

MANUAL DE USUARIO

Ruth Nohemy Ardón Lechuga
Carnet: 201602975

INTERFAZ GRÁFICA

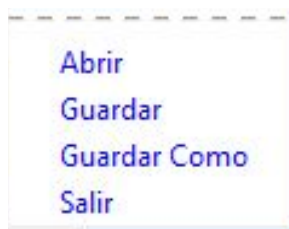
Interfaz general del editor August IDE



Se compone de los siguientes componentes:

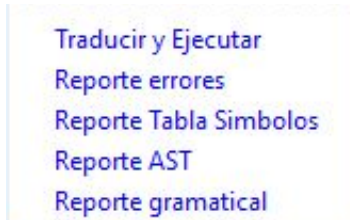
- Barra de menú lateral superior, contiene los submenús: Archivo, ejecutar, Opciones y ayuda.
- Área de edición de texto, en esta sección se ingresa el código a ejecutar. Se compone de una barra con líneas y un scrollbar de navegación.
- Tab panel inferior, se compone de cuatro secciones principales: Consola, código 3D, errores y tabla de símbolos.

MENÚ ARCHIVO



- Abrir: Opción para abrir un documento.
- Guardar: Opción para guardar un documento.
- Guardar Como: Opción para guardar un archivo aún no existente en una nueva ruta.
- Salir: Cerrar el IDE.

MENÚ EJECUTAR



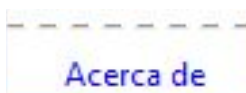
- Traducir y Ejecutar: Traduce el código que se ingresa en C minus a Código Augus y procede a ejecutar el código Augus.
- Reporte errores: Se producirá el archivo de errores léxicos, sintácticos y semánticos.
- Reporte tabla de símbolos: Se producirá el archivo de tabla de símbolos.
- Reporte AST: Se producirá el reporte de ast en dependencia de la ejecución que se haya realizado.
- Reporte gramatical: Se producirá el reporte gramatical en dependencia de la ejecución que se haya realizado.

MENÚ OPCIONES



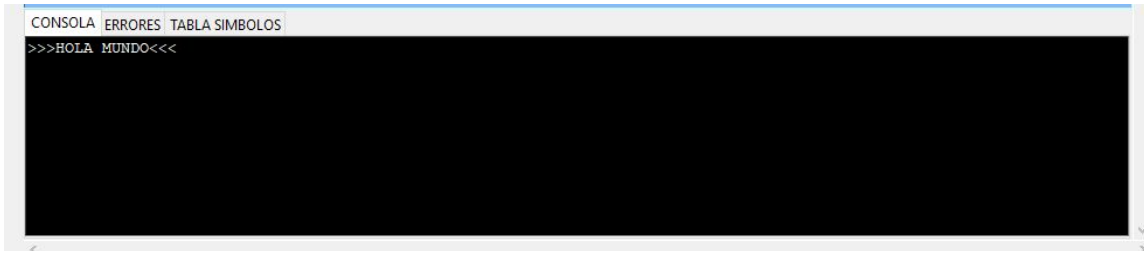
- Font: Abre un menú para poder cambiar el estilo de fuente del editor de texto.
- Color Fondo: Da opción para cambiar el fondo del editor de texto.

MENÚ ACERCA



- Acerca de: Muestra información relevante en torno al editor.

SECCIÓN DE CONSOLA



SECCIÓN DE ERRORES

CONSOLA ERRORES				TABLA SIMBOLOS
Tipo	Línea	Columna	Error	
SEMANTICO	3	7	El identificador \$t1 no se ha encontrado.	
SEMANTICO	3	1	No se ha podido procesar la información del print.	

REPORTE DE ERRORES

Errores

Tipo	Mensaje	Linea	Columna
SEMANTICO	El identificador \$t1 no se ha encontrado.	3	7
SEMANTICO	No se ha podido procesar la información del print.	3	1

