

Tecnológico Nacional de México

Instituto Tecnológico de Pachuca

Ingeniería en Sistemas Computacionales

Materia: Lenguajes y autómatas

Documentación

Alumno:

Marco Antonio Lara López

José Pablo Martínez González

Yesenia Morales Ordoñez

SEMESTRE: Enero – junio 2024

14/05/2024

1.- Tabla de tokens

Una tabla de tokens en un analizador léxico es una estructura de datos utilizada para almacenar información sobre los tokens identificados durante el proceso de análisis léxico de un programa o texto. Un token es una secuencia de caracteres con un significado coherente en el lenguaje de programación o en el contexto de análisis.

```
tokens = [  
    ('numeros', r'\d+'),  
    ('minusculta', r'[a-z]'),  
    ('mayuscula', r'[A-Z]'),  
    ('especiales', r'[!&\_ | @#$%^~]'),  
    ('operadores', r'[=, -, *, /, =, <, >]'),  
]
```

2.- Reglas de coincidencia

son instrucciones definidas por el programador en un analizador léxico para identificar y clasificar los tokens en un flujo de caracteres. Estas reglas se utilizan para dividir el código fuente en unidades léxicas significativas, como palabras clave, identificadores, operadores, números, etc.

```
"r'\d+', r'[a-z]', r'[A-Z]', r'[!&\_ | @#$%^~]', r'[=, -, *, /, =, <, >]' "
```

3.- Asociar reglas

Implica definir qué hacer cuando se encuentra una coincidencia entre el patrón de una regla y el texto de entrada. Estas acciones pueden incluir la creación de un token, el registro de información adicional sobre el token, la ejecución de algún código específico o simplemente la omisión de ciertos caracteres.

```
password = input("Pon tu contraseña: ")  
if is_strong_password(password):  
    print("Tu contraseña es segura!")  
else:  
    print("Tu contraseña es débil. Aquí hay algunas sugerencias de mejora:")  
    print("- Utilice un mínimo de 12 caracteres.")  
    print("- Incluya letras minúsculas, letras mayúsculas, números y caracteres especiales.")
```

4.- Manejo de casos especiales

El manejo de casos especiales en un analizador léxico es crucial para garantizar que el análisis del código fuente se realice correctamente, incluso en situaciones que pueden desviarse de las reglas estándar.

```
tokens = [  
    ('numeros', r'\d+'),  
    ('letras', r'[a-zA-Z]'),  
    ('minuscula', r'[a-z]'),  
    ('mayuscula', r'[A-Z]'),  
    ('especiales', r'![&_\|@#\$%^~]'),  
    ('operadores', r'[=, -, *, /, =, <, >]'),  
]
```

5.- Pruebas de reglas

Las pruebas de las reglas en un analizador léxico son fundamentales para garantizar que el analizador funcione correctamente y produzca los tokens esperados para diferentes tipos de entradas.

```
Pon tu contraseña: Ma8*r3/1@  
Tu contraseña es débil. Aquí hay algunas sugerencias de mejora:  
- Utilice un mínimo de 12 caracteres.  
- Incluya letras minúsculas, letras mayúsculas, números y caracteres especiales.
```

```
Pon tu contraseña: Ma1e8o*r3/1@  
Tu contraseña es segura!
```