

Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ciencias y Sistemas  
Organización de Lenguajes y Compiladores 1 A



## **Manual de Usuario**

Fabian Esteban Reyna Juárez

Carné: 202003919

Guatemala, febrero 2022

## Contenido

<b>Introducción</b> .....	3
<b>Requerimientos</b> .....	3
<b>Manual de Usuario</b> .....	4
1. Ventana Inicial.....	4
2. Menú archivo.....	4
3. Archivos EXP (o archivo de entrada) .....	5
4. Analizar Entrada .....	7
5. Generar Autómatas .....	7
6. Reportes.....	8
6.1. Arboles .....	8
6.2. Siguietes .....	8
6.3. Transiciones .....	9
6.4. Autómata Finito Determinista (AFD).....	9
6.5. Autómata Finito No Determinista (AFND).....	9
6.6. Salidas.....	9
6.7. Errores.....	10

## Introducción

La aplicación permite realizar el método del árbol y método Thompson de manera automática solamente recibiendo la expresión regular en notación polaca (o prefija) con el objetivo de ayudar al estudiante en la elaboración de este. Además, puede comprobar cadenas en sus expresiones regulares, las cuales son analizadas por medio del autómata finito determinista resultante del análisis de la expresión regular.

Se posee una sección de reportes en formato PDF que tendrán la información de cada uno de los métodos con sus respectivos procesos.

## Requerimientos

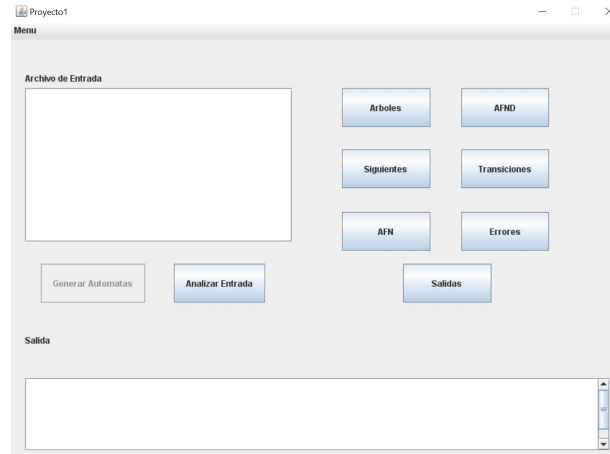
El sistema puede ser instalado en cualquier versión de Windows que cumpla con los siguientes requerimientos:

- **Versión de Java:** 1.8.1-291 o superior
- **JDK:** 1.8.0\_111 o superior
- **JRE:** 1.8.0\_291 o superior

# Manual de Usuario

## 1. Ventana Inicial

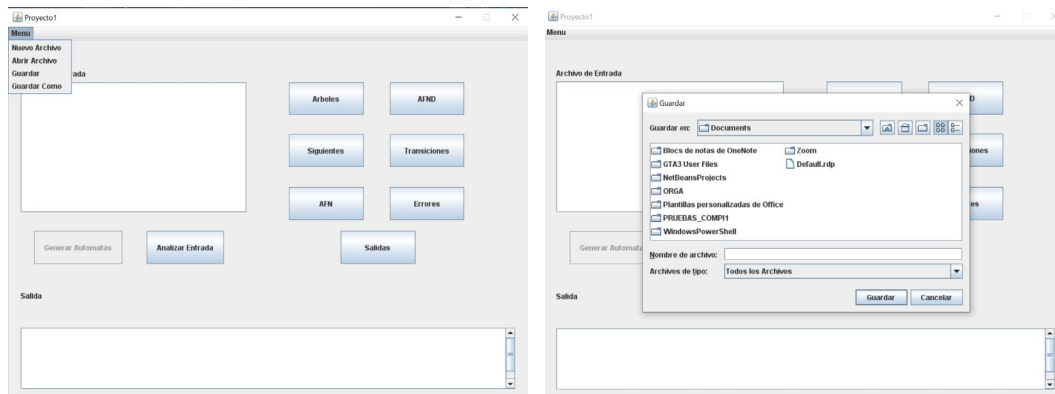
Al iniciar la aplicación se despliega una ventana que posee el área de texto del archivo de entrada y de las salidas, el menú y los botones que mostraran los reportes existentes.



## 2. Menú archivo

En la esquina superior izquierda se encuentra el menú de archivo el cual posee cuatro acciones, las cuales son:

- **Nuevo Archivo:** Permite crear un nuevo archivo en el área de entrada, vaciando todo el contenido que esta posea.
- **Abrir Archivo:** Permite abrir un archivo ya existente. La extensión aceptada por la aplicación es .EXP.
- **Guardar:** Permite guardar el archivo que este en el área de texto. En caso de no existir permite guardarlo abriendo el explorador de archivos para que el usuario lo almacene donde desee y con el nombre que desee. No es necesario colocarle la extensión ya que esta se agrega automáticamente.
- **Guardar Como:** Abre el explorador de archivos para que el usuario lo almacene donde desee y con el nombre que desee. No es necesario colocarle la extensión ya que esta se agrega automáticamente.



