

Escuela de Ciencias y Sistemas

Organización de Lenguajes y Compiladores 1

Proyecto 2
Manual de Usuario

César André Ramírez Dávila

202010816

Fecha: 04/11/2022

Índice

<i>Introduccion</i>	<i>3</i>
<i>Objetivos</i>	<i>3</i>
<i>Requerimientos</i>	<i>4</i>
<i>Uso del programa.....</i>	<i>5</i>
1. Archivo de Entrada	5
2. Interfaz	6
2.1. Cargar Archivo	6
2.2. Árbol de sintaxis Abstracta	7
2.3. Reporte de Errores	7
3. Guardar Archivo	8

Introduccion

El presente documento describe las instrucciones para el uso correcto del programa del programa. El documento familiariza al usuario que hace uso del programa, sin tener complicaciones a la hora de ejecutarlo.

Objetivos

- Instruir el uso adecuado del Sistema de información, para el acceso adecuado en el uso de este, mostrando los pasos de desarrollo del programa, así como la descripción de las funciones y métodos que puede en la realizacion del programa.
- Comprender uso de un analizador léxico y sintáctico.
- Obtener mayor conocimiento en el lenguaje de programación angular, json y typescript

El presente manual está enfocado para personas que conocen código similar a java o personas que quieren aprender programación.

Requerimientos

- La aplicación puede ser ejecutada en cualquier sistema operativo que tenga instalado node.js en el sistema.
- IDE recomendado: Visual studio code
- Equipo Intel Pentium o superior
- Espacio en el disco duro, al menos 500 mb
- Memoria ram recomendada 4gb (por el servidor)

Uso del programa

1. Archivo de Entrada

```
1  /**
2
3      Organizacion de lenguajes y compiladores 1
4      Segundo semestre 2022
5      Universidad San Carlos de Guatemala
6
7  ****/
8
9  //Seccion de declaraciones
10
11  int variable_1;
12  double variable_2;
13  boolean variable_3;
14  char variable_4;
15  string variable_5;
16  int variable_1, variable_11;
17  double variable_2, variable_22;
18  boolean variable_3, variable_33;
19  char variable_4, variable_44;
20  string variable_5, variable_55;
21  int var_1 = 12;
22  int tm1_,tmp2_,tmp3_,tmp4_ = 12+5;
23  double var_2 = 15.02;
24  double var_2_1 = 2022.00;
25  double var_2_2 = 0.10;
26  double var_2_3 = 1.0;
27  double var_2_4 = 51.1589462625459;
28  boolean var_3 = true;
29  boolean var_3_1 = false;
30  char var_4_1 = 'a';
31  char var_4_2 = '2';
32  char var_4_3 = 'W';
33  char var_4_4 = '@';
34  char var_4_5 = '!';
35  char var_4_6 = '%';
36  char var_4_7 = '/';
37  string var_5_1 = "Hola";
38  string var_5_2 = "Mundo";
39  string var_5_3 = "este es un mensaje@";
40  string var_5_4 = "Organizacion de lenguajes y compiladores
41  1";
```

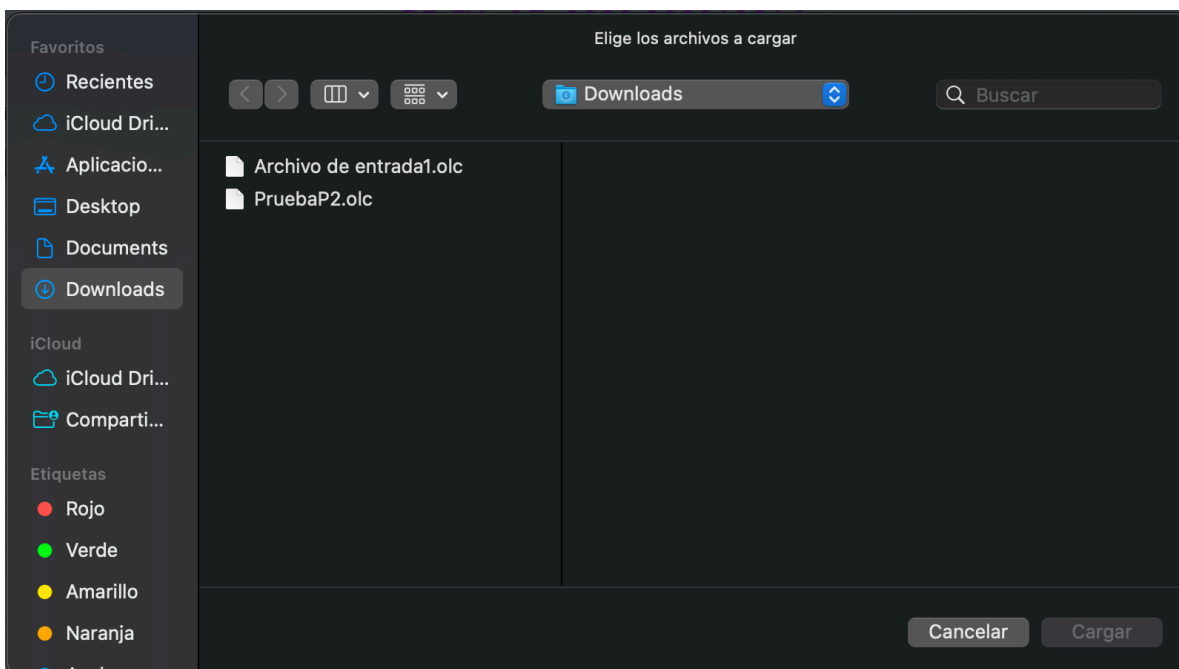
La extensión del archivo debe ser .OLC y debe contener una estructura de código tipo java similar a la mostrada.

