

Lenguajes Formales y de Programación  
Kevin Golwer Enrique Ruiz Barbales  
201603009

# MANUAL DE TECNICO

# TABLA DE CONTENIDO

## Contenido

|                             |    |
|-----------------------------|----|
| Caratula _____              | 1  |
| Indice _____                | 2  |
| Introduccion _____          | 3  |
| Descripciones _____         | 4  |
| Listado de metodos _____    | 5  |
| Listado de metodos _____    | 6  |
| Descripcion Metodos _____   | 7  |
| Descripcion Metodos _____   | 8  |
| Descripcion Metodos _____   | 9  |
| Descripcion Metodos _____   | 10 |
| Automata _____              | 11 |
| Tabla de transiciones _____ | 12 |
| Automata Con estados _____  | 13 |

## Introducción

En el presente manual se encontrara una breve descripción del funcionamiento interno del analizador léxico , el listado de métodos creados y utilizados incluyendo el nombre el tipo y su modificador de acceso , también se encontrara una breve descripción del funcionamiento de cada método usado , el número de métodos usados , también el entorno de desarrollo en el cual fue creado y también el lenguaje de programación usado en este caso fue c#, en el caso de este programa se hizo uso de recursividad para mejorar el funcionamiento del mismo y evitar más métodos y agrandamiento del código del mismo.

## Descripciones

LEGUAJE EN EL CUAL FUE DESARROLLADO:

C#

ENTORNO DE DESARROLLO:

Visual Studio 2013

NUMERO DE CLASES

4

NUMERO DE MÉTODOS

34

## Listado de Métodos

| FOMR1   |           |                |
|---------|-----------|----------------|
| Public  | ArrayList | lineaCombo     |
| Public  | ArrayList | Llenalin       |
| Public  | ArrayList | Coincidencias  |
| Private | Void      | Ingresaventana |
| Private | Void      | pesta          |

| LECTURA |                |                      |
|---------|----------------|----------------------|
| Public  | Void           | abre                 |
| Public  | Void           | Lee                  |
| Public  | ArrayList      | Leeyanaliza2         |
| Public  | String         | Arregla              |
| Public  | Void           | Separación           |
| Public  | Void           | Inter                |
| Public  | Void           | Análisis             |
| Public  | Void           | Letra                |
| Public  | Void           | desconocido          |
| Public  | Void           | Numero               |
| Public  | Void           | Signo                |
| Public  | Void           | Asterisco            |
| Public  | Void           | Guion                |
| Public  | Void           | Diagonal             |
| Public  | Void           | Punto                |
| Public  | Arraylist      | getArray             |
| Public  | Boolean        | Comentario1          |
| Public  | Void           | Comenarion2          |
| Public  | Sting          | Tipo2                |
| Public  | Void           | Repetición           |
| Public  | Void           | Html                 |
| Public  | Void           | Imprime              |
| Public  | Arraylist      | Lexicol              |
| Pubic   | Arraylist      | Errores              |
| Public  | Boolean        | verificaLetra        |
| Public  | Boolean        | Verificanmero        |
| Public  | <u>Boolean</u> | Verificasicno        |
| Public  | Arraylist      | comenarionMultilinea |

# MANUAL TECNICO

|               |      |         |
|---------------|------|---------|
| COINCIDENCIAS |      |         |
| Public        | Void | obtiene |

## Descripción breve de métodos

### CLASE FORM1

Public ArrayList lineaCOMBO (int index)

Este método se encarga de obtener el texto de determinada venta en el analizador.

Public ArrayList llenalin (String a)

Este método se encarga de obtener el texto del textbox línea por línea

Public ArrayList Coincidencias ()

Método que se encarga de buscar coincidencias entre 2 pestanas devuelve un arraylist con la información solicitada.

Private void ingresaventana ()

Este método se encarga de llenar con texto los textbox de las pestanas

Private void pesta ()

Este método se encarga de crear las ventanas

### CLASE LECTURA

Public void abre ()

Este método se encarga de gestionar la apertura de un nuevo archivo

Public void lee (String ruta)

Este método se encarga de obtener las líneas del txt para ingresar al textbox

Public ArrayList leeyanaliza2(ArrayList ruta)

Este método se encarga de realizar el análisis léxico , separación de tokens y demás funciones.

Public String arregla(String arregla)

Este método se encarga de corregir las diagonales invertidas en la ruta del archivo que se obtiene de cualquier método.

Public void separacion(String lexema)

Este método se encarga de separar la línea que se obtiene del txt y separar individualmente cada token para luego ser analizado.

```
Public void inter()
```

Este método se encarga de enviar los tokens separados sin analizar a el autómata para se analizados y realizar sus respectivos procedimientos.

```
Public void analisis(String token, int pos)
```

Este método es el estado inicial del autómata, este se encarga de ver con cual de todas los estados cumplen el carácter a analizar.

```
Public void LETRA(String token , int pos)
```

Este método es el primer estado el cual hará el análisis de un identificador.

```
Public void DESCONOCIDO(String token, int pos)
```

Este método es el encargado de gestionar el estado de caracteres desconocidos.

```
Public void NUMERO(String token, int pos,int num)
```

Este método se encarga de analizar una cadena de números con punto incluido si lo tuviese.

```
Public void SIGNO(String token, int
```

Este método se encarga de gestionar los signos del lenguaje sin tomar en cuenta el punto y el asterisco.

```
Public void ASTERISCO(String token, int pos)
```

Este método se encarga de analizar este carácter.

```
Public void GUION(String token, int pos)
```

Este método se encarga de analizar si una letra es seguida de este para se concatenado.

```
Public void DIAGONAL(String token, int pos)
```

Este método se encarga de analizar el principio de un comentario multilinea.

```
Public void PUNTO(String token, int pos, int Contador)
```

Este método se encarga de analizar si lo primero que encuentro es un punto .



Public ArrayList getArray()

Este método se encarga de retornar los tokens sin repetir.

Public Boolean comentario1(String linea)

Este método se encarga de analizar si al empezar hay un comentario de una sola linea

Public void comentario2(int posicion,int bandera)

Este método se encarga de gestionar el análisis de comentario multilinea al insertar las líneas en el textbox nada mas.

Public String tipo2(Char cara)

Este método se encarga determinar el tipo de carácter que se está usando, regresa el tipo.

Public void Repeticion(ArrayList a1 ,ArrayList a2)

Este método se encarga revisar que no existan tokens repetidos en la lista de tokens para asi eliminarlos y dejar solo 1 solo tokens de ese tipo.

Public void html(ArrayList array)

Este método se encarga generar el código html con su ruta respetiva.

Public void imprime()

Este método se encarga imprimir en consola la información de los tokens.

Public ArrayList lexicoL(String totemp)

Este método se encarga de analizar error de tipo léxico.

Public ArrayList Errores(String totemp)

Este método se encarga gestionar los errores encontrados para incluirlos en el HTML

Public Boolean verificaLetra(Char letratokens)

Este método se encarga de verificar que el carácter sea una letra

Public Boolean verificaNumero(Char letratokens)

Este método se encarga de verificar que el carácter sea un numero

Public Boolean verificaSigno(Char letratokens)

Este método se encarga de verificar que el carácter sea una signo

Public ArrayList comenarioMultilinea (String linea, int valor)

Este método se encarga del análisis de comentario multilinea para el ingreso al textBox de las pestañas

