

Universidad Nacional Autónoma de México

Facultad de Ciencias

Compiladores 2025-1

Método de elementos punteados para la obtención del AFD

Alumno: Osorio Escandón Huriel

Observaremos el análisis léxico para un lenguaje simple que reconoce identificadores formados por letras, números enteros y decimales, operadores de asignación, operadores de suma y espacios en blanco.

Para el análisis se usará un Autómata Finito Determinista (AFD) que reconoce las diferentes categorías léxicas mediante el método de elementos punteados.

AFD

- **Estados:**

- q_0 : Estado inicial después de reconocer un token
- q_1 : Estado para reconocer identificadores formados solo por letras
- q_2 : Estado para reconocer números enteros
- q_3 : Estado para reconocer números decimales
- q_4 : Estado para reconocer al operador de asignación =
- q_5 : Estado para reconocer al operador de suma +

- **Transiciones:**

En q_0)

- Si se encuentra una letra, vamos a q_1 (identificador)
- Si se encuentra un dígito, vamos a q_2 (número entero)
- Si se encuentra un igual =, vamos a q_4 (operador de asignación)
- Si se encuentra un más +, vamos a q_5 (operador de suma)
- Si se encuentra un espacio (), vamos a q_5 (espacio ignorado)

En q_1) (identificadores)

- Si se encuentra una letra, nos quedamos en q_1
- Si encontramos cualquier otro carácter, regresamos a q_0 (id reconocido)

En q_2) (número entero)

- Si se encuentra un dígito, nos quedamos en q_2 (acumulando número entero)
- Si se encuentra un punto, vamos a q_3 (empieza un decimal)
- Si se encuentra cualquier otro carácter, regresamos a q_0 (número entero reconocido)

En q_3) (número decimal)

- Si se encuentra un dígito, nos quedamos en q_3 (acumulando número decimal)
- Si se encuentra cualquier otro carácter, regresamos a q_0 (número decimal reconocido)

En q_4) (operador de asignación)

- Regresamos a q_0 (operador de asignación reconocido)

En q_5) (operador de suma)

- Regresamos a q_0 (operador de suma reconocido)

Método de Elementos Punteados

Utilizando el ejemplo del código, $varA = 100 + 25.75 + varB$, se tiene lo siguiente:

- **Estado q_0 :** Encuentra “v” (letra), se tiene una transición a q_1 (comenzando identificador).
- **Estado q_1 :** Encuentra “arA” (letras), permanece en q_1 (identificador).
- **Estado q_1 :** Encuentra un espacio (), regresamos a q_0 (identificador **varA** reconocido).
- **Estado q_0 :** Encuentra “=”, se tiene una transición a q_4 (operador de asignación).
- **Estado q_4 :** Regresa a q_0 (operador de asignación reconocido).
- **Estado q_0 :** Encuentra “100” (dígitos) → transición a q_2 (comenzando número entero).
- **Estado q_2 :** Se queda en q_2 (número entero).
- **Estado q_2 :** Encuentra un espacio (), regresamos a q_0 (número entero **100** reconocido).
- **Estado q_0 :** Encuentra “+”, se tiene una transición a q_5 (operador de suma).
- **Estado q_5 :** Regresa a q_0 (operador de suma reconocido).
- **Estado q_0 :** Encuentra **25** (dígitos), se tiene una transición a q_2 (comenzando número entero).
- **Estado q_2 :** Encuentra “.”, se tiene una transición a q_3 (comenzando número decimal).
- **Estado q_3 :** Encuentra **75**, se queda en q_3 (número decimal).
- **Estado q_3 :** Encuentra un espacio (), regresamos a q_0 (número decimal **25.75** reconocido).
- **Estado q_0 :** Encuentra “+”, se tiene una transición a q_5 (operador de suma).
- **Estado q_5 :** Regresamos a q_0 (operador de suma reconocido).
- **Estado q_0 :** Encuentra **varB** (letras), se tiene una transición a q_1 (comenzando identificador).
- **Estado q_1 :** Encuentra el final de la cadena, entonces, regresamos a q_0 (identificador **varB** reconocido).