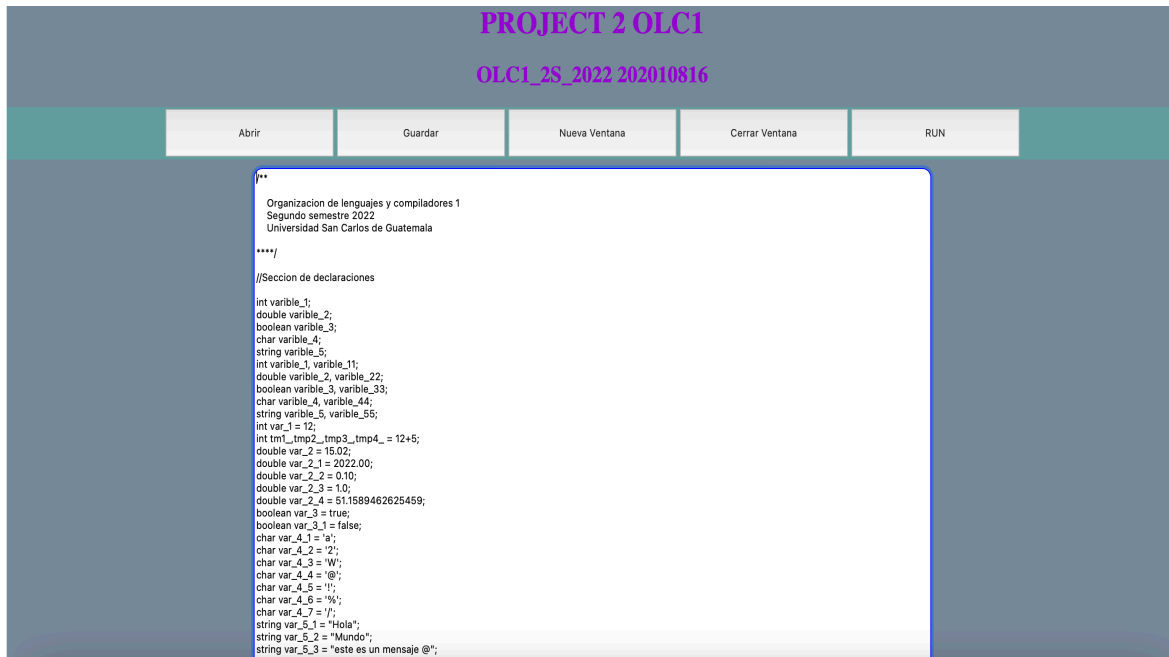
2. Interfaz



Al iniciar el programa tenemos la siguiente interfaz, en la cual debemos cargar un archivo.

2.1. Cargar Archivo





Nos carga el contenido del Archivo donde podemos modificar o crear el Árbol de Sintaxis Abstracta, reporte de tokens y reporte errores.

2.2. Árbol de sintaxis Abstracta

Al presionar el botón RUN se compila la entrada y se obtiene el árbol



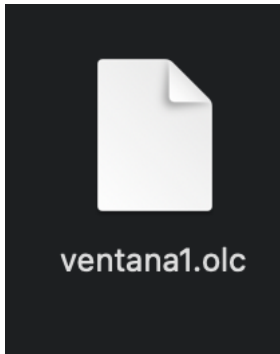
2.3. Reporte de Errores

Obtenemos el reporte de errores

Tipo error	Descripcion	Línea	Columna
Lexico	No se reconoce el caracter #	2	1
Lexico	No se reconoce el caracter #	2	2
Lexico	No se reconoce el caracter @	3	1
Lexico	No se reconoce el caracter @	3	2

3. Guardar Archivo

Este botón almacena automáticamente el contenido del editor de código con un nombre definido.



Nos permite guardar el archivo por si este es editado en la caja de texto.

```
1  /**
2
3      Organizacion de lenguajes y compiladores 1
4      Segundo semestre 2022
5      Universidad San Carlos de Guatemala
6
7  ****/
8  @@
9  //Seccion de declaraciones
10 ###
11 int variable_1;
12 double variable_2;
13 boolean variable_3;
14 char variable_4;
15 string variable_5;
16 int variable_1, variable_11;
17 double variable_2, variable_22;
18 boolean variable_3, variable_33;
19 char variable_4, variable_44;
20 @@
21
22 guardando desde frontend
```