



AVANCES DE LA CONSTRUCCIÓN DEL TRADUCTOR

OMAR ALEJANDRO QUIROZ TRUJILLO

MAESTRO: MICHEL EMANUEL LOPEZ FRANCO

DECISIONES IMPORTANTES

En el transcurso de este proyecto tome la decisión de aprender mejor el lenguaje de programación **PYTHON** esto porque estuve manejando lenguajes de “C++, C#, Ensamblador (Emu y Verilog), etc” me gustaría aprender más y gracias a este proyecto me está comenzando a gustar a ser Full-stack ya que me gusta tanto ensamblador, back y front, pero es cuestión de seguir estudiando para mejorare mi lógica y conocer mejor los lenguajes de programación.

Hubo momentos difíciles donde no entendía y me preguntaba ¿Qué puedo hacer aquí? ¿Cómo puedo acabar este programa? Pero con ayuda de mis propios compañeros pude lograr a tener mejor noción del programa. Todavía no logro adaptarme para crear una interfaz, estoy estudiando más las librerías de **PYTHON** para poder usarlas de la mejor manera, me encanta realizar proyectos así ya que estos son más retos para mí, pero se trata de mejorar cada día

MANERAS EN COMO SE MANEJAN LOS ERRORES

LÉXICOS Y SINTÁCTICOS

Los errores léxicos, como errores de escritura o de estructura básica del código, son fácilmente identificables durante la escritura del código gracias a las señales visuales que ofrecen los editores de texto o IDEs. Por ejemplo, un error tipográfico como escribir mal el nombre de una variable será resaltado en rojo por el editor. Estos errores suelen ser corregidos de forma manual al revisar el código.

Por otro lado, los errores sintácticos son aquellos que ocurren cuando el código no sigue las reglas gramaticales del lenguaje de programación. Estos errores son detectados por el compilador durante la fase de compilación del programa. Si hay algún error sintáctico, el compilador muestra mensajes de error indicando qué parte del código no cumple con la sintaxis del lenguaje. La corrección de estos errores implica ajustar el código para que cumpla con las reglas de sintaxis del lenguaje.

La prevención de errores léxicos y sintácticos se logra mediante el uso de buenas prácticas de codificación, como seguir convenciones de codificación establecidas, escribir código de manera clara y legible, y realizar pruebas exhaustivas. Esto ayuda a minimizar la aparición de errores y facilita la identificación y corrección de problemas antes de que se conviertan en errores más graves.

AVANCES DEL PROYECTO

```
main.py > ...
1  import analizador
2  print ("-----")
3  texto = input("Ingresa el texto que quieres analizar: ")
4  print ("-----")
5  analizador = analizador.AnalizadorLexico()
6  analizador.analizador(texto)
7  analizador.getListTokens()
8  print ("-----")
9  print("Esta es la salida bro")
10 print ("-----")
11 analizador.imprimirSalida()
12 print ("-----")
13 print("La pila es la siguiente bro")
14 print ("-----")
15 analizador.imprimirSintactico()
16 print ("-----")
17
18 #OMAR ALEJANDRO QUIROZ TRUJILLO
```

```

analizador.py > _
1  from pila import Estado, NoTerminal, Pila, Terminal
2  from regla import Regla
3
4  #símbolos
5  ERROR = -1;
6  INICIAL = 55;
7  IDENTIFICADOR = 0;
8  ENTERO = 1;
9  REAL = 2;
10 CADENA = 3;
11 TIPO = 4;
12 OPSUMA = 5;
13 OPMUL = 6;
14 OPRELAC = 7;
15 OPOR = 8;
16 OPAND = 9;
17 OPMOD = 10;
18 OPIGUALDAD = 11;
19 PUNTOCOMA = 12;
20 COMA = 13;
21 ABREPARENTESIS = 14;
22 CIERRAPARENTESIS = 15;
23 ABRELLAVE = 16;
24 CIERRALLAVE = 17;
25 IGUAL = 18;
26 IF = 19;
27 WHILE = 20;
28 RETURN = 21;
29 ELSE = 22;
30 SIGMOPESOS = 23;
31
32 #se define el analizador léxico
33 class AnalizadorLexico:
34     tiposDatos = {"int", "float", "void"}
35     reservadas = {"if", "while", "return", "else"}
36

```

SALIDAS

```

PS C:\Users\Omar\OneDrive\documentos\Escritorio\Septimo Semestre\SEM TRADUCTORES II\GRAMATICA COMPILADOR> & C:\Python3\python.exe "C:\Users\Omar\OneDrive\documentos\Escritorio\Septimo Semestre\SEM TRADUCTORES II\GRAMATICA COMPILADOR/main.py"
-----
Ingresa el texto que quieras analizar: J0SE
J0SE es un identificador

$ es un signo de $
-----
Esta es la salida bre
-----
R2
R1
R0
-----
la pila es la siguiente bre
-----
$5
$5 $2
$5 programaci
-----

```

Ingresa el texto que quieres analizar: 12+45

12 es un identificador

+ operador suma

45 es un entero

\$ es un signo de \$

Esta es la salida bro

R2

R1

R0

La pila es la siguiente bro

\$5

\$5 \$2

\$5 programa1

```

-----
Ingresa el texto que quieres analizar: $0, $1, $2
-----
es un identificador

$ es un signo de $

0 es un entero

, es una coma

$ es un signo de $

1 es un entero

, es una coma

$ es un signo de $

2 es un entero

$ es un signo de $

-----
Esta es la salida bro
-----
R2
R1
R0
R0
R0
R0
-----
La pila es la siguiente bro
-----
$5

$5 $2

$5 programa1

```

FALLAS

Actualmente la falla que tengo es en la pila, algunas veces lanza el mismo resultado, pero en ocasiones si cambia, falta darle detalle en esas pequeñas cosas. Aún falta comentarlo mejor para así se vea más organizado, en la semana del 04 de marzo me daré a la tarea de hacerlo mejor