

MANUAL DE USUARIO

Nombre: Gerson David
Otoniel González
morales

Carné: 202000774



INTRODUCCION

El siguiente programa está constituido sobre una interfaz gráfica cuyo objetivo es el de interpretar el lenguaje llamado DataForge, entre sus funciones se encuentra la de mostrar gráficas, operaciones aritméticas, asignación de variables entre otros.

REQUISITOS

Contar con el software JDK y el IDE Netbeans.
Windows 10 como mínimo.
Tener instaladas las librerías de JFreeChart y las
librerías de CUP y Jflex para java en caso el
programa las necesite.

INTERFAZ GRÁFICA

Como se puede observar la interfaz cuenta con múltiples espacios de trabajo entre ellos la barra de opciones que cuenta con las opciones de archivo y reportes, para editar texto se tiene el espacio que le pertenece a entrada, donde se puede escribir código, se tiene cuadro de consola donde aparecerán cualquier instrucción que necesite se impresa en ella, el área de Ver gráfica en la cual se visualizarán las graficas que sean creadas.



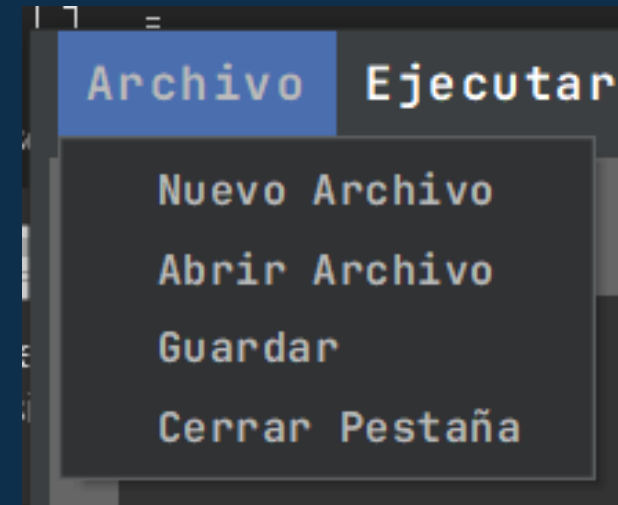
OPCION ARCHIVO

Para el menú de archivo se cuenta con múltiples opciones. Nuevo Archivo: esta opción crea una ventana en la cual se puede introducir código que se ejecutará.

Abrir Archivo: esta opción abrirá el explorador de archivos para poder elegir un archivo que ya esté creado y lo cargara a la interfaz.

Guardar: esta opción guarda el archivo en el que se esté trabajando, si el archivo no tiene nombre desplegará la opción de guardar con un nombre en nuestro dispositivo caso contrario de que el archivo ya existe en nuestra computadora esta opción solamente guardara los cambios que se hayan hecho.

Cerrar Pestaña: esta opción cerrará la pestaña en la que se esté actualmente y no guardará cambios por lo que es importante guardar antes de cerrar.



OPCION REPORTES

Este menú cuenta con las siguiente opciones.

Reporte de Tokens: crea y despliega un archivo html en el navegador predeterminado en el cual se mostrará una tabla con todo los tokens reconocidos por la interfaz, es decir, una tabla con todas la palabras reconocidas por el interprete.

Tabla de Simbolos: crea y despliega un archivo html en el navegador predeterminado en el cual se mostrará una tabla con todas las variables creadas junto con su valor, fila y columna en la que se encuentre.

