



GRAMATICA ESTRUCTURA CREATE TABLE

$G = (N, T, P, S)$

N (No terminales):

{ Escritura, A }

T (Terminales):

{ FIN, Id, Numero, Literal, Escribir }

S (Símbolo inicial):

{ Escritura }

P (Producciones):

Escritura \rightarrow Escribir A FIN

A \rightarrow Literal | Numero | Identificador (Id)

Tabla de Análisis Semántico

Terminales	Escribir	Literal	Numero	Id	FIN	\$
Escritura	Escribir A FIN					
A		A \rightarrow Literal	A \rightarrow Numero	A \rightarrow Id		

Estados:

- q0: Estado inicial.
- q1: Estado después de leer el terminal Escribir.
- q2: Estado después de leer A, que puede ser un Literal, Número, o Id.
- q3: Estado después de leer el terminal FIN (estado final).

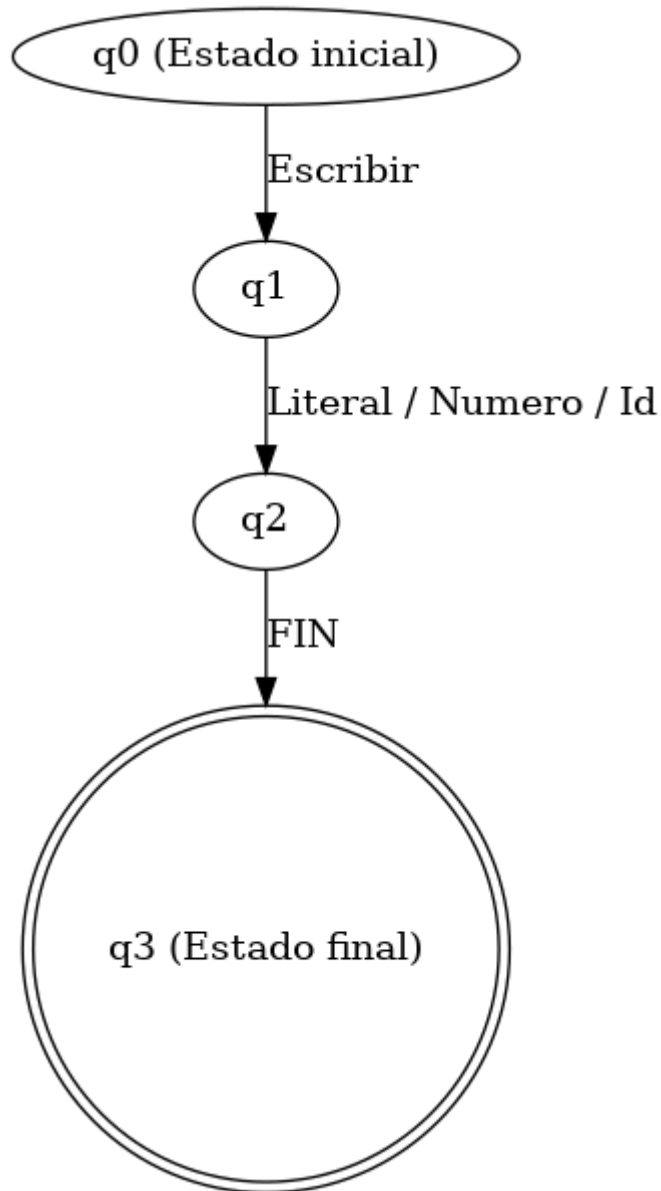
Transiciones:

1. Desde q0 a q1: Cuando se lee el terminal Escribir.
2. Desde q1 a q2: Cuando se lee un Literal, Número, o Id.
3. Desde q2 a q3: Cuando se lee el terminal FIN (estado de aceptación).



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala

Universidad San Carlos de Guatemala
Centro Universitario de Occidente
División de Ciencias de la Ingeniería
Ingeniería en Ciencia y Sistemas
Laboratorio de Lenguajes Formales y de
Programación
Proyecto Final



Gramática para Modificadores de Tablas

$G = (N, T, P, S)$

No terminales (N):

- **Modificador**
- **Tabla**
- **TipoDato**
- **Identificador**
- **Constraint**

Terminales (T):

- **ALTER**
- **TABLE**
- **ADD**
- **COLUMN**
- **ALTER**
- **DROP**
- **CONSTRAINT**
- **TYPE**
- **UNIQUE**
- **FOREIGN**
- **REFERENCES**
- **CASCADE**
- **IF**
- **EXISTS**
- **,**
- **;**

