

Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Auxiliar: Maynor Pilo Tuy

Organización de Lenguajes y Compiladores 1

Proyecto 2 TypeWise

Lesther Kevin Federico López Miculax

Carnet: 202110897

03/05/2023

DATOS DE DESARROLLADOR

Nombre del desarrollador: Lesther Kevin Federico López Miculax

Correo electrónico: lestherlopez64@gmail.com

Nacionalidad: guatemalteca

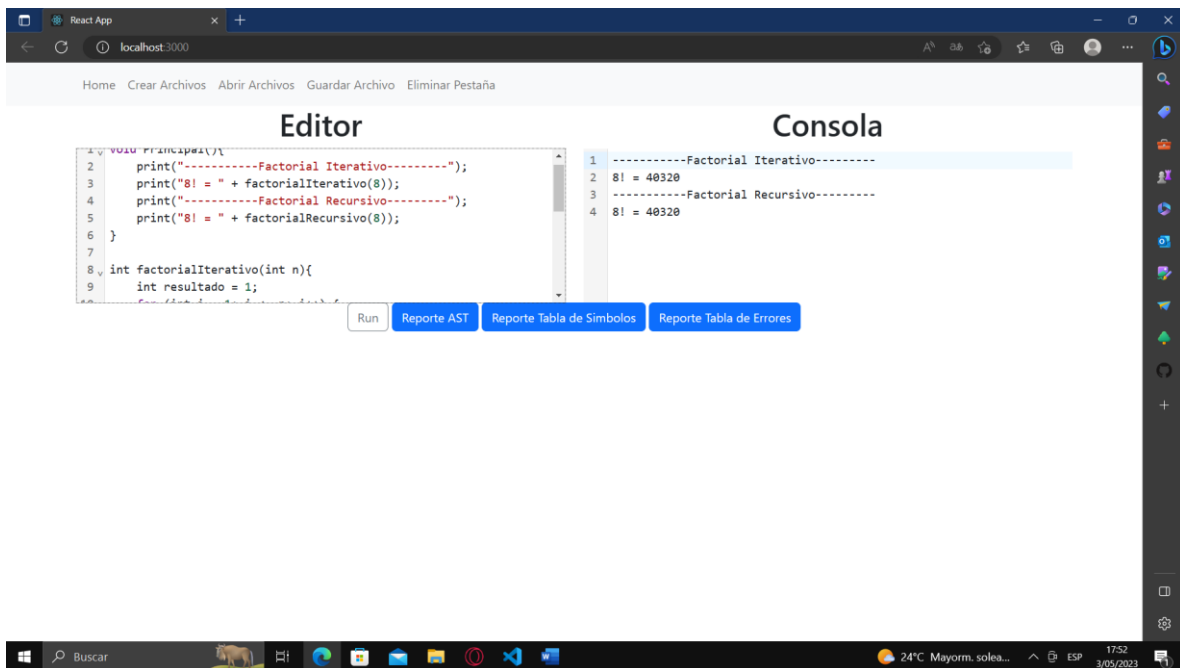
Carné universitario: 202110897

DESCRIPCION GENERAL

El curso de Organización de Lenguajes y Compiladores 1 de la Universidad de San Carlos de Guatemala es un curso importante para los estudiantes de informática, ya que les enseña cómo funcionan los lenguajes de programación y cómo se pueden crear compiladores e intérpretes para procesarlos. Por lo que, el programa o proyecto denominado EXREGAN, que fue presentado anteriormente, ha sido bien recibido por el curso de manera aprobatoria. Este programa ayuda en la generación de expresiones regulares, lo cual es fundamental en el proceso de análisis léxico de los lenguajes de programación. Debido al éxito de EXREGAN, el curso ahora está interesado en desarrollar un nuevo lenguaje llamado TypeWise. Este lenguaje será un intérprete que permitirá a los estudiantes de Introducción a la Programación y Computación 1 realizar sus primeras prácticas de programación. Se espera que TypeWise sea un lenguaje sencillo, fácil de aprender y utilizar, pero a la vez poderoso para que los estudiantes puedan experimentar y construir programas útiles desde el principio. De este modo, TypeWise tiene diversas especificaciones para el usuario las cuales se especificarán a continuación de manera detallada con el propósito de que la experiencia del usuario sea la mejor, logrando el objetivo de typewise.

MENU PRINCIPAL

El menú o interfaz grafica de typewise es altamente interactiva y accesible, por lo que el usuario no debería tener mayores complicaciones para lograr utilizarla. Existen varios componentes los cuales cada uno tiene una funcionalidad que permite que el usuario obtenga las funciones que desea.



EDITOR

Una parte fundamental de TypeWise es el editor, el cual, al abrir un archivo en el programa, su contenido se presentará en una caja de texto que se encuentra en la interfaz gráfica. Esta caja de texto es una sección crucial que se caracteriza por ser una zona rectangular en la que los usuarios pueden escribir y hacer los cambios necesarios en los archivos antes de guardarlos. Es importante destacar que esta caja de texto incluye una barra de desplazamiento, que se puede utilizar para moverse hacia arriba o hacia abajo en el

contenido. No obstante, es importante señalar que esta opción solo es funcional si el contenido de la caja de texto es demasiado extenso.

