

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS
PRIMER SEMESTRE 2022

OLC C



Laboratorio Organización de Lenguajes y Compiladores 1 – Proyecto 1

Manual de usuario

Nombre:	José David Panaza Batres
Carné:	202111478
Auxiliar:	José Diego Pérez Toralla
Fecha:	20 de marzo de 2023

Manual de usuario

Propósito

Este es un manual de usuario con el fin de resolver u orientar sobre cualquier duda que surja sobre el uso del programa en cuestión. Debido a que contiene funciones y botones los cuales pueden causar dudas. Sin embargo, todo lo que aparece en la ventana está estrictamente definido en el código fuente, sobre lo cual se puede consultar más adelante.

Alcance

Este programa está diseñado para que cualquier estudiante de una universidad cargue sus archivos OCL, el cual tiene una estructura, donde se definen conjuntos, expresiones y evaluaciones. Para que cree los respectivos análisis utilizando el método del árbol, el método de Thompson y mostrar todo lo que conlleva realizar dichos diagramas.

Descripción del sistema

Es un sistema completamente desarrollado en Java, haciendo uso de librerías como JFLEX y CUP. Es un sistema que puede leer archivos OLC. Analizarlos, crear objetos para después analizarlos, dependiendo el texto. Esta muestra los reportes en imágenes generadas en graphviz donde muestra la mayoría de los reportes. Tales como: arboles, autómatas, tablas entre otros.

Características principales

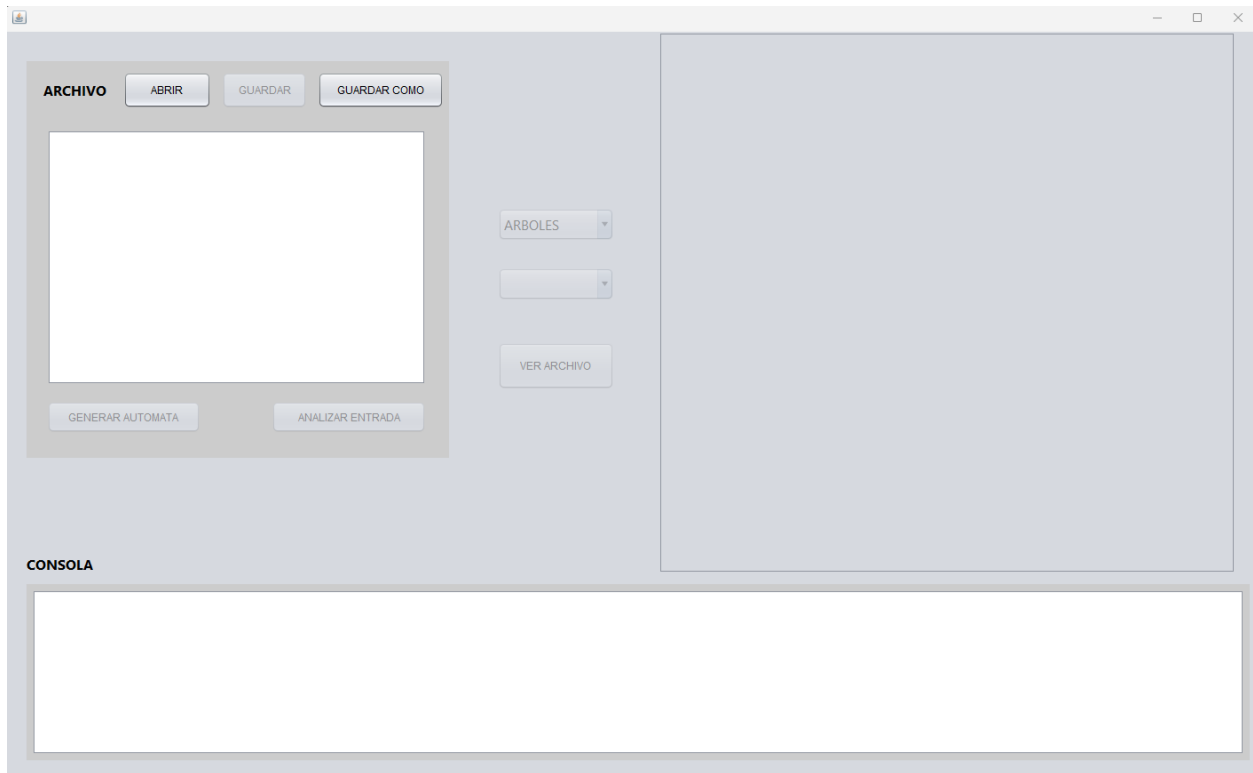
- Leer archivos OLC
- Analizar el texto a través JFLEX y CUP.
- Realizar el método del árbol y Thompson.
- Generar reportes de graphviz.

