



**GOBIERNO DE
MÉXICO**

EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



**TECNOLÓGICO
NACIONAL DE MÉXICO**



Instituto Tecnológico de CD Madero.

Proyecto Lenguaje Automatas 1

ALUMNO

Rodrigo Adolfo Avila Chaverry.

Número de Control: 21070308.

MAESTRO

Fernando Manzanares González

Lenguaje Automatas 1

05-06-2024.

Ciudad Madero, Tamaulipas, México.



GOBIERNO DE
MÉXICO

EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO



Prueba Uno

```

50      self.tokens = tokens
51      self.pos = 0
  
```

Token: IDENTIFIER, Value: programTestProgram
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: varx
Token: DELIMITER, Value: ,
Token: IDENTIFIER, Value: y
Token: OPERATOR, Value: :
Token: KEYWORD, Value: integer
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: beginx
Token: OPERATOR, Value: :
Token: OPERATOR, Value: =
Token: NUMBER, Value: 10
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: y
Token: OPERATOR, Value: :
Token: OPERATOR, Value: =
Token: IDENTIFIER, Value: x
Token: OPERATOR, Value: +
Token: NUMBER, Value: 5
Token: DELIMITER, Value: ;
Token: KEYWORD, Value: end
Token: DELIMITER, Value: .
Parsing program

Analizador Léxico y Sintáctico para Pascal

```

programTestProgram;varx,y:integer;beginx:=10;y:=x+5;end.
  
```

Verificar Código

Prueba Dos

```

Token: IDENTIFIER, Value: programExampleProgram
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: vara
Token: DELIMITER, Value: ,
Token: IDENTIFIER, Value: b
Token: DELIMITER, Value: ,
Token: IDENTIFIER, Value: result
Token: OPERATOR, Value: :
Token: KEYWORD, Value: integer
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: procedureAddNumbers
Token: DELIMITER, Value: (
Token: IDENTIFIER, Value: x
Token: DELIMITER, Value: ,
Token: IDENTIFIER, Value: y
Token: OPERATOR, Value: :
Token: KEYWORD, Value: integer
Token: DELIMITER, Value: )
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: beginresult
Token: OPERATOR, Value: :
Token: OPERATOR, Value: =
Token: IDENTIFIER, Value: x
Token: OPERATOR, Value: +
Token: IDENTIFIER, Value: y
Token: DELIMITER, Value: ;
Token: KEYWORD, Value: end
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: begina
Token: OPERATOR, Value: :
Token: OPERATOR, Value: =
Token: NUMBER, Value: 10
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: b
Token: OPERATOR, Value: :
Token: OPERATOR, Value: =
Token: NUMBER, Value: 20
Token: DELIMITER, Value: ;
Token: IDENTIFIER, Value: AddNumbers
Token: DELIMITER, Value: (
  
```

Analizador Léxico y Sintáctico para Pascal

```

programExampleProgram;vara,b,result:integer;procedureAddNumbers(x,y:integer);beginresult:=x+y;end;begina:=10;b:=20;AddNumbers(a,b);end.
  
```

Verificar Código



Modelo de Autómata Finito.

Estados del Autómata Léxico

- **Inicio (S0):** Estado inicial del autómata.
- **Palabra Clave (S1):** Detecta palabras clave de Pascal.
- **Número (S2):** Detecta números enteros y reales.
- **Identificador (S3):** Detecta identificadores (nombres de variables y funciones).
- **Operador (S4):** Detecta operadores (+, -, *, /, :=, etc.).
- **Delimitador (S5):** Detecta delimitadores (punto y coma, coma, paréntesis, etc.).
- **Espacio en Blanco (S6):** Detecta y ignora espacios en blanco.
- **Desconocido (S7):** Detecta caracteres no reconocidos.

Transiciones

1. Desde **S0**:
 - Si se encuentra una palabra clave, ir a **S1**.
 - Si se encuentra un número, ir a **S2**.
 - Si se encuentra un identificador, ir a **S3**.
 - Si se encuentra un operador, ir a **S4**.
 - Si se encuentra un delimitador, ir a **S5**.
 - Si se encuentra un espacio en blanco, ir a **S6**.
 - Para cualquier otro carácter, ir a **S7**.
2. Desde **S1, S2, S3, S4, S5, S6, S7**:
 - Volver a **S0** después de tokenizar el carácter o la secuencia de caracteres actual.

Estados del Autómata Sintáctico

- **Inicio (S0):** Estado inicial del autómata.
- **Esperar Program (S1):** Espera la palabra clave program.



- **Esperar Identificador (S2):** Espera un identificador.
- **Esperar Punto y Coma (S3):** Espera un punto y coma ;.
- **Esperar Bloque (S4):** Espera un bloque de declaraciones y sentencias.
- **Esperar Punto (S5):** Espera un punto ..