Símbolo inicial

S = Modificador

Producciones (P):

1. Modificador → ALTER TABLE Identificador ADD COLUMN Identificador TipoDato;
2. Modificador → ALTER TABLE Identificador ALTER COLUMN Identificador TYPE TipoDato;
3. Modificador → ALTER TABLE Identificador DROP COLUMN Identificador;
4. Modificador → ALTER TABLE Identificador ADD CONSTRAINT Identificador TipoDato;
5. Modificador → ALTER TABLE Identificador ADD CONSTRAINT Identificador UNIQUE (Identificador);
6. Modificador → ALTER TABLE Identificador ADD CONSTRAINT Identificador FOREIGN KEY (Identificador) REFERENCES Identificador (Identificador);
7. Modificador → DROP TABLE IF EXISTS Identificador CASCADE;

Tabla de Análisis Semántico

No Terminal	Primero	Siguiente
Modificador	ALTER	\$
Tabla	ALTER TABLE	\$
TipoDato	SERIAL, INTEGER, BIGINT, VARCHAR, DECIMAL, DATE, TEXT, BOOLEAN	;
Constraint	CONSTRAINT	;
Identificador	Id	;



Estados del Autómata

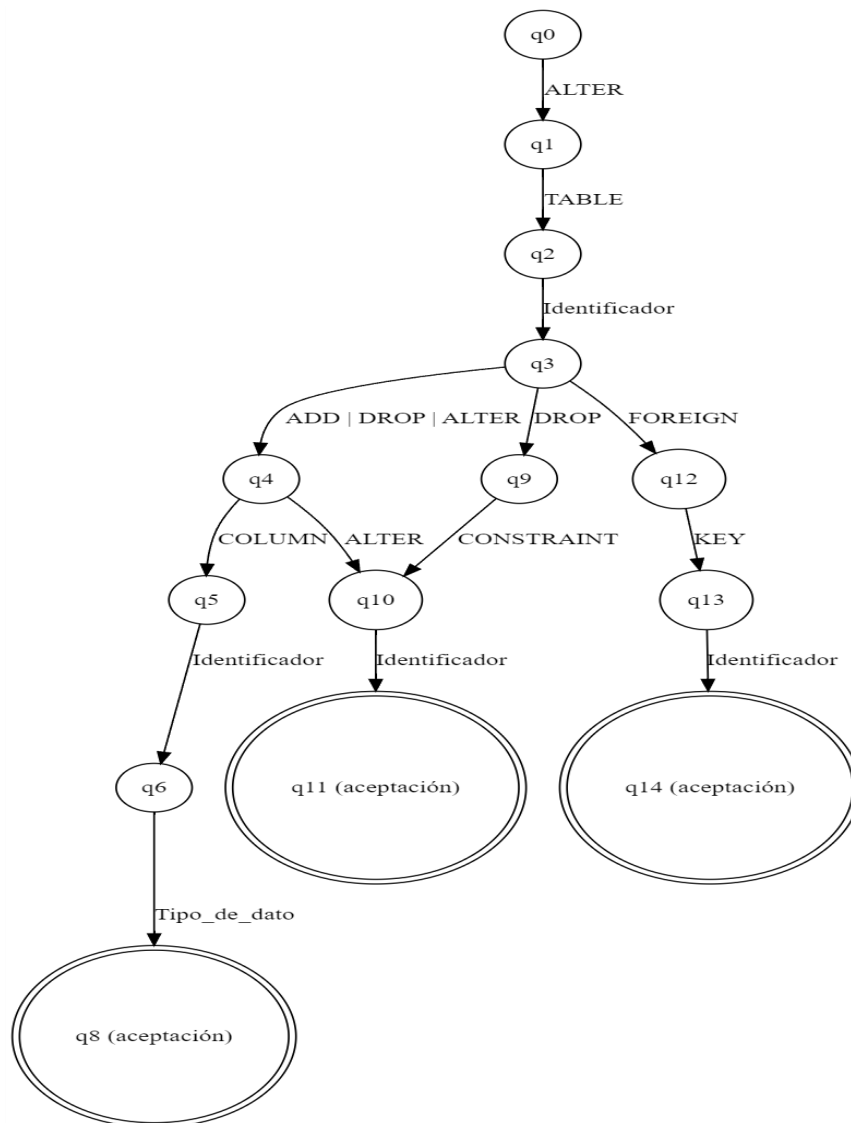
1. q0: Estado inicial.
2. q1: Después de leer ALTER.
3. q2: Después de leer TABLE.
4. q3: Después de leer un Identificador.
5. q4: Después de leer ADD.
6. q5: Después de leer COLUMN.
7. q6: Después de leer otro Identificador (nombre de la nueva columna).
8. q7: Después de leer TYPE.
9. q8: Después de leer un Tipo_de_dato.
10. q9: Después de leer DROP.
11. q10: Después de leer CONSTRAINT.
12. q11: Estado de aceptación para ADD COLUMN y ALTER COLUMN.
13. q12: Después de leer FOREIGN.
14. q13: Después de leer REFERENCES.
15. q14: Estado de aceptación para DROP TABLE.

