

Escuela de Ciencias y Sistemas

Organización de Lenguajes y Compiladores 1

Proyecto 1  
Manual de Usuario

César André Ramírez Dávila

202010816

Fecha: 19/09/2022

## Tabla de contenido

<b>Introduccion .....</b>	<b>3</b>
<b>Objetivos .....</b>	<b>3</b>
<b>Requerimientos .....</b>	<b>4</b>
<b>Uso del programa.....</b>	<b>5</b>
<b>1. Archivo de Entrada .....</b>	<b>5</b>
<b>2. Interfaz .....</b>	<b>5</b>
2.1. Cargar Archivo .....	6
2.2. Arbol Sintactico .....	6
<b>3. Traducciones .....</b>	<b>7</b>
3.1. Traducción a Python.....	8
<b>4. Guardar Archivo.....</b>	<b>9</b>

## 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 del pseudocódigo
- Obtener mayor conocimiento en el lenguaje de programación python y golang

El presente manual está enfocado para personas que conocen pseudocódigo o personas que quieren aprender programación.

## Requerimientos

- La aplicación puede ser ejecutada en cualquier sistema operativo que tenga instalado Java en el sistema.
- IDE recomendado: Netbeans o visual studio code (opcional)
- Equipo Intel Pentium o superior
- Espacio en el disco duro, al menos 500 mb
- Memoria ram recomendada 2gb

# Uso del programa

## 1. Archivo de Entrada

```
inicio
/////inicio de la traduccion

/*
    Primer archivo de entrada para proyecto organizacion de lenguajes y compiladores 1
*/

//seccion de declraciones de funciones de variables
Ingresar _variable1_      coMo NumEro c0n_valor 5;
ingresaR _v1_, _v2_, _v3_ c0mo Cadena con_vAlor "esta es una cadena";
ingresaR _curso1_         c0mo cadena con_vAlor "olc";
ingresaR _curso2_         c0mo cadena con_vAlor "olc";
ingresaR _curso3_         c0mo cadena con_vAlor "olc";
ingresar _pi1_            como numero con_valor 3;

ingresar _operaciones1Basica_      como numero con_valor 1+(1);           //2 el resultado
ingresar _operaciones1Basica2_     como numero con_valor _operaciones1Basica+_operaciones1Basica_;           //
ingresar _operaciones1Intermedia_ Como nUmero con_valor 15+(9*8)+200/8*3+9; //171 el resultado
ingresar _operaciones1Avanzadas1_ Com0 numEro con_valor ((15+9)*8+200/8*3+9); //291 el resultado

ingresar _operacionRela1_ como Boolean con_valor 5+5 mayor 5;
ingresar _operacionRela3_ como boolean con_valor _operaciones1Basica_ mayor 8;

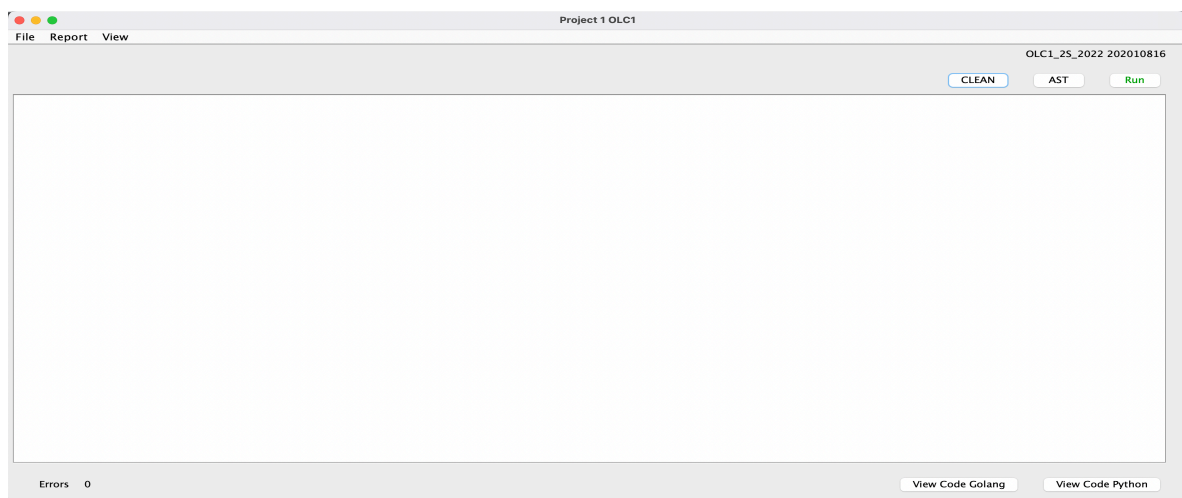
//seccion de asignaciones
_v1_      -> "esta es la cadena numero 1";
_v2_, _v3_ -> "estas cadenas deben ser diferentes";
_curso1_ , _curso2_ , _curso3_ -> "Organizacion de lenguajes y compiladores 1";

imprimir_nl _encabezado1_;
imprimir_nl _encabezado2_;
imprimir "...";
imprimir _anio1_ ;
imprimir _anio2_ ;
imprimir _anio3_ ;
imprimir _anio4_ ;
imprimir_nl ".";
imprimir_nl (_v3_);

fin
```

