

UNIVERSIDAD SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERIA
ESCUELA DE CIENCIAS Y SISTEMAS
ORGANIZACIÓN DE LENGUAJES Y COMPILADORES 1
SECCION "N"

MANUAL DE USUARIO

Juan Pablo Pérez Bú

201807201

Guatemala, 19 de septiembre del 2022

A CERCA DEL TRADUCTOR

Esta aplicación fue creada con la finalidad de poder convertir una entrada de texto en formato de pseudocódigo en una salida en dos lenguajes de programación diferentes; siendo así el caso de Go y Python. La aplicación también es capaz de detectar los errores léxico y sintácticos y generar un archivo (el mismo que el de entrada) con las advertencias impresas, indicando la información necesaria para la corrección de dichos errores.

Requerimientos

- **CPU:** Intel Core i3 o su equivalente
- **RAM:** 4GB (Recomendado)
- **Memoria disponible:** 50 MB
- **SO:** Windows 7 en adelante

Para ejecutar la aplicación haga doble click sobre el archivo:

“Proyecto1-1.0-SNAPSHOT.jar”

Ayuda adicional:

Para la comprensión de la gramática del pseudocódigo puede visualizar el archivo “gramática.pdf” ubicada en la carpeta “doc”.

Este traductor únicamente admite archivos de entrada con extensión “*.olc”.

Atajos de teclado en el editor de código:

- **CTRL+A:** abre un selector de archivos para abrir un documento
- **CTRL+S:** guarda el archivo que se está editando actualmente o si no hay algún archivo abierto entonces se guarda como un nuevo documento.
- **CTRL+R:** ejecuta el proceso de compilación del pseudocódigo.
- **CTRL+Z:** elimina el último cambio realizado en el documento.
- **CTRL+Y:** agrega el último cambio eliminado del documento.

Edición de Código:

```
2  /*CONDICIONAL SEGUN*/
3  ingresar _a_ como numero con_valor 4;
4  ingresar _b_ como numero con_valor 7;
5  si _a_ menor _b_ entonces
6      ingresar _c_ como numero con_valor 5;
7  o_si _b_ menor _a_ entonces
8      ingresar _c_ como numero con_valor 6;
9  o_si _b_ es_igual _a_ entonces
10     ingresar _c_ como numero con_valor 7;
11 de_lo_contrario
12     ingresar _c_ como numero con_valor 8;
13 fin_si
14 /*CONDICIONAL SEGUN*/
15 ingresar _s_ como numero con_valor 2;
16 segun _s_ hacer
17     { 1 ? entonces
18         ingresar _f_ como caracter con_valor '${50}';
19     { 2 ? entonces
20         ingresar _f_ como caracter con_valor '${51}';
21     { 3 ? entonces
22         ingresar _f_ como caracter con_valor '${52}';
23     de_lo_contrario entonces
24         ingresar _f_ como caracter con_valor '${53}';
25 fin_según
26 para _i_ -> 0 hasta 5 con incremental 2 hacer
27     _cont_ -> _cont_+1;
28 fin_para
29 fin
```

En este editor de texto es donde se ingresará el pseudocódigo que posteriormente será compilado

Terminal:

```
Linea: 14 Columna: 23
var _a_ = 4
var _b_ = 7
var _c_ = 5
var _c_ = 6
var _c_ = 7
var _c_ = 8
CB/*CONDICIONAL SEGUN*/
LINE:14 COL:2
if _a_ < _b_ {
var _c_ = 5
```

En esta salida de texto se mostrarán los errores léxicos y sintácticos que puedan haber, así como una parte del proceso de traducción.

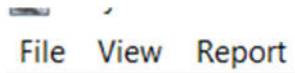
Barra de herramientas:



En este apartado se encuentran 5 opciones, en el orden de izquierda a derecha:

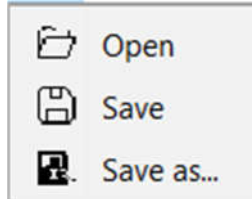
- **RUN:** realiza el proceso de compilación.
- **CLEAN:** limpia el editor de código.
- **PY:** Solo funcional luego de la compilación, genera un archivo con la salida de pseudocódigo traducida al lenguaje Python.
- **GO:** Misma función que PY, solo que este emite una salida de código en lenguaje Go.
- **Limpiar Terminal:** limpia el terminal.

Barra de menús:



Cada opción contiene un menú distinto:

- **File:** contiene las opciones relacionadas al manejo de archivos



- **View:** contiene las opciones de visualización de los manuales de usuario y técnico.



- **Report:** contiene las opciones de visualización de reportes de errores y diagramas de flujo.