Home | Crear Archivos | Abrir Archivos | Guardar Archivo | Eliminar Pestaña

Editor

```
1 void Principal(){
2     print("-----Factorial Iterativo-----");
3     print("8! = " + factorialIterativo(8));
4     print("-----Factorial Recursivo-----");
5     print("8! = " + factorialRecursivo(8));
6 }
7
8 int factorialIterativo(int n){
9     int resultado = 1;
```

CONSOLA

El proyecto cuenta con una consola que es una parte esencial de la interfaz gráfica. Esta consola permite ver la salida obtenida después de evaluar una cadena, lo que representa el resultado final. Además, si hay algún error durante el análisis, se mostrará un mensaje de error en esta consola. Por lo tanto, resulta fundamental para verificar el proceso de evaluación y detectar posibles errores.

Consola

```
1 -----Factorial Iterativo-----
2 8! = 40320
3 -----Factorial Recursivo-----
4 8! = 40320
```

BARRA SUPERIOR

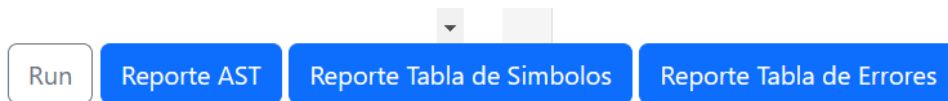
La interfaz tiene dentro de sus propiedades una barra superior que tiene diversos botones que cada uno de ellos tiene diferentes funcionalidades que cumplen con sus objetivos de facilitar el desempeño del estudiante o usuario que haga uso de typewise. Por lo que, se especificara a continuación que hace cada uno de estos botones y que conlleva.

Crear Archivos Abrir Archivos Guardar Archivo Eliminar Pestaña

- Crear archivo puede abrir una ventana donde el usuario puede ingresar un nombre para el archivo y seleccionar una ubicación para guardar. Luego haga clic en el botón Guardar y se creará un nuevo archivo con lo que se escribió en su editor de texto.
- Al hacer clic en el botón Abrir archivo, se abre un cuadro de diálogo que muestra una lista de archivos disponibles en el sistema de archivos. Los usuarios pueden buscar directorios y seleccionar archivos para abrir. Una vez que se selecciona el archivo, el usuario puede hacer clic en el botón Abrir para cargar el archivo en la aplicación. Por consiguiente, el texto del archivo se mostrará en el editor de la página.
- Al presionar el botón de guardar archivo, lo que sucederá es que los cambios realizados dentro del editor se guardaran dentro del archivo abierto.

BOTONES

Finalmente, dentro de los componentes de la interfaz se encuentran 4 botones que tiene cada uno al igual que los de la barra superior una funcionalidad específica, los cuales se especificaran adelante.



- En primera instancia, el botón run es el botón con mayor importancia, debido a que al seleccionar este botón la aplicación tomara el texto que esta en nuestro editor y lo analiza pasando por todas las funciones de la aplicación de manera que retorna en la consola el resultado final de lo enviado.
- Por otro lado, el botón de reporte ast lo que hace es abrir un modal que muestra dentro de el como se visualiza el AST del texto ingresado, de esta manera, el usuario tendrá a disposición y a la mano cómo funciona el ast de su expresión.
- El tercer botón corresponde a la interpretación de los símbolos que contiene el texto de entrada en una tabla, por lo que al presionar este botón se mostrara en un modal la tabla de símbolos con información adecuada y específica.
- Finalmente, si existiera algún tipo de inconveniente que evite el análisis léxico o sintáctico del archivo de entrada, se realizara la gráfica de una tabla que se presentará en un modal, donde vendrá la información necesaria para conocer de donde proviene el error, ya sea sintáctico o léxico, por lo que esta información es: No. Error, tipo de

error, descripción, línea y columna del error, de este modo, resulta efectivo este tipo de reporte para corregir el error con tiempo.

Identificador	Tipo	Tipo	Entorno	Linea	Columna
Principal	Método	VOID	global	1	0
factorialIterativo	Función	INT	global	8	0
factorialRecursivo	Función	INT	global	16	0
resultado	variable	INT	Funcion factorialIterativo	9	4
i	variable	INT	Funcion factorialIterativo	10	9

#	Tipo de error	Descripción	Linea	Columna
---	---------------	-------------	-------	---------

ARCHIVO DE ENTRADA

Por último, una especificación importante es el archivo de entrada que acepta y es capaz de reconocer TW, el cual es un archivo con extensión .tw como las iniciales del nombre del proyecto, por lo que este archivo también tiene una estructura y reglas como la declaración de variables, funciones, método, entre otros. En la siguiente imagen se da un ejemplo de como es la estructura de este tipo de archivos.

seis / 50248 / AppData / Local / temp / temp5_Archivos de prueba-20230424.zip / recursividad.t

```
void Principal(){
    print("-----Factorial Iterativo-----");
    print("8! = " + factorialIterativo(8));
    print("-----Factorial Recursivo-----");
    print("8! = " + factorialRecursivo(8));
}

int factorialIterativo(int n){
    int resultado = 1;
    for (int i = 1; i <= n; i++) {
        resultado = resultado * i;
    }
    return resultado;
}

int factorialRecursivo(int n) {
    if (n == 0) {
        return 1;
    }
    return (n * factorialRecursivo(n - 1));
}
```