



Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ciencias y Sistemas  
Organización de Lenguajes y Compiladores 1

# MANUAL DE USUARIO

Henry Ronely Mendoza Aguilar  
Carnet: 202004810  
PROYECTO 2

## **Acerca de la Aplicación “Compscript”**

La aplicación tiene como principal objetivo el uso de un lenguaje de programación llamado “Compscript” diseñado para desarrollar las primeras practicas para adentrarse en el mundo de la programación y el desarrollo de la lógica para resolver problemas más complejos.

Las principales funciones para la aplicación son mediante el despliegue en la nube basado en la arquitectura cliente-servidor por lo que no le será necesario al usuario instalar programas externos para el funcionamiento correcto de la aplicación.

El uso del software es muy fácil de controlar de parte del usuario.

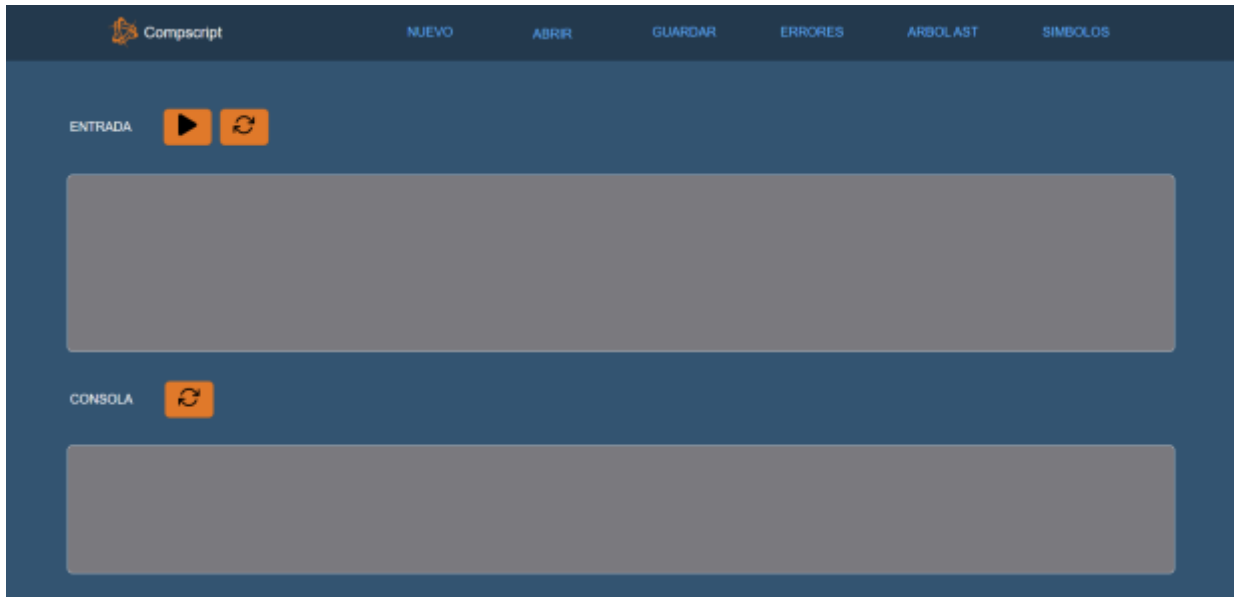
### **Requisitos del sistema**

- Tener Instalado un Navegador Web Chrome o Firefox
- Windows 7 o superior, Linux y MacOS

<b>CPU</b>	<b>Pentium 2 a 266 MHz</b>
<b>RAM</b>	512 MB
<b>Conexión a internet</b>	Debido al despliegue en la nube

