



Universidad de Guadalajara
Centro Universitario
De
Ciencias Exactas
E
Ingenierías
(CUCEI)

Materia:
SEMINARIO DE SOLUCION DE PROBLEMAS DE
TRADUCTORES DE LENGUAJES II

Nombre: Aceves López Jonathan

Código:
217481363

Reporte del Código - Analizador Léxico en Python con Tkinter

Instrucciones:

Genera un pequeño analizador léxico en Python que identifique identificadores y números reales contruidos de la siguiente manera:

Identificadores: letra(letra|digito)*

Números Reales: entero.entero+

Código:

```
# Importar la biblioteca re para trabajar con expresiones regulares
import re
# Importar la biblioteca tkinter para construir la interfaz gráfica
import tkinter as tk

# Función que realiza el análisis léxico de la expresión
def analizador_lexico(expresion):
    # Definir patrones para identificadores, números reales y operadores
    patron_identificador = r'[a-zA-Z][a-zA-Z0-9_]*'
    patron_numero_real = r'\d+\.\d+|\d+'
    patron_operador = r'[-+*/]'

    # Combinar patrones en una expresión regular total
    patron_total = f'({patron_identificador})|({patron_numero_real})|({patron_operador})|(\s)'

    # Encontrar todos los tokens en la expresión
    tokens = re.findall(patron_total, expresion)

    # Procesar los tokens y construir el resultado
    resultado = ""
    for token in tokens:
        identificador, numero_real, operador, otros = token
        if identificador:
            resultado += f'{identificador} tipo de dato: identificador\n'
        elif numero_real:
            resultado += f'{numero_real} tipo de dato: número real\n'
        elif operador:
            resultado += f'{operador} tipo de dato: operador\n'
        elif otros and otros.strip():
            resultado += f'{otros} tipo de dato: desconocido\n'

    return resultado

# Función que se ejecuta al hacer clic en el botón "Analizar"
def analizar_expresion():
    expresion = entrada.get() # Obtener la expresión del campo de entrada
    resultado = analizador_lexico(expresion) # Realizar el análisis léxico
    resultado_text.config(state=tk.NORMAL) # Permitir la edición del área de texto
    resultado_text.delete(1.0, tk.END) # Borrar el contenido actual del área de texto
    resultado_text.insert(tk.END, resultado) # Insertar el nuevo resultado
    resultado_text.config(state=tk.DISABLED) # Configurar el área de texto como de solo lectura

# Crear la ventana principal
ventana = tk.Tk()
ventana.title("Analizador Léxico") # Establecer el título de la ventana

# Crear elementos de la interfaz
etiqueta = tk.Label(ventana, text="Ingrese la expresión:")
etiqueta.pack(pady=10)

entrada = tk.Entry(ventana, width=30)
entrada.pack(pady=10)

boton_analizar = tk.Button(ventana, text="Analizar", command=analizar_expresion)
boton_analizar.pack(pady=10)

resultado_text = tk.Text(ventana, height=10, width=40, state=tk.DISABLED)
resultado_text.pack(pady=10)

# Iniciar el bucle principal de la aplicación
ventana.mainloop()
```

Definición de Patrones:

Se utilizan expresiones regulares para definir patrones que representan identificadores y números reales.

Función analizador_lexico:

Toma una expresión como entrada y utiliza expresiones regulares para identificar identificadores y números reales.

Procesa los tokens y construye una cadena de resultado.

Función analizar_expresion:

Se ejecuta al hacer clic en el botón "Analizar".

Obtiene la expresión ingresada, llama a analizador_lexico y muestra los resultados en un área de texto.

Interfaz Gráfica con Tkinter:

Se crea una interfaz gráfica con Tkinter que permite al usuario ingresar una expresión y analizarla léxicamente.