# Autómata Finito Determinista para el análisis léxico

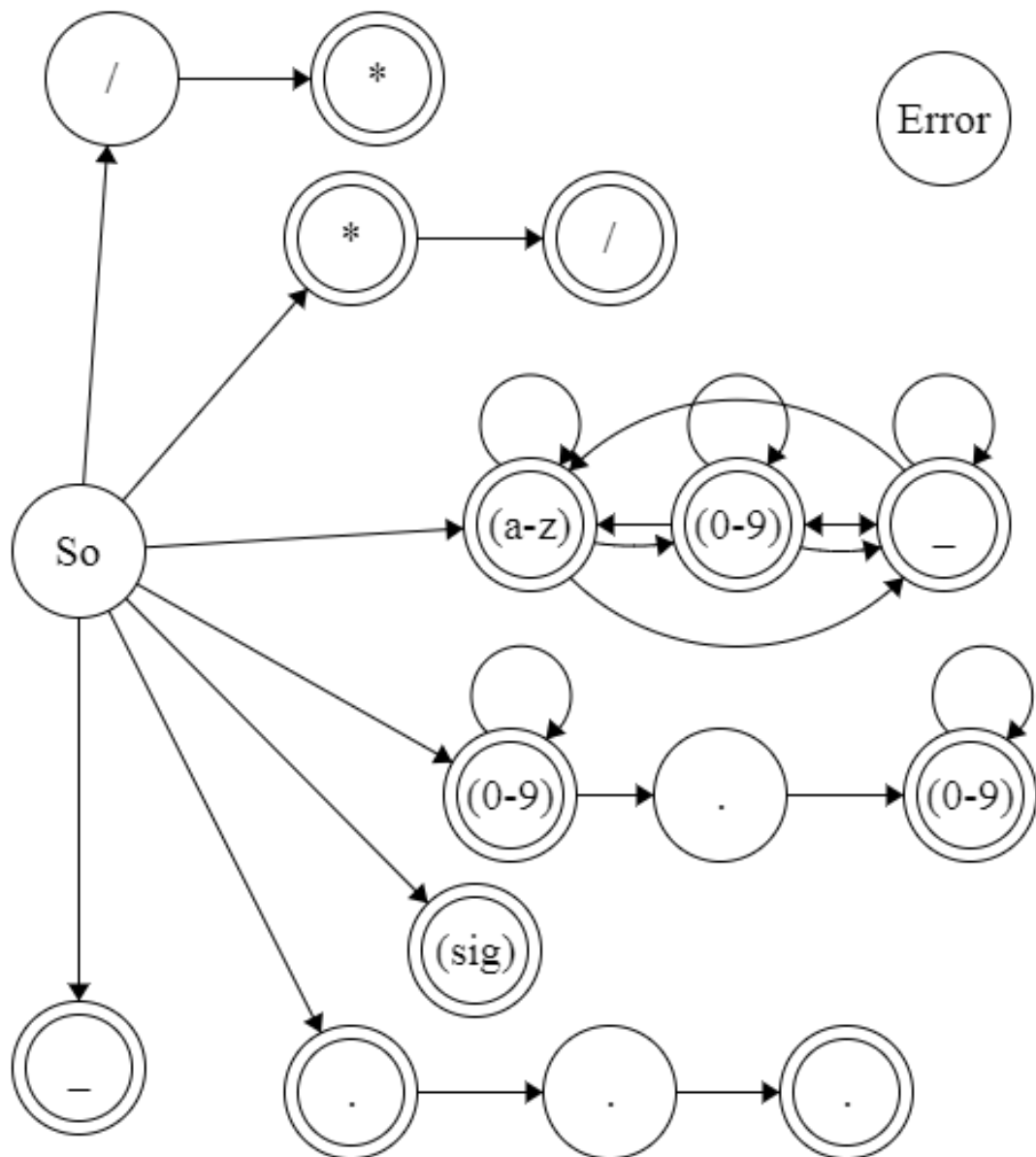


Tabla de transiciones

|     | Letra<br>[a-z] | Numero<br>[0-9] | Diagonal<br>/ | Asterisco<br>* | Punto<br>. | Signos | Guion<br>- | Estado Acepta |
|-----|----------------|-----------------|---------------|----------------|------------|--------|------------|---------------|
| S0  | S5             | S8              | S1            | S3             | S12        | S11    | S15        | No            |
| S1  | -              | -               | -             | S2             | -          | -      | -          | No            |
| S2  | -              | -               | -             | -              | -          | -      | -          | Si            |
| S3  | -              | -               | S4            | -              | -          | -      | -          | Si            |
| S4  | -              | -               | -             | -              | -          | -      | -          | Si            |
| S5  | S5             | S6              | -             | -              | -          | -      | S7         | Si            |
| S6  | S5             | S6              | -             | -              | -          | -      | S7         | Si            |
| S7  | S5             | S6              | -             | -              | -          | -      | S7S        | Si            |
| S8  | -              | S8              | -             | -              | S9         | -      | -          | Si            |
| S9  | -              | S10             | -             | -              | -          | -      | -          | No            |
| S10 | -              | S10             | -             | -              | -          | -      | -          | Si            |
| S11 | -              | -               | -             | -              | -          | -      | -          | Si            |
| S12 | -              | -               | -             | -              | S13        | -      | -          | Si            |
| S13 | -              | -               | -             | -              | S14        | -      | -          | No            |
| S14 | -              | -               | -             | -              | -          | -      | -          | Si            |
| S15 | -              | -               | -             | -              | -          | -      | -          | Si            |

