

Universidad de San Carlos de Guatemala

Ingeniería en sistemas

ORGANIZACION DE LENGUAJES Y COMPILADORES 1

Laboratorio

Manual Técnico

Emily Flor de Maria Cordon Ruiz

202111530

Lenguaje e IDE:

Lenguaje: Java

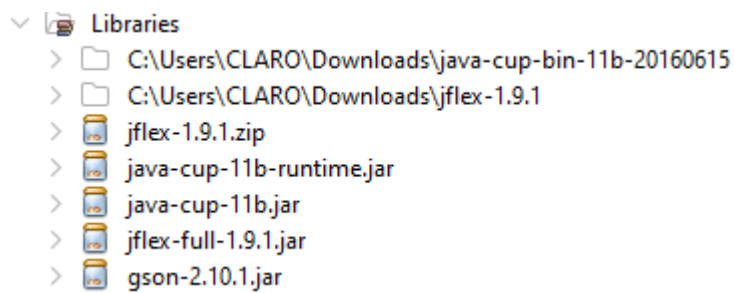
```
C:\Users\CLARO>java --version
java 19.0.2 2023-01-17
Java(TM) SE Runtime Environment (build 19.0.2+7-44)
Java HotSpot(TM) 64-Bit Server VM (build 19.0.2+7-44, mixed mode, sharing)
```

IDE: Netbeans

Proyecto1 Compi - Apache NetBeans IDE 17

Librerías Usadas:

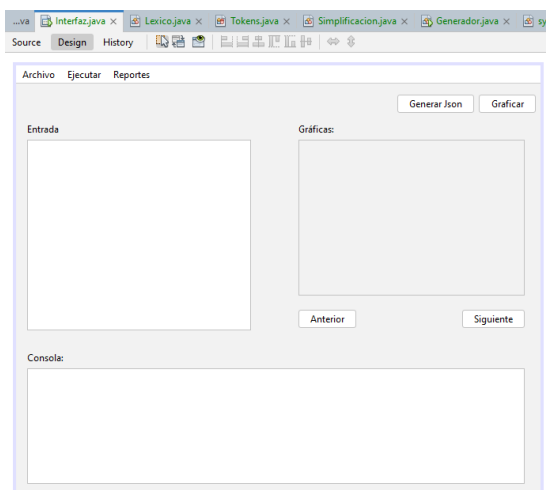
Se utilizaron 3 librerías:



Jflex, java cup y gson.

Funciones Importantes:

Interfaz.java



La interfaz contiene los métodos para poder abrir un archivo nuevo, crear un archivo nuevo y guardarlo, modificar el archivo, ejecutar los analizadores, realizar los reportes en html, realizar los json y graficar los conjuntos.

Léxico.flex:

```
L=[a-zA-Z]
D=[0-9]
ID={{L}({L}|{D}) *}
asciiChar=[\21-\x7E]
space=[ \t\r\n]+
%type Symbol
%%

"CONJ"      { actualizarPosicion(); return new Symbol(sym.Reservada, line, column, yytext()); }
"conjunto"  { actualizarPosicion(); return new Symbol(sym.Reservada, line, column, yytext()); }
"OPERA"     { actualizarPosicion(); return new Symbol(sym.Reservada, line, column, yytext()); }
"operacion" { actualizarPosicion(); return new Symbol(sym.Reservada, line, column, yytext()); }
"EVALUAR"   { actualizarPosicion(); return new Symbol(sym.Reservada, line, column, yytext()); }
{space}     { actualizarPosicion(); /* Ignorar espacios y saltos de línea */ }
"#"         { actualizarPosicion(); /* Ignorar comentarios */ }
"<!"        { actualizarPosicion(); /* Ignorar comentarios */ }
"!>"       { actualizarPosicion(); /* Ignorar comentarios */ }
":"         { actualizarPosicion(); return new Symbol(sym.DosPuntos, line, column, yytext()); }
"->"       { actualizarPosicion(); return new Symbol(sym.Flecha, line, column, yytext()); }
"~"         { actualizarPosicion(); return new Symbol(sym.Notacion, line, column, yytext()); }
";"         { actualizarPosicion(); return new Symbol(sym.PuntoYComa, line, column, yytext()); }
",,"        { actualizarPosicion(); return new Symbol(sym.Coma, line, column, yytext()); }
```

Los resultados del análisis léxico se muestran en el área de consola (TextAreaConsola). Estos incluyen tanto los tokens reconocidos correctamente como los errores léxicos encontrados. La salida proporciona información útil para los usuarios del sistema para identificar y corregir errores en la entrada antes de proceder a etapas posteriores del análisis, como el análisis sintáctico.

El analizador léxico es crucial para asegurar que la entrada al sistema esté correctamente estructurada antes de cualquier procesamiento adicional, como la evaluación de operaciones de conjuntos o la generación de gráficos basados en dichas operaciones.

Sintactico.cup:

```
};

terminal Reservada, DosPuntos, Flecha, Notacion, PuntoYComa, Coma, LlaveAbrir,
LlaveCerrar, ParentesisAbrir, ParentesisCerrar, Union, Interseccion,
Complemento, Diferencia, Simbolo, ID;

non terminal inicio, declaracionOperaciones, declaracion, operacion, conjDef, elementoLista, elemento, cor

precedence left LlaveAbrir, LlaveCerrar;

start with inicio;

inicio ::= LlaveAbrir declaracionOperaciones LlaveCerrar
        | PuntoYComa;

declaracionOperaciones ::= declaracionOperaciones declaracion
                           | declaracionOperaciones operacion
                           | declaracionOperaciones evaluar
                           | declaracion
                           | operacion
                           | evaluar
                           | PuntoYComa;

declaracion ::= Reservada DosPuntos ID Flecha conjDef PuntoYComa;
```

El resultado exitoso del análisis sintáctico es un árbol de análisis que representa la estructura correcta de la entrada de texto. Este árbol se utiliza para la evaluación de las operaciones de conjuntos y la generación de gráficos que representan visualmente estas operaciones. Los resultados se muestran en el área de consola (TextAreaConsola) y, si no se detectan errores, se continúa con las siguientes etapas del procesamiento.

El analizador sintáctico es una pieza clave del sistema, ya que asegura que las operaciones de conjuntos sean interpretadas correctamente de acuerdo con las reglas definidas, lo que permite un análisis y representación gráfica precisos.