Requerimientos

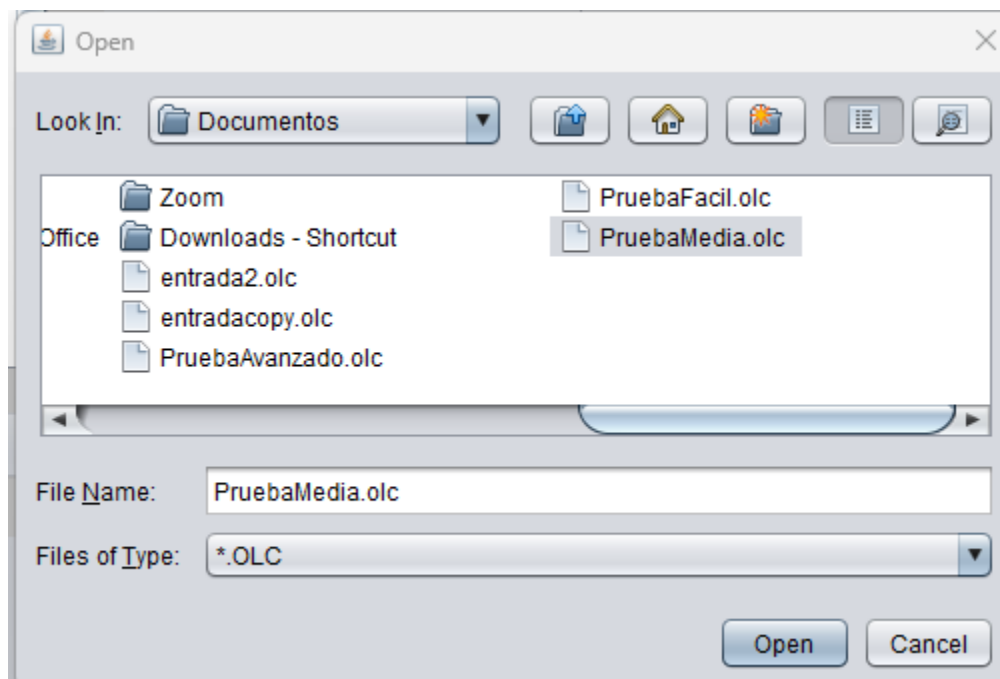
Contar con un archivo. OLC con un buen formato.