### 3. Archivos EXP (o archivo de entrada)

El archivo EXP inicia con llave de apertura y termina con llave de cierre, y posee 4 secciones principales entre estas, las cuales son:

- **Declaración de conjuntos:** La declaración de conjuntos sigue la estructura:

CONT: [Nombre Conjunto]->[Definición del Conjunto];

Donde:

- El nombre del conjunto debe iniciar con una letra y puede venir seguido de más letras, números o guion bajo.
- La definición del conjunto puede venir en rango o enlistando los caracteres de dicho conjunto. El rango valido será desde el ASCII 32 hasta el 125 omitiendo los ASCII de las letras y dígitos.
  - Rango: el rango debe venir con la secuencia [terminal] ~ [terminal]  
a ~ b
  - Lista: debe venir una lista de terminales separados por comas  
[terminal], [terminal], [terminal].....  
A, B, C, d
- **Declaración de Expresiones regulares:** Esta declaración sigue con la estructura:

[Identificador Expresión Regular]->[Expresión regular en prefijo];

Donde:

- El Identificador de la expresión regular debe iniciar con una letra y puede venir seguido de más letras, números o guion bajo.
- La expresión regular debe venir en prefijo (o notación polaca). Se puede colocar conjuntos dentro de la expresión regular, pero estos deben venir entre llaves. Y las cadenas deben venir entre comillas dobles.

Notación	Definición
.ab	Concatenación entre a y b
ab	Disyunción entre a y b
*a	0 o más veces la a
+a	1 o más veces la a
?a	0 o una vez la a

{ [Conjunto] }      Forma correcta de colocar un conjunto

" [Cadena] "      Forma correcta de colocar una cadena

- **Separadores:** Signos que separan la sección de definición de expresiones regulares de la sección donde se declaran las cadenas a escanear. Los símbolos son %%.
- **Declaración de cadenas a escanear:** Para definir la cadena que se quiere analizar con cierta expresión regular es necesario seguir el siguiente formato:

[Nombre Expresión Regular]: [Cadena];

Donde:

- El nombre de la expresión regular debe ser el mismo que el que en donde se define la expresión regular.
- La cadena debe venir entre comillas para poder ser válida.

```

{
  CONJ.letra->a~z;
  CONJ.digito->0~9;

  ExpReg1->.{letra}*{letra}{digito}{digito};

  %%

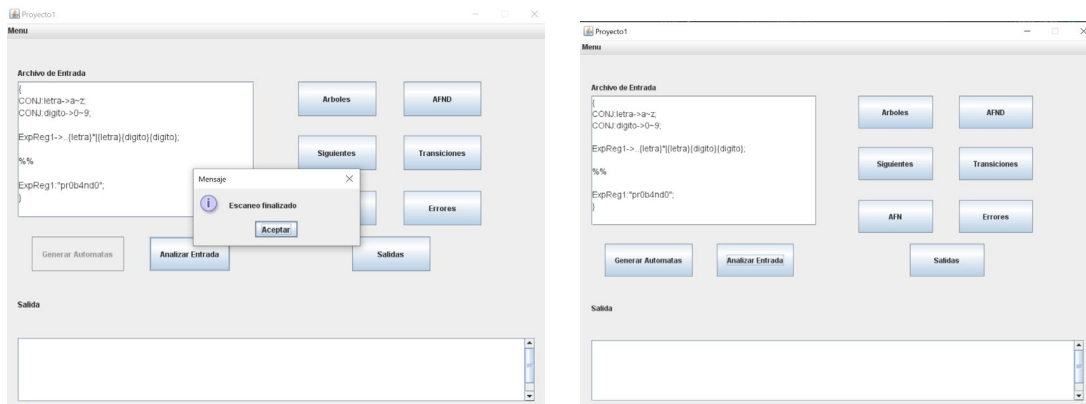
  ExpReg1:"pr0b4nd0";
}

```

Ejemplo de archivo EXP

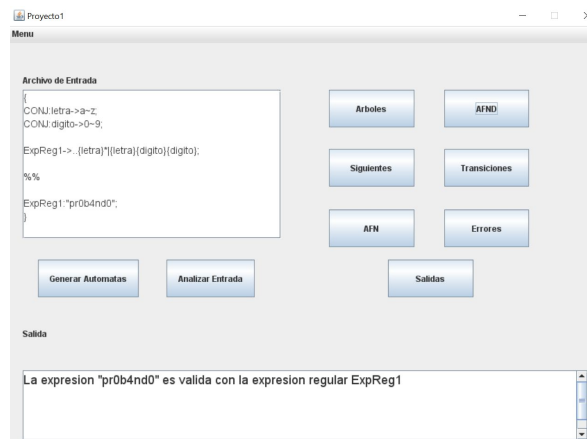
#### 4. Analizar Entrada

Para poder iniciar el proceso de generar los métodos solicitados, es necesario analizar las entradas, para ello la aplicación analiza el texto que se encuentre en el área de texto de entrada. Al finalizar este proceso se habilita el botón de generar autómatas.



#### 5. Generar Autómatas

Una vez habilitada esta opción se procede a la realización del método del árbol y de Thompson. Al finalizar se mostrará en la salida los resultados obtenidos del análisis de la cadena con la expresión regular.

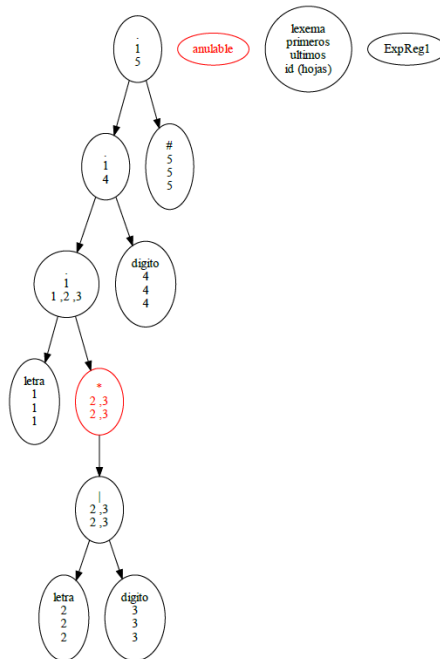


## 6. Reportes

Al finalizar la generación de autómatas es posible acceder a los distintos reportes utilizando los botones que se encuentran al lado derecho del área de entrada, los reportes que se generan son:

## 6.1.Arboles

Reporte PDF del árbol utilizado en el método del árbol, con sus funciones anulable, primera y última posición.



## 6.2. Siguientes

Tabla que muestra la tabla de siguientes correspondiente del método del árbol.

### Siguientes ExpReg1

Lexema	ID	Siguientes
letra	1	2,3,4
letra	2	2,3,4
digito	3	2,3,4
digito	4	5
#	5	



### 6.3. Transiciones

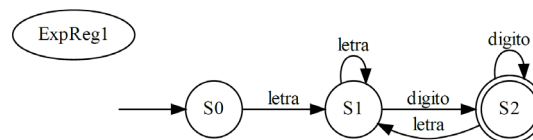
Tabla que muestra la tabla de transiciones correspondiente del método del árbol.

## Transiciones ExpReg1

Estado	letra	digito
S0{1}	S1	-
S1{2,3,4}	S1	S2
S2{2,3,4,5}	S1	S2

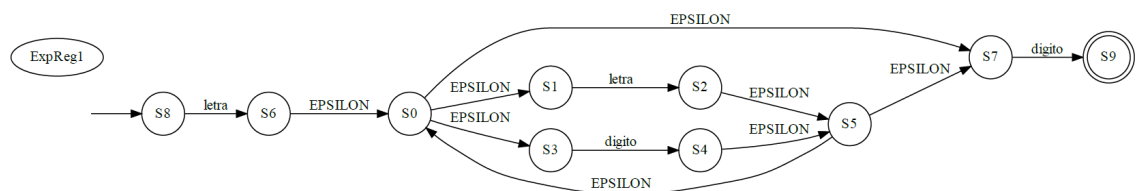
### 6.4. Autómata Finito Determinista (AFD)

Grafo que muestra el autómata finito determinista correspondiente del método del árbol.



### 6.5. Autómata Finito No Determinista (AFND)

Grafo que muestra el autómata finito no determinista correspondiente del método de Thompson.



### 6.6. Salidas

Archivo JSON que muestra los resultados del escaneo de las cadenas.

```
[
  {
    "ExpresionRegular": "ExpReg1",
    "Resultado": "Cadena Valida",
    "Valor": "pr0b4nd0"
  }
]
```

## 6.7. Errores

Archivo HTML que muestra los errores encontrados en el archivo de entrada.

**No se encontraron errores en el archivo de entrada :)**

Sin errores

### Reporte de Errores

#	Tipo de error	Descripcion	Linea	Columna
1	Lexico	Se esperaba cierre de cadena	6	48
2	Sintactico	%%%	8	1

Errores