



# MANUAL DE USUARIO

ALLAN GÓMEZ

ORGANIZACION DE LENGUAJES Y COMPILADORES 1

# PROPÓSITO

La finalidad de esta documentación es garantizar el uso correcto y seguro del programa por parte del usuario, en el cual se podrá encontrar información sobre las funcionalidad del sistema, explicando paso a paso como acceder cada opción del sistema y para qué sirve cada una de ellas.

# ALCANCE

Implementar un software que permita al usuario realizar operaciones aritméticas y estadísticas, además de poder generar diversos gráficos a partir de una colección de datos.

# DESCRIPCIÓN DEL SISTEMA

La aplicación objetivo cumplir con los requerimientos solicitados por el curso de Organización de Lenguajes y Compiladores 1, perteneciente a la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala. Dichos requerimientos incluyen el análisis de los datos de entrada con un formato previamente establecido, la generación de reportes entre otras funcionalidades. Así como, desarrollar el lenguaje para los estudiantes de introducción a la programación y que puedan manejarlo de la mejor manera. Con dichos datos se despliegan varias funcionalidades para el manejo de la información. Como último la aplicación cuenta con una opción de generación de reportes en tablas y gráficos mediante la herramienta de graphviz.

# CARACTERÍSTICAS PRINCIPALES

- Fácil de utilizar.
- Es gratuito.
- Interfaz amigable.
- Tiempo de respuesta más rápido.
- Es un programa robusto.

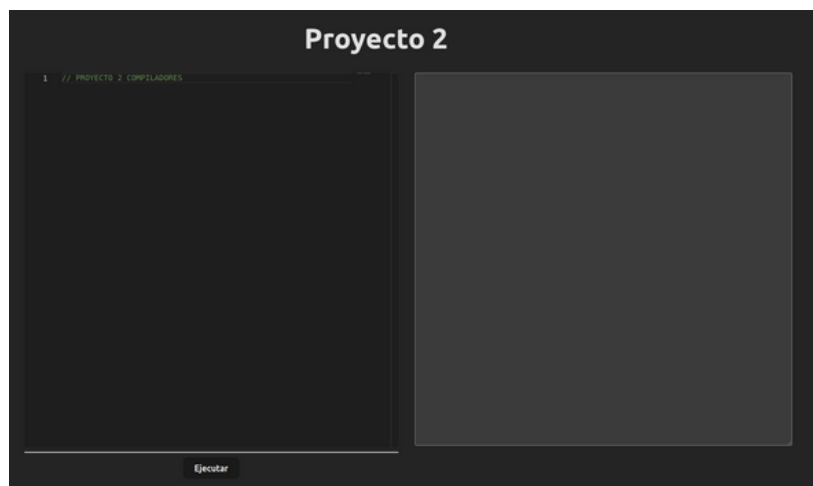
## REQUERIMIENTOS

- Versión reciente de nodejs
- Sistema operativo Windows XP o posterior, Linux o MAC.
- Evitar modificar parte parcial o total del código.
- Visual Studio Code o algun otro IDE

# FUNCIONALIDAD DEL SISTEMA

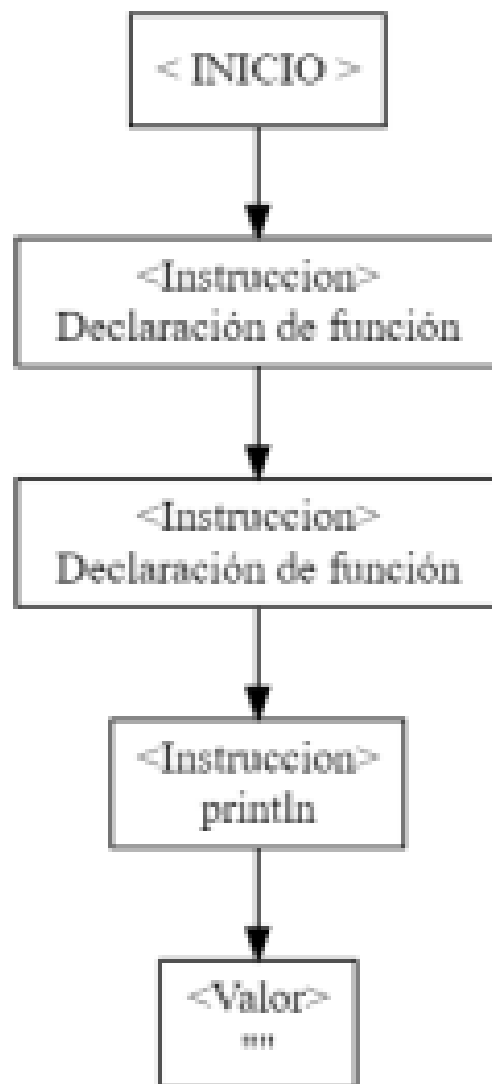
## Ejecutar:

Esta opción permite ejecutar el código en la consola de entrada, de manera que se muestre el resultado en la consola de salida y también la respectiva información el gráfico ast generado mediante Graphviz y los datos en caso de error de la tabla de símbolos y errores.



En la caja de texto que parece interfaz para editar código es donde se ingresaran todas las instrucciones que el usuario desee y al momento de darle al botón ejecutar se desplegara la información que se haya impreso en la caja de texto llamada "Consola". En caso de errores se desplegarán los debidos mensajes en la consola.

Al momento de ejecutar se genera el código de el árbol de sintaxis abstracta esto debido a si el usuario desease modificar algo del árbol. También se actualizará el gráfico del árbol de sintaxis abstracta en la página al momento de darle click al botón ejecutar.



## Tabla de símbolos

A su vez también se actualizará la tabla de símbolos que se encuentra en el despliegue de las ventanas para los reportes

#	Nombre	Tipo	Valor	Ámbito	Fila	Columna
1	metodo1	void	[object Object]	global	3	1
2	n	int	10	global -> figura1	7	9
3	figura1	void	[object Object]	global	7	1
4	cadenaFigura	string	..... ...	global -> figura1	8	9
5	i	double	11	global -> figura1	9	9
6	j	double	31	global -> figura1 -> Cuerpo while	14	13
7	absolutoi	double	15	global -> figura1 -> Cuerpo while -> Cuerpo while	17	17
8	absolutoj	double	15	global -> figura1 -> Cuerpo while -> Cuerpo while	19	17

Y en caso de existir errores se actualizará la tabla de errores que se encuentra también en el menú de reportes

#	Tipo	Error	Fila	Columna
1	Error semántico	no existe una variable con ese nombre	33	18
2	Error semántico	Asignación incorrecta, la variable con nombre 'x1' es de tipo [double] y se le esta tratando de asignar un tipo [error]	0	1

## Despliegue de menu

- Abrir: Permite abrir un archivo previamente creado para poder seguir editándolo
- Guardar: Permite guardar el archivo que está siendo editado.
- Guardar como: Permite guardar el archivo que está siendo editado, con otro nombre o extensión del archivo.

## Despliegue de reportes

- Tabla de simbolos: Permite visualizar la tabla de simbolos
- Tabla de errores: Permite visualizar la tabla de errores
- ASTR : Permite visualizar el gravo del AST