Reportes

Reporte de Errores

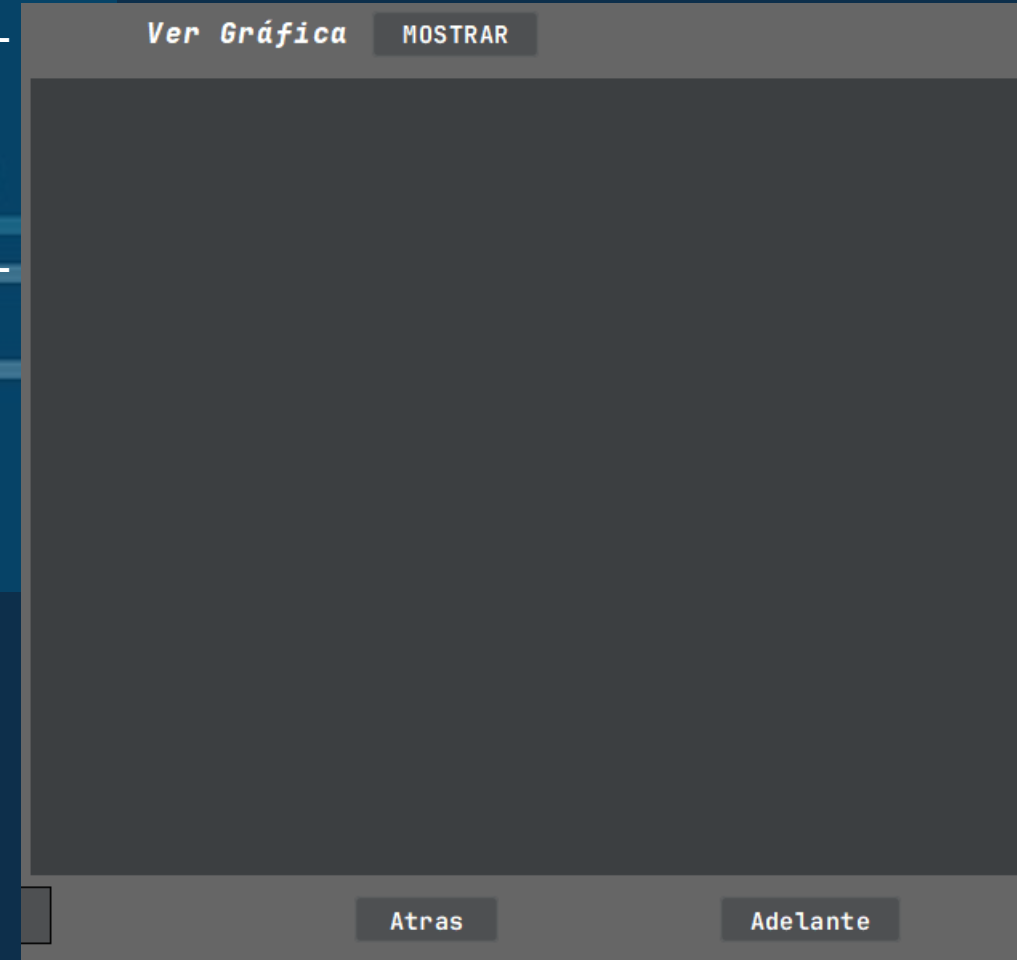
Reporte de Tokens

Tabla de símbolos

BOTON RUN Y ESPACIO DE GRAFICAS

El Botón RUN: este botón toma el texto cargado al espacio de entrada y la interpreta, cualquier instrucción que requiera que se imprima en consola aparecerá en el espacio de consola, cualquier instrucción que requiera de graficar aparecerá en el siguiente espacio.

Ver gráfica: en este espacio se puede observar el botón mostrar, el cual cargara todas la gráficas al espacio dedicado para este propósito, los botones en la parte inferior llamados Atrás y Adelante, Mostrarán las graficas y permitirán desplazarse a través de la lista de graficas para poder observarlas.



INSTRUCCIONES DEDICADAS A MOSTRAR EN CONSOLA

Las instrucciones que mostrarán un resultado en consola siguen la siguiente sintaxis:

```
"console : : print = <EXPRESION> , <EXPRESION> , ... end ;"
```

Esta instrucción mostraran en consola cualquier expresión que se le indica esto significa que puede imprimir por consola variables, arreglos, palabra o párrafos entre comillas dobles, números entre otros.

La siguiente instrucción se utiliza para el mismo propósito cuaya sintaxis es la siguiente:

```
console::column = <CADENA> -> <ARREGLO> end;
```

Esta instrucción imprime estrictamente un arreglo y solamente un arreglo.

ARREGLOS

Un arreglo es un conjunto de datos ya sean números, strings (entiéndase a todo dato entre comillas dobles), variables, operaciones aritméticas, otros arreglos. Todos estos datos encerrados entre corchetes como el siguiente ejemplo:

```
[1,2,3,4,5,6]
```

Todo arreglo que se imprima se mostrará en consola con un título y cada dato impreso con un salto de línea.

OPERACIONES ARITMÉTICAS

Toda operación aritmética que utiliza DataForge sigue la siguiente sintaxis:

OPERACIÓN(NUMERO, NUMERO)

Donde OPERACIÓN pueden ser de la siguiente lista: SUM, RES, MUL, DEV y MOD. Intuitivamente SUM calcula la suma de dos números, RES la resta entre dos números, MUL multiplicación entre dos números, DIV la división entre dos números y MOD es el resto de la división entre dos números.

OPERACIONES ESTADÍSTICAS

Las operaciones estadísticas siguen las siguiente
sintáxis:

OPERACIÓN ([ARREGLO])

Donde OPERACIÓN pertenece a las siguiente operaciones:
Media, Mediana, Moda, Varianza, Max y Min. Intuitivamente
Media calcula la media de ARREGLO donde arreglo ya fue
explicado con anterioridad, Max retorna el valor mas
grande de un arreglo y Min retorna el valor mas pequeño
de un arreglo, cabe decir que los arreglos que admiten
estas operaciones deben ser de tipo double, es decir,
números estrictamente números.

COMENTARIOS

Existen dos tipos de comentarios.

Comentarios de una línea: estos comentarios comienzan con “!” y todo lo que esté después de esa línea el interprete lo omitirá al momento de haber un salto de línea el comentario ya será interpretado.

Comentarios multilínea: estos comentarios deben de estar encerrados entre “<! !>” todo lo que este entre esos símbolos será omitido por el interprete.

GRAFICAS

Para crear una grafica se necesita de una sintaxis por grafica siendo el primer tipo de gráfica la grafica de Pie o Pastel que sigue la siguiente sintaxis:

```
graphPie(  
label::char[] = ["Uno", "Dos", "Tres"] end;  
values::double = [50, 30, 20] end;  
titulo::char[] = "Ejemplo Gráfica de Pie" end;  
EXEC grapPie end;  
) end;
```

La sintaxis es clara por lo que solo debe seguirse el orden, en caso por ejemplo la palabra label se encuentre de ultimo no se graficará y se encontrará con un error de sintaxis o sintactico.

GRAFICAS

Para crear una grafica se necesita de una sintaxis por grafica siendo el primer tipo de gráfica la grafica de Barras que sigue la siguiente sintaxis:

```
graphBar(  
  titulo::char[] = "Estudiantes" end;  
  ejeX::char[] = ["1 Parcial", "2 parcial", "Final"] end;  
  ejeY::double = [50, 30, 70] end;  
  tituloX::char[] = "Actividades" end;  
  tituloY::char[] = "Notas" end;  
  EXEC grapBar end;  
) end;
```

La sintaxis es clara por lo que solo debe seguirse el orden, en caso por ejemplo la palabra label se encuentre de ultimo no se graficará y se encontrará con un error de sintaxis o sintactico.