Transiciones

- q0 → q1: ALTER
- q1 → q2: TABLE
- q2 → q3: Identificador
- q3 → q4: ADD / DROP / ALTER
- q4 → q5: COLUMN
- q5 → q6: Identificador
- q6 → q8: Tipo_de_dato (aceptación de ADD COLUMN)
- q3 → q7: DROP
- q4 → q9: ALTER
- q9 → q10: CONSTRAINT



- $q_{10} \rightarrow q_{11}$: Identificador (aceptación de ADD CONSTRAINT)
- $q_3 \rightarrow q_{12}$: FOREIGN
- $q_{12} \rightarrow q_{13}$: KEY
- $q_{13} \rightarrow q_{14}$: Identificador (aceptación de FOREIGN KEY)
- $q_2 \rightarrow q_{14}$: DROP (aceptación de DROP TABLE)





USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala

Universidad San Carlos de Guatemala
Centro Universitario de Occidente
División de Ciencias de la Ingeniería
Ingeniería en Ciencia y Sistemas
Laboratorio de Lenguajes Formales y de
Programación
Proyecto Final

Gramática para Inserciones

$G = (N, T, P, S)$

No terminales (N):

- Insercion
- ListalIdentificadores
- ListaValores
- Dato
- Expresion

Terminales (T):

- INSERT
- INTO
- VALUES
- ,
- ;
- (
-)
- TRUE
- FALSE
- Id
- Numero (para enteros y decimales)
- Cadena
- Fecha

Símbolo inicial

S = Insercion



Producciones (P):

1. Insercion \rightarrow INSERT INTO Identificador (ListalIdentificadores) VALUES (ListaValores);
2. Insercion \rightarrow INSERT INTO Identificador (ListalIdentificadores) VALUES (ListaValores), ... (ListaValores);
3. ListalIdentificadores \rightarrow Identificador , ListalIdentificadores | Identificador
4. ListaValores \rightarrow Dato , ListaValores | Dato
5. Dato \rightarrow Numero | Cadena | Fecha | TRUE | FALSE | Expresion
6. Expresion \rightarrow Dato + Dato | Dato - Dato | Dato * Dato | Dato / Dato
7. Expresion \rightarrow Dato < Dato | Dato > Dato | Dato <= Dato | Dato >= Dato | Dato == Dato | Dato <> Dato
8. Expresion \rightarrow Dato OR Dato | Dato AND Dato
9. Expresion \rightarrow (Expresion)

Estructura de Inserción Múltiple

$G = (N, T, P, S)$

No terminales (N):

- ListaValoresMultiple

Producciones (P):

- ListaValoresMultiple \rightarrow Dato , ListaValoresMultiple | Dato



Tabla de Análisis Semántico

No Terminal	Primero	Siguiente
Insercion	INSERT	\$
ListalIdentificadores	Id	VALUES
ListaValores	Numero, Cadena, Fecha, TRUE, FALSE, (,)
Dato	Numero, Cadena, Fecha, TRUE, FALSE, (,)
Expresion	Numero, TRUE, FALSE, (,)

Autómata para la Instrucción INSERT INTO

1. Definición de los Componentes del Autómata

- Estados (Q): Conjunto de estados del autómata.
- Alfabeto (Σ): Conjunto de símbolos de entrada.
- Transiciones (δ): Función de transición que define cómo se mueve de un estado a otro.
- Estado Inicial (q_0): Estado donde comienza el autómata.
- Estados Aceptantes (F): Conjunto de estados donde el autómata acepta la cadena.