REPORTE TABLA DE SÍMBOLOS

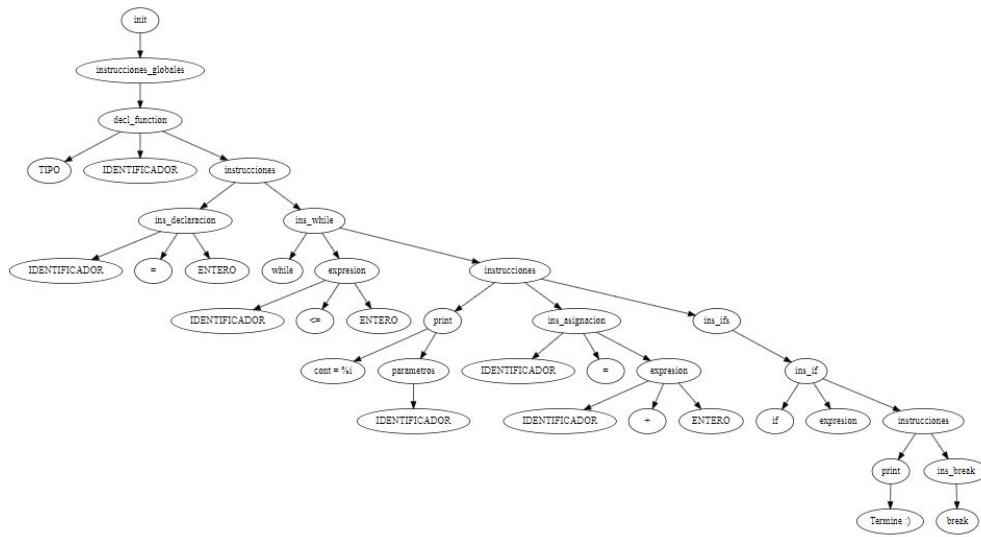
Simbolos

Identificador	Tipo	Dimension	Valor	Linea	Columna	Ambito
---------------	------	-----------	-------	-------	---------	--------

Etiquetas

Identificador	Tipo	Linea	Columna
main	CONTROL	1	5

REPORTE AST



REPORTE GRAMATICAL

Reporte Gramatical

Produccion	Regla Semántica
tipo -> int	t[0] = t[1]
tipo -> int	t[0] = t[1]
expresion -> ENTERO	t[0] = Expresion(t[1],None,TIPO_OPERACION.ENTERO,t.lineno(1),find_column(entrada, t.slice[1]))
asignacion_dec -> IDENTIFICADOR sig_asig expresion	t[0] = Declaracion(TIPO_DATO.ENTERO,t[1],None,t[3],t.lineno(1),find_column(entrada, t.slice[1]))
lista_asignaciones_dec -> asignacion_dec	t[0] = [t[1]]
declaracion -> tipo lista_asignaciones_dec PTCOMA	t[0] = t[2]
instruccion -> declaracion PTCOMA	t[0] = t[1]
instrucciones -> instruccion	t[0] = t[1]
expresion -> IDENTIFICADOR	t[0] = Expresion(t[1],None,TIPO_OPERACION.IDENTIFICADOR,t.lineno(1),find_column(entrada, t.slice[1]))
expresion -> ENTERO	t[0] = Expresion(t[1],None,TIPO_OPERACION.ENTERO,t.lineno(1),find_column(entrada, t.slice[1]))
expresion -> expresion <= expresion	t[0] = Expresion(t[1],t[3],<=,t.lineno(2),find_column(entrada, t.slice[2]),t[2])