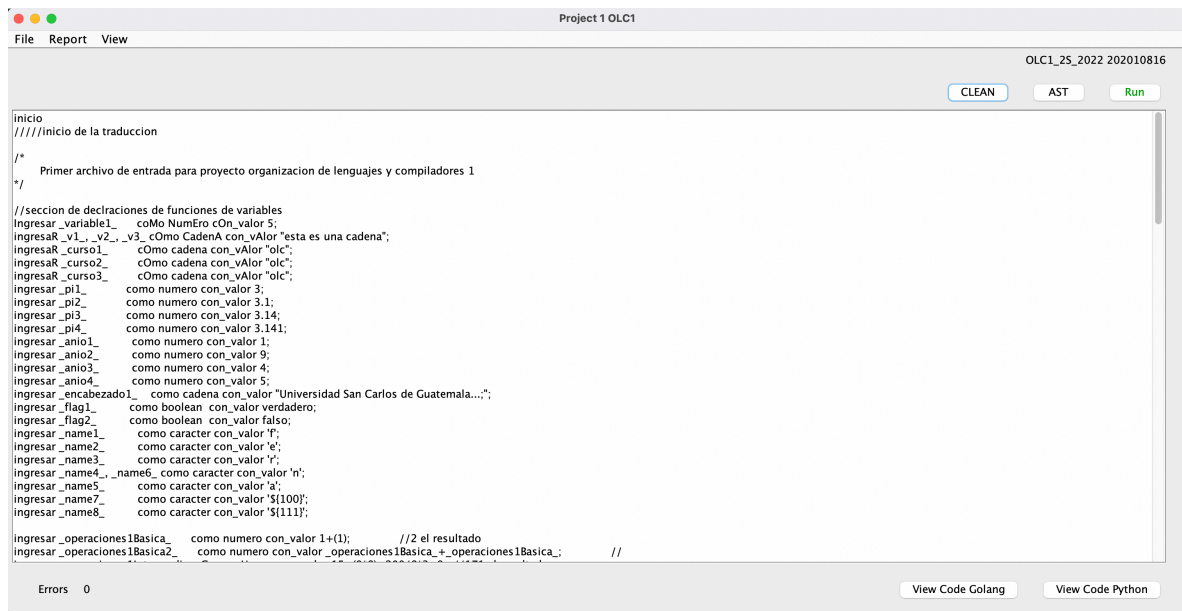
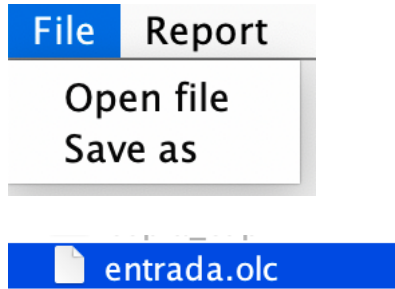
La extensión del archivo debe ser .OLC y debe contener la palabra INICIO luego una estructura de pseudocódigo similar a la mostrada y por último la palabra FIN.

## 2. Interfaz



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

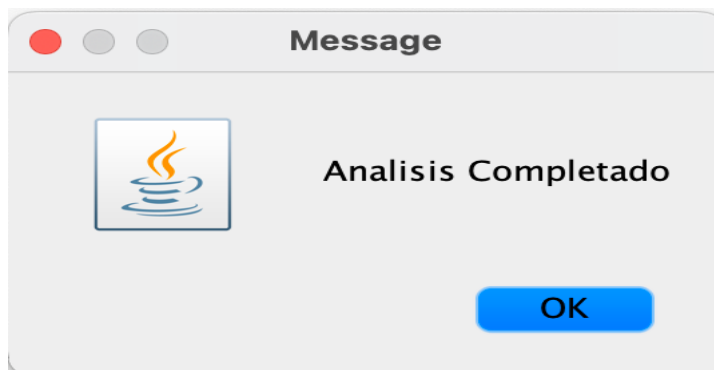
## 2.1. Cargar Archivo

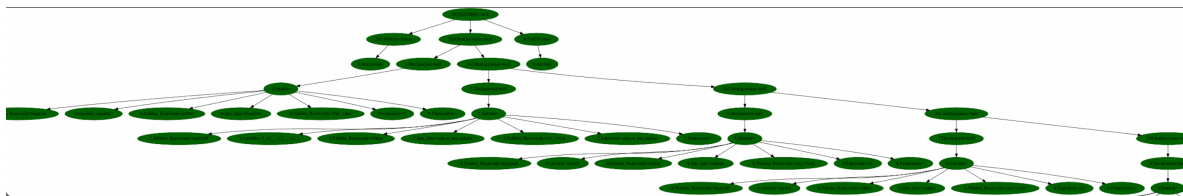


Nos carga el contenido del Archivo donde podemos modificar o crear el Arbol Sintactico o hacer la traducción a otros lenguajes.

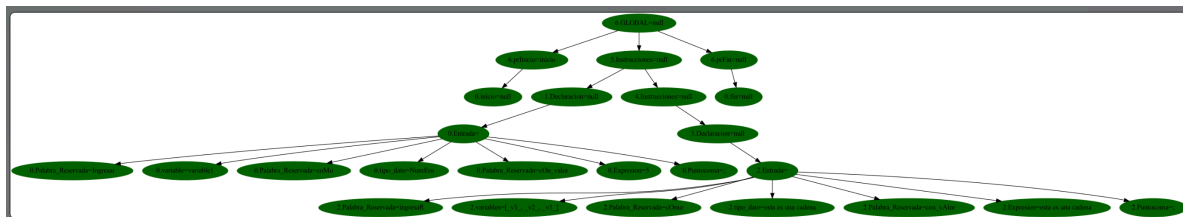
## 2.2. Arbol Sintactico

Al presionar el botón AST nos genera el siguiente mensaje si no hay errores





El AST tiene la siguiente estructura con muchas instrucciones



Este es un archivo con pocas instrucciones

### 3. Traducciones

#### REPORTE ERRORES

Num	Lexema	Fila	Columna	Descripcion	Tipo
0	@	8	2	NO PERTENECE AL LENGUAJE	LEXICO
1	\$	8	3	NO PERTENECE AL LENGUAJE	LEXICO
2	&	8	4	NO PERTENECE AL LENGUAJE	LEXICO
3	c	59	35	NO PERTENECE AL LENGUAJE	LEXICO
4	@	61	1	NO PERTENECE AL LENGUAJE	LEXICO
5	\$	70	1	NO PERTENECE AL LENGUAJE	LEXICO
6	r	111	31	NO PERTENECE AL LENGUAJE	LEXICO
7	ingresaR	13	1	Caracter no esperado	SINTACTICO
8	)	46	84	Caracter no esperado	SINTACTICO
9	_v1_	64	1	Caracter no esperado	SINTACTICO
10	"Organizacion de lenguajes y compiladores 1"	67	32	Caracter no esperado	SINTACTICO
11	1	139	16	Caracter no esperado	SINTACTICO
12	3	141	28	Caracter no esperado	SINTACTICO
13	3	142	38	Caracter no esperado	SINTACTICO

Si el programa cuenta con errores nos abre automáticamente el archivo e indicando fila, columna y tipo de error encontrado.

### 3.1. Traducción a Python

Al presionar el botón de View Code in Python nos genera el archivo de Salida con las instrucciones analizadas previamente.

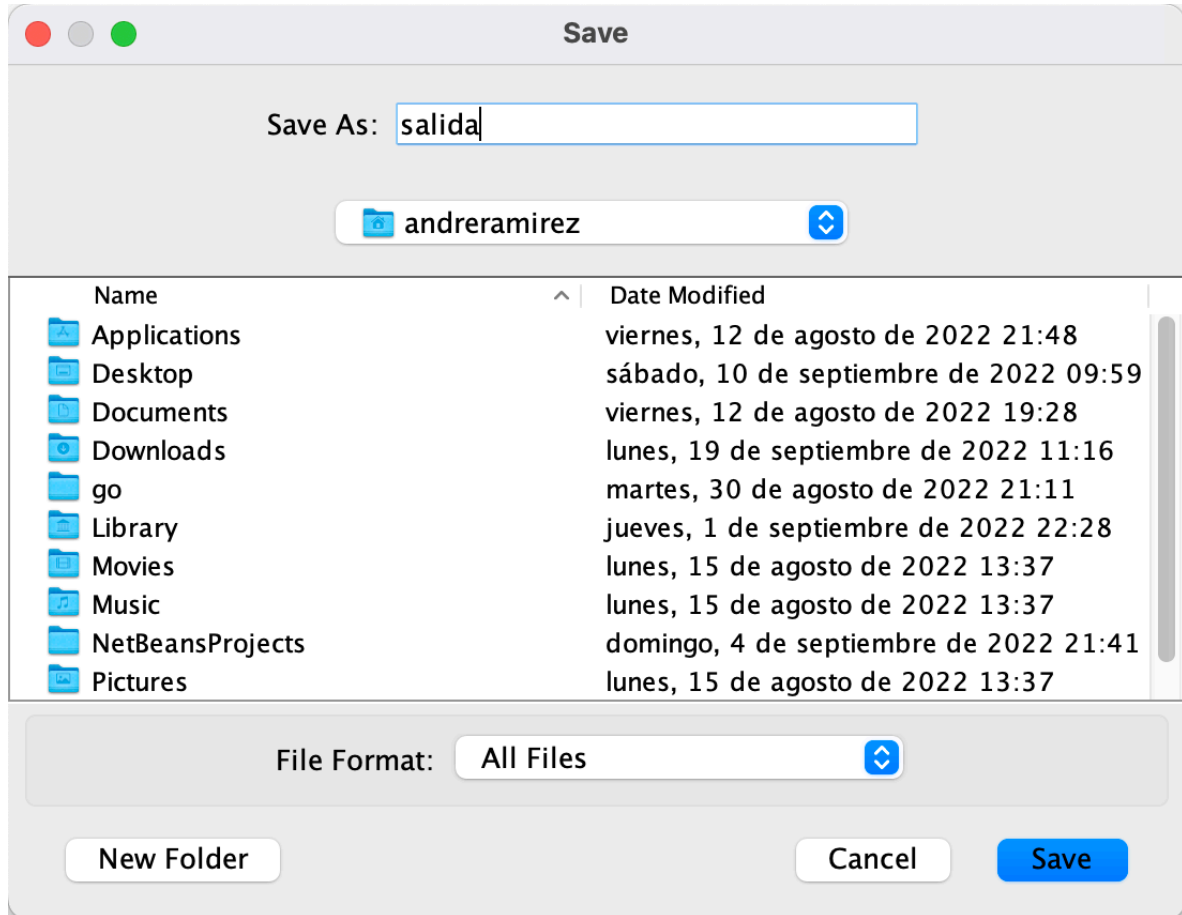
```
SalidaTraducidaPY.py

def main():
    variable1 = 5
    v3 = "esta es una cadena"
    v2 = "esta es una cadena"
    v1 = "esta es una cadena"
    curso1 = "olc"
    curso2 = "olc"
    curso3 = "olc"
    pi1 = 3
    pi2 = 3.1
    pi3 = 3.14
    pi4 = 3.141
    anio1 = 1
    anio2 = 9
    anio3 = 4
    anio4 = 5
    encabezado1 = "Universidad San Carlos de Guatemala...;"
    encabezado2 = "Escuela de Ciencias y Sistemas"
    Segundo semestre
    "
    flag1 = True
    flag2 = False
    name1 = 'f'
    name2 = 'e'
    name3 = 'r'
    name6 = 'n'
    name4 = 'n'
    name5 = 'a'
    name7 = 'd'
    name8 = 'o'
    operaciones1Basica = 1 + (1)
    operaciones1Basica2 = _operaciones1Basica_ + _operaciones1Basica_
    operaciones1Intermedia = 15 + (9 * 8) + 200 / 8 * 3 + 9
    operaciones1Avanzadas1 = ((15 + 9) * 8 + 200 / 8 * 3 + 9)
    operaciones1Avanzadas2 = 30**[22.2 - 2.2] + (2)
    operaciones1Avanzadas3 = (30**[2]) + (2)
    operaciones1Avanzadas4 = (30**[10 - 8 + 9 - 4 * 2 - 1]) + (2)
    operaciones1Avanzadas6 = (5 * 8)%(1 + 5 + 6)
    operacionRela1 = 5 + 5>5
    operacionRela3 = _operaciones1Basica_>8
    operacionRela3 = (_operaciones1Basica_ + 6)>=8
    operacionRela3 = (_operaciones1Basica_ + 6)<=8
    operacionRela4 = _operaciones1Basica_==8
    operacionRela5 = _operaciones1Basica_==_operaciones1Basica_
    operacionRela6 = _operaciones1Basica_==_operaciones1Basica_ + 1
    operacionRela7 = _operaciones1Basica_==( _operaciones1Basica_ ) * (8 + 5)
    operacionRela5 = _operaciones1Basica_!=_operaciones1Basica_
    v1 = "esta es la cadena numero 1"
    v3 = "estas cadenas deben ser diferentes"
    v2 = "estas cadenas deben ser diferentes"
    curso3 = "Organizacion de lenguajes y compiladores 1"
    curso2 = "Organizacion de lenguajes y compiladores 1"
    curso1 = "Organizacion de lenguajes y compiladores 1"
```

Igualmente con el lenguaje Go con su respectiva sintaxis.



#### 4. Guardar Archivo



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