Transiciones

3. Desde **S0** a **S1** si se encuentra el token **KEYWORD** con valor **program**.
4. Desde **S1** a **S2** si se encuentra el token **IDENTIFIER**.
5. Desde **S2** a **S3** si se encuentra el token **DELIMITER** con valor ;.
6. Desde **S3** a **S4** si se encuentra un bloque válido.
7. Desde **S4** a **S5** si se encuentra el token **DELIMITER** con valor ..



Diagrama del Autómata Léxico

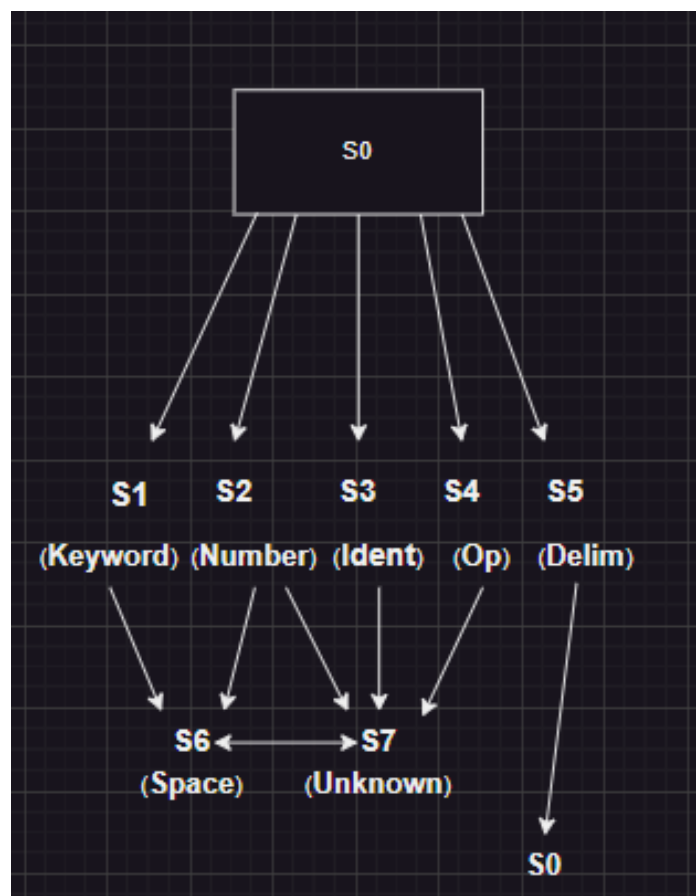
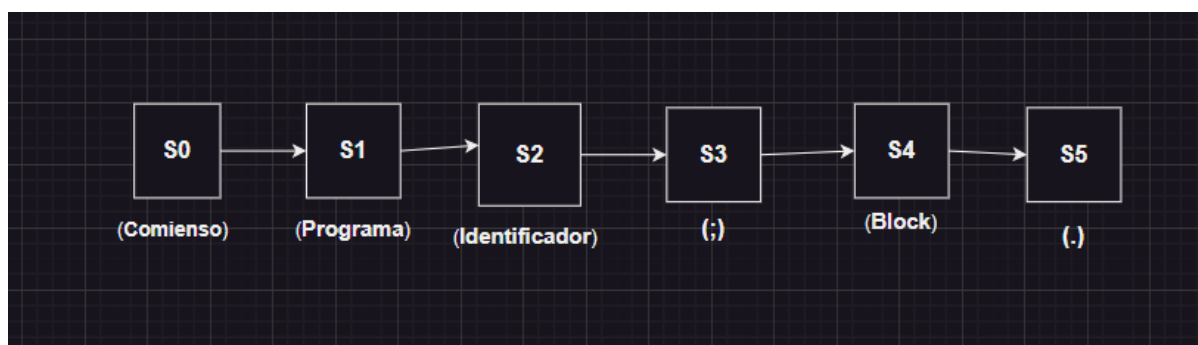


Diagrama del Autómata Sintáctico





Estado de transiciones: Detalles de las Transiciones Léxicas

Estado S0:

- Palabra clave (program|begin|end|...) -> S1
- Número (\d+(\.\d+)?) -> S2
- Identificador ([A-Za-z_][A-Za-z0-9_]*) -> S3
- Operador ([\+|-|*|/|:=<>]) -> S4
- Delimitador ([\;|\(|\)|\[\]|\.]) -> S5
- Espacio en blanco (\s+) -> S6
- Cualquier otro carácter -> S7

Estado S1, S2, S3, S4, S5, S6, S7:

- Cualquier entrada válida -> S0

Estado de transiciones: Detalles de las Transiciones Sintácticas

- **Estado S0:**
 - 'program' -> S1
- **Estado S1:**
 - Identificador -> S2
- **Estado S2:**
 - ';' -> S3
- **Estado S3:**
 - Bloque (inicio con 'begin') -> S4



**GOBIERNO DE
MÉXICO**

EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



**TECNOLÓGICO
NACIONAL DE MÉXICO**



- Estado S4:

- '!' -> S5