ENTRADA, CÓDIGO Y FLUJO

Editor Minor C

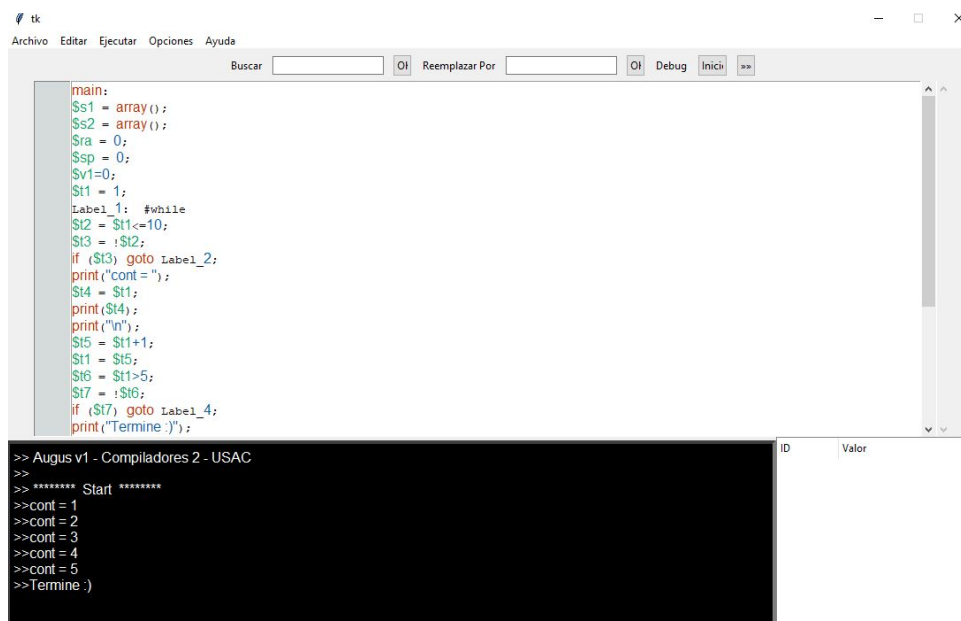


The screenshot shows the Minor C editor window. The main text area contains a C program with the following code:

```
1 int main()
2 {
3     int cont = 1;
4     while(cont<=10){
5         printf("cont = %i\n",cont);
6         cont++;
7
8         if(cont > 5){
9             printf("Termine :");
10            break;
11        }
12    }
13 }
14
15
```

Below the code editor, there is a console window with the text "analisis realizado..." and a tab bar with "CONSOLA", "CÓDIGO 3D", "ERRORES", and "TABLA SIMBOLOS".

Ejecutor de Augus



The screenshot shows the Augus executor window. The main text area contains assembly code with the following code:

```
main:
$s1 = array();
$s2 = array();
$ra = 0;
$sp = 0;
$V1=0;
$t1 = 1;
Label_1: #while
$t2 = $t1<=10;
$t3 = !t2;
if ($t3) goto Label_2;
print("cont = ");
$t4 = $t1;
print($t4);
print("\n");
$t5 = $t1+1;
$t1 = $t5;
$t6 = $t1>5;
if ($t6) goto Label_4;
print("Termine :");
```

Below the code editor, there is a console window with the text "Augus v1 - Compiladores 2 - USAC" and a table with columns "ID" and "Valor". The console output shows the execution of the program, including the values of the variables and the output of the print statements.

LENGUAJE

Operadores

1	}	26	!
2	.	27	&&
3	=>	28	
4	++	29	~
5	--	30	&
6	+	31	
7	-	32	^
8	*	33	!=
9	/	34	?
10	%	35	;
11	<<	36	:
12	>>	37	.
13	<=	38	(
14	<	39)
15	>	40	[
16	==	41]
17	+=		
18	-=		
19	*=		
20	/=		
21	%=		
22	<<=		
23	>>=		
24	&=		
25	=		

Instrucciones

1. Etiquetas: identificadores de sección seguidos por el signo dos puntos, para ser utilizada la declaración goto
2. Declaraciones de expresiones
3. If-if else-else: *if (condición) declaración else declaración*. También se puede extender a múltiples condiciones con else if.
4. Switch: *switch(condición){ case comparación declaración1 case comparación declaración2 default: declaración _default}*.
5. While: es un ciclo con una condición de salida al inicio del ciclo. Con la sintaxis: *while (condición) declaración*.
6. Do-While: es un ciclo con una condición de salida al final del ciclo. Con la sintaxis: *do declaración while (condición);*
7. For: es un ciclo con tres partes, una inicialización, otra comparación y una tercera de modificación de variable. Con la sintaxis: *for(inicialización; condición; cambio) declaración*
8. Goto: salto no condicional que salta hacia una etiqueta.
9. Break: finaliza cualquier ciclo y el switch
10. Continue: Finaliza una iteración para continuar en la siguiente.
11. Return: finaliza la ejecución de una función, se puede agregar un valor de retorno.

Funciones

La declaración de una función incluye su nombre, la lista de parámetros y un valor de retorno. La llamada a una función se puede hacer dentro de una declaración escribiendo el nombre de la función y los parámetros necesarios. También se define la función main como obligatoria, no teniendo ningún parámetro que definir, retornando un valor entero. Es permitido hacer funciones recursivas, y no se permiten hacer funciones anidadas

Flujo del programa

1. Se inicia el IDE.
2. Se ingresa el código fuente en el componente de texto o se abre un documento desde la pestaña de Archivo> Abrir.
3. Se ejecuta el código mediante: Traducir y Ejecutar. Si el código presenta errores, ver paso cuatro si no ir paso cinco.
4. Los errores pueden visualizarse en la sección inferior de errores o mediante ejecutar > Reporte errores. Se deben de arreglar dichos errores, sin embargo, el editor intentará recuperarse de los mismos.
5. Si el código se ejecuta correctamente, se abrirá el ejecutor de Augus en el cual saldrá el resultado de la ejecución. Se puede visualizar datos relevantes mediante los reportes de Tabla de Símbolos, gramaticales, errores, AST que se encuentran en el menú de ejecutar.