Gramática para Lectura (SELECT)

$$G = (N, T, P, S)$$

Definición de los componentes

No terminales (N):

- Seleccion
- SeleccionDeColumna
- FuncionAgregacion
- Sentencias
- Join
- Where
- Group
- Order



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala

Universidad San Carlos de Guatemala
Centro Universitario de Occidente
División de Ciencias de la Ingeniería
Ingeniería en Ciencia y Sistemas
Laboratorio de Lenguajes Formales y de
Programación
Proyecto Final

- Limit
- Dato
- Identificador

Terminales (T):

- SELECT
- FROM
- *
- AS
- JOIN
- ON
- WHERE
- GROUP BY
- ORDER BY
- LIMIT
- DESC
- ASC
- Id (para identificadores)
- Numero (para enteros)}

Símbolo inicial

S = Seleccion

Producciones (P)

1. Seleccion \rightarrow SELECT * FROM Identificador [Sentencias];
2. Seleccion \rightarrow SELECT SeleccionDeColumna FROM Identificador [Sentencias];
3. SeleccionDeColumna \rightarrow Identificador | FuncionAgregacion | Identificador.Identificador | Identificador AS Identificador
4. FuncionAgregacion \rightarrow SUM(Identificador) | AVG(Identificador) | COUNT(Identificador) | MIN(Identificador) | MAX(Identificador)
5. Sentencias \rightarrow Join | Where | Group | Order | Limit | Join Sentencias | Where Sentencias | Group Sentencias | Order Sentencias | Limit Sentencias | ϵ
6. Join \rightarrow JOIN Identificador Identificador ON Identificador.Identificador = Identificador.Identificador
7. Join \rightarrow JOIN Identificador Identificador ON Identificador = Identificador.Identificador
8. Where \rightarrow WHERE Identificador = Dato | Identificador | Identificador.Identificador
9. Where \rightarrow WHERE Dato OpRacional AND Identificador = Identificador.Identificador
10. Group \rightarrow GROUP BY Identificador | Identificador.Identificador



11. Order → ORDER BY Identificador | Identificador.Identificador [DESC | ASC]
12. Limit → LIMIT Numero
13. Dato → Numero | Cadena | Fecha | TRUE | FALSE
14. OpRacional → < | > | <= | >= | = | <>

Tabla de Análisis Semántico

No Terminal	Primero	Siguiente
Selección	SELECT	\$
SelecciónDeColumna	*, Identificador, FuncionAgregacion	FROM
Sentencias	JOIN, WHERE, GROUP, ORDER, LIMIT	;
Join	JOIN	WHERE, GROUP, ORDER, LIMIT
Where	WHERE	GROUP, ORDER, LIMIT
Group	GROUP BY	ORDER, LIMIT
Order	ORDER BY	LIMIT
Limit	LIMIT	;
Dato	Numero, Cadena, TRUE, FALSE	
OpRacional	<, >, <=, >=, =, <>	AND, GROUP, ORDER, LIMIT

Componentes del Autómata

- Q0: Estado inicial
- Q1: Estado que espera SELECT
- Q2: Estado que espera * o una selección de columna
- Q3: Estado que espera FROM
- Q4: Estado que espera un identificador (tabla)
- Q5: Estado que espera sentencias adicionales
- Q6: Estado final (aceptación)
- Transiciones
- (Q0, SELECT) -> Q1
- *(Q1,) -> Q2
- (Q1, Identificador) -> Q2
- (Q2, FROM) -> Q3
- (Q3, Identificador) -> Q4
- (Q4, Join) -> Q5 (si hay un JOIN)
- (Q4, Where) -> Q5 (si hay un WHERE)
- (Q4, Group) -> Q5 (si hay un GROUP BY)



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala

Universidad San Carlos de Guatemala
Centro Universitario de Occidente
División de Ciencias de la Ingeniería
Ingeniería en Ciencia y Sistemas
Laboratorio de Lenguajes Formales y de
Programación
Proyecto Final

- (Q4, Order) -> Q5 (si hay un ORDER BY)
- (Q4, Limit) -> Q5 (si hay un LIMIT)
- (Q5, ;) -> Q6 (fin de la instrucción)

Gramática para Actualización (UPDATE)

G = (N, T, P, S)

No terminales (N):

- Actualizacion
- Set
- Where
- Dato
- Identificador

Terminales (T):

- UPDATE
- SET
- WHERE
- Id (para identificadores)
- Numero (para enteros)
- Cadena (para cadenas de texto)
- TRUE
- FALSE
- = (operador de asignación)

Símbolo inicial

- S = Actualizacion

Producciones (P)

1. Actualizacion → UPDATE Identificador SET Set [Where];
2. Set → Identificador = Dato | Identificador = Dato, Set



3. Where → WHERE Identificador = Dato | Identificador |
Identificