El uso del programa se describe en el manual de usuario.