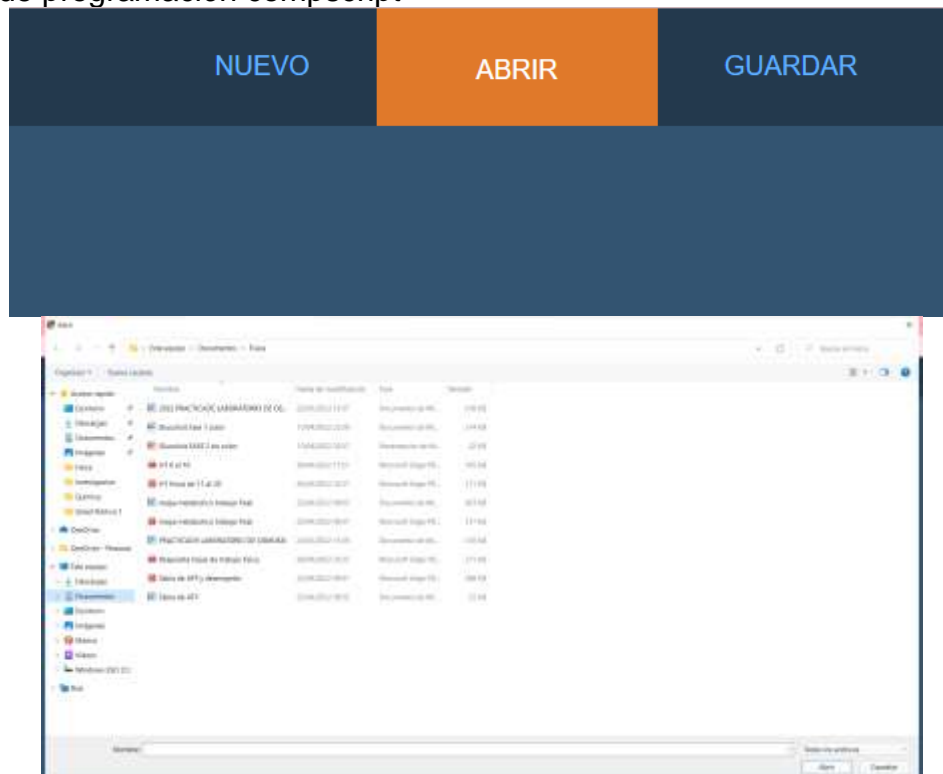
## Pantalla Principal

La ventana principal se compone de una barra de navegación la cual contiene las opciones necesarias para la ejecución del programa y visualización de reportes, Posteriormente cuenta con un área de texto en la cual se admitirá el lenguaje establecido un botón de ejecutar para realizar la compilación del código y un botón de recarga. En la parte inferior se visualiza un cuadro de texto el cual funciona como salida al programa compilado.



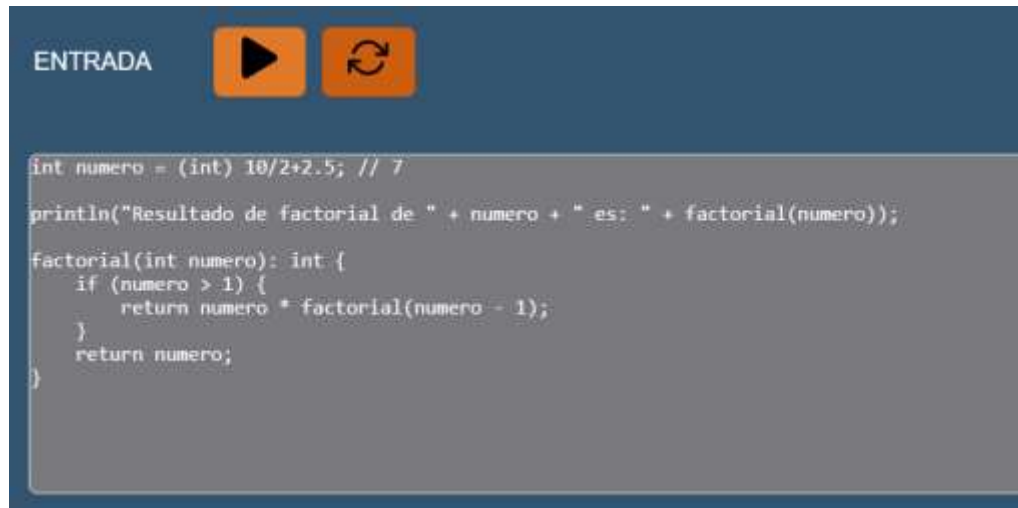
## Abrir

Al seleccionar esta opción de la barra de opciones se abrirá una ventana para escoger un archivo con extensión cst el cual es la extensión de archivos que emplea el lenguaje de programación compscript



## Ver Archivo

Para visualizar el contenido del archivo se debe de presionar el botón con símbolo de recarga de esta forma se mostrará el contenido en la consola de entrada



The screenshot shows the Compscript IDE interface. At the top, there is a dark blue header with the word 'ENTRADA' on the left and two orange buttons: a play button and a refresh button. Below the header is a code editor with a light gray background. The code in the editor is as follows:

```
int numero = (int) 10/2+2.5; // 7
println("Resultado de factorial de " + numero + " es: " + factorial(numero));
factorial(int numero): int {
    if (numero > 1) {
        return numero * factorial(numero - 1);
    }
    return numero;
}
```

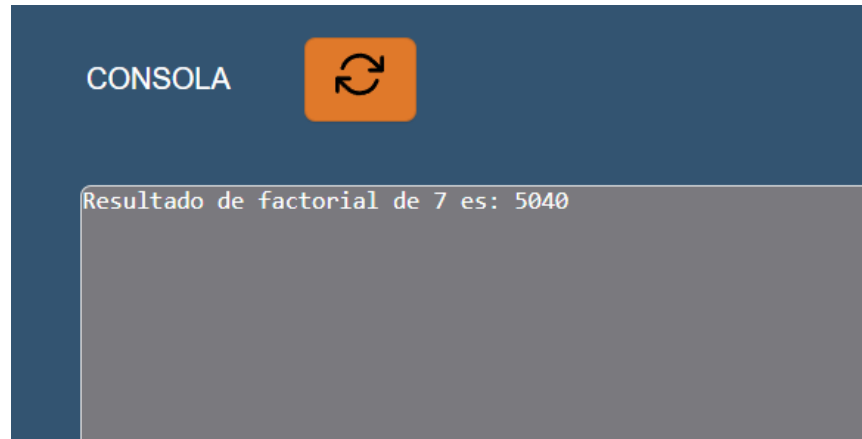
## Ejecución

Para la compilación del programa se debe presionar el botón con el símbolo play y mostrando un mensaje de confirmación.



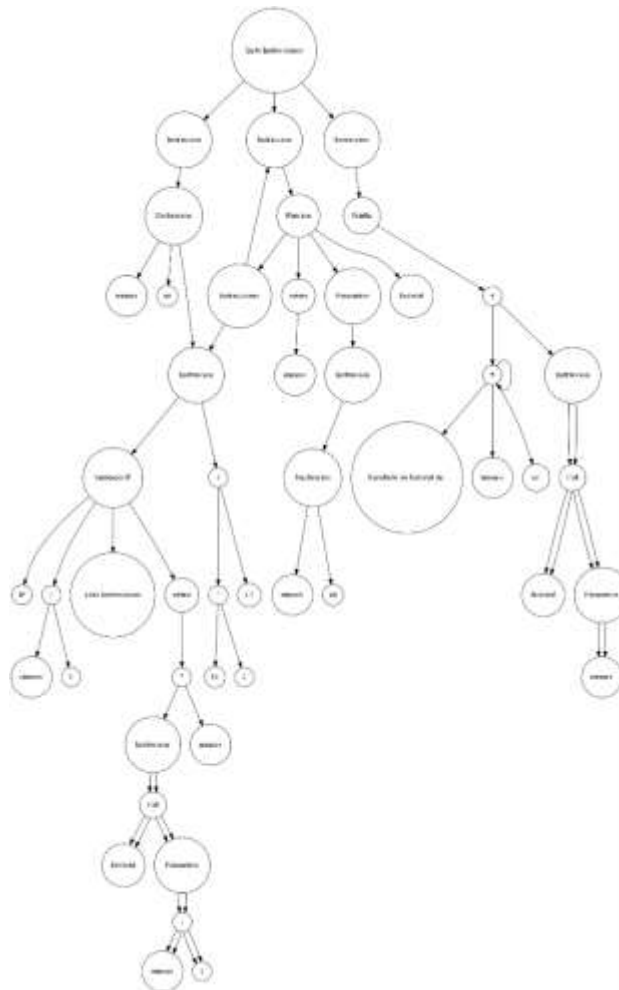
## Mostrar Salida

Para mostrar la salida en consola se debe de presionar el botón con símbolo de recarga en el área de salida, seguidamente se mostrará el contenido en el área de texto de salida tome en cuenta que este contenido no se puede modificar.



## Mostrar Arbol AST

En la barra de navegación en la opción de Arbol AST si se presiona esta opción se mostrará una imagen del árbol generado a partir del documento compilado.



## Tabla de símbolos

En la barra de navegación se debe de presionar en la opción de Símbolos para ejecutar el reporte de los símbolos almacenados durante la ejecución.

Tabla de Símbolos			
Identificador	Tipo	Tipo	Valor
factorial	Funcion/Metodo	int	None
numero	Variable	int	1

## Reporte de errores

Para visualizar el reporte de errores se debe seleccionar la opción de errores en la barra de navegación con esto se desplegará una ventana con la información si se obtienen errores durante la ejecución. Pueden existir errores de tipo léxico, sintáctico y semántico.

Reporte de Errores			
Caracter	Tipo de Error	Fila	Columna
*	Lexico	3	0
factorial	Sintactico	6	0
Tipo de dato incompatible, no se puede asignar el valor a prueba		Semantico	3

## Descripción del Lenguaje

### Tipos de datos admitidos

<b>Int</b>	Guarda números enteros
<b>Double</b>	Guarda números decimales
<b>String</b>	Guarda cadenas de palabras
<b>Char</b>	Guarda un único carácter
<b>Boolean</b>	Obtiene valor de true o false

### Declaración y Asignación

Declaración	Sintaxis
Variable	Tipo identificador = Valor
Vector	Tipo identificador [] = new Tipo [tamaño]
Matriz	Tipo identificador [][] = new Tipo [tamaño][tamañp]
Vector	Tipo identificador [] = [Lista de Valores]
Matriz	Tipo identificador [][] = [[Lista],[Lista],...]

Asignacion	Sintaxis
Variable	identificador = Valor
Vector	Tipo identificador [] = new Tipo [tamaño]
Matriz	Tipo identificador [][] = new Tipo [tamaño][tamañp]

### Ciclos

Ciclo	Sintaxis
For	For(declaracion; validación; incremento){Lista Instrucciones}
While	While(validación){Lista instrucciones}
Do while	Do{Lista instrucciones}while(validación)

### Sentencias

Sentencia	Sintaxis
if	If(validación){Lista Instrucciones}else if(validación){Lista Instrucciones}else{Lista instrucciones}
switch	Switch(valor){ Case valor: lista instrucciones ... Lista Case break; default Lista instrucciones}
función	Identificador(parámetros):tipo{Lista de instrucciones return;}
Método	Identificador(parámetros):void{Lista de instrucciones}
run	Run identificador (parámetros);