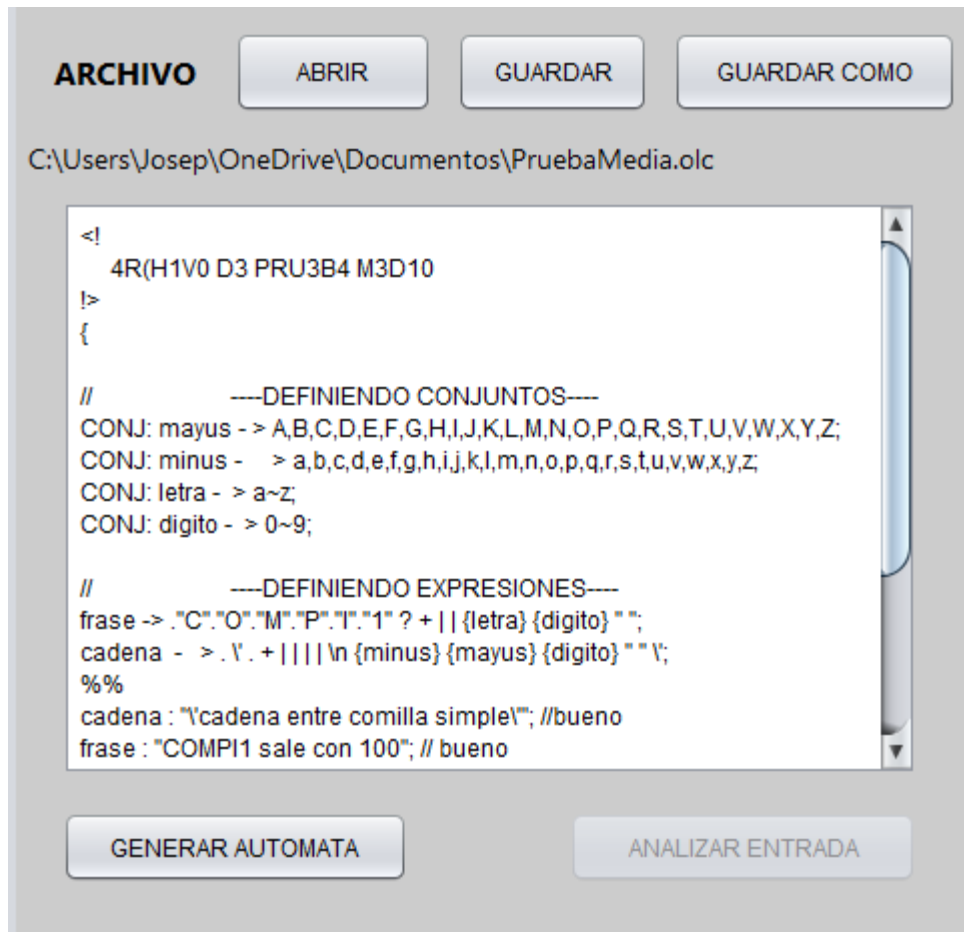
El programa cuenta con 1 pantalla principal, la cual tiene este aspecto.

1. Main Frame



2. Abrir un archivo. Abre un JFileChooser y pega el texto en el TextArea

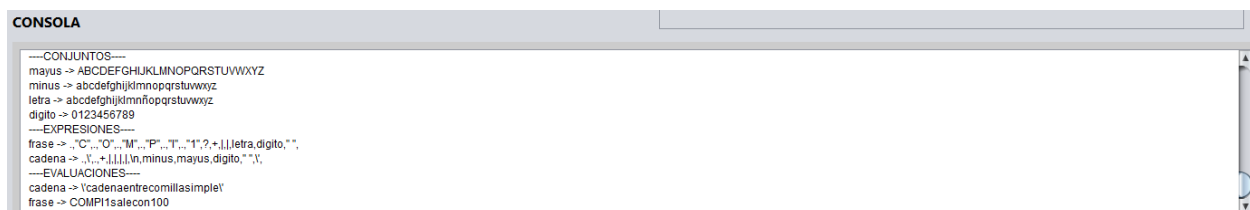




Se puede modificar el texto y guardar el archivo. Para guardar hay dos opciones:

3. Guardar: Sobre escribe el archivo y muestra un mensaje.
4. Guardar Como: Pregunta

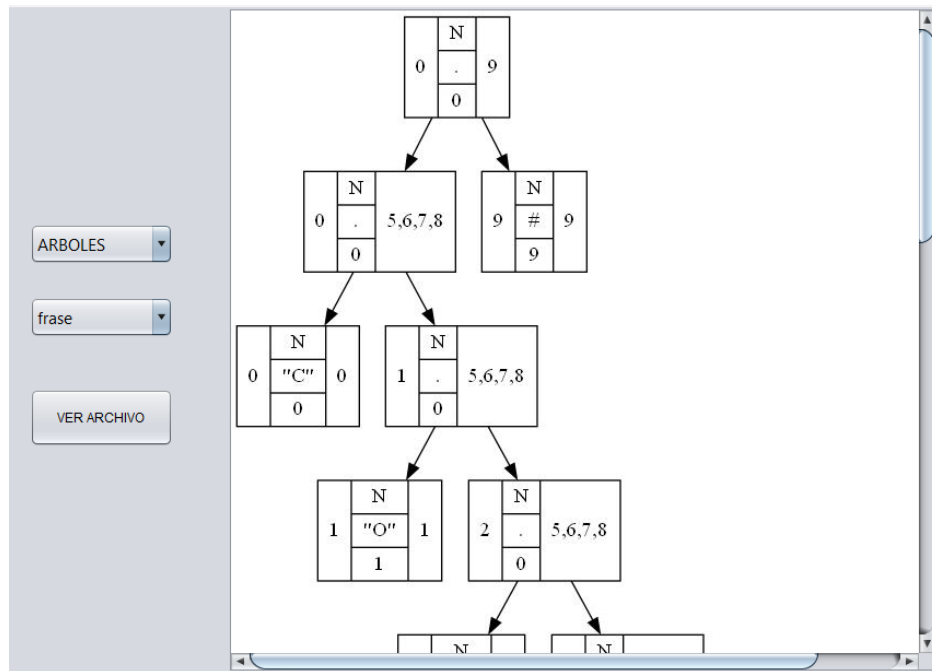
3. Se muestra en consola todos los caracteres aceptados. Los conjuntos, expresiones y las evaluaciones reconocidas.



5. Se habilita un menú donde se puede visualizar los reportes graphviz en una imagen. Están dos combobox.

Combobox superior: esta muestra opciones predefinidas tales como Arboles, Siguietes, Transiciones, AFD, AFND. Están representan las carpetas creadas.

Combobox inferior: esta muestra las expresiones definidas en el archivo, para poder elegir y ver lo que tiene cada una respectivamente.



6. Al presionar el botón de analizar entrada, evalúa las evaluaciones con las expresiones. Y muestra un mensaje en consola sobre el estado de cada una. Además, crea archivos JSON de salidas, los cuales indican cada estado de las evaluaciones.

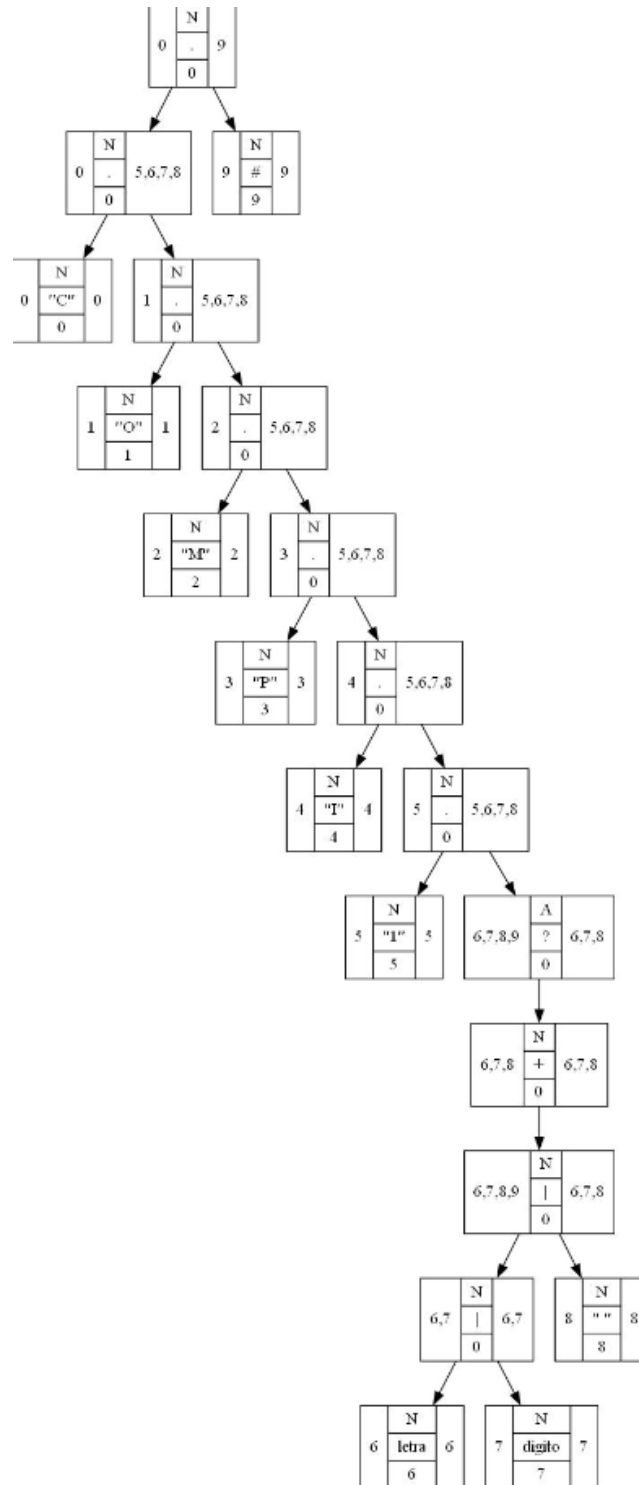
cadena - 'cadenaentrecomillasimple': correcta
frase - COMPI1salecon100: correcta

```
[
  {
    "Valor": "cadena",
    "ExpresionRegular": "\\ 'cadenaentrecomillasimple\\ '",
    "Resultado": "Cadena Valida"
  },
  {
    "Valor": "frase",
    "ExpresionRegular": "COMPI1salecon100",
    "Resultado": "Cadena Valida"
  }
]
```

Reportes

Ejemplo expresión: frase -> ."C"."O"."M"."P"."I"."1" ? + | | {letra} {digito} " " ;

1. Arboles:



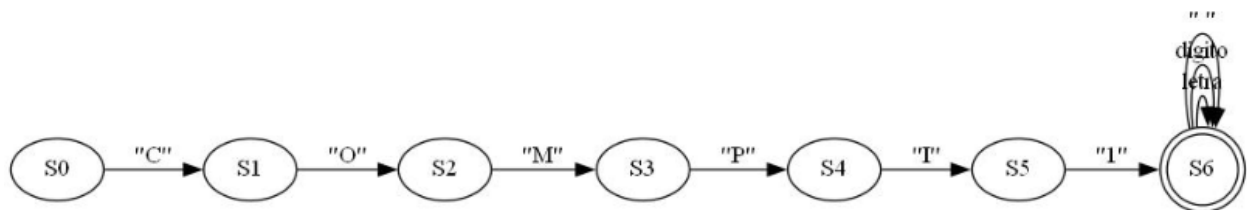
2. Siguietes:

Hojas		Siguietes
"C"	0	[1]
"O"	1	[2]
"M"	2	[3]
"P"	3	[4]
"I"	4	[5]
"1"	5	[6, 7, 8, 9]
letra	6	[6, 7, 8, 9]
digito	7	[6, 7, 8, 9]
" "	8	[6, 7, 8, 9]

3. Transiciones:

Estado	Terminales								
	"C"	"O"	"M"	"P"	"I"	"1"	letra	digito	" "
S0 [0]	S1	--	--	--	--	--	--	--	--
S1 [1]	--	S2	--	--	--	--	--	--	--
S2 [2]	--	--	S3	--	--	--	--	--	--
S3 [3]	--	--	--	S4	--	--	--	--	--
S4 [4]	--	--	--	--	S5	--	--	--	--
S5 [5]	--	--	--	--	--	S6	--	--	--
S6 [6, 7, 8, 9]	--	--	--	--	--	--	S6	S6	S6

4. AFD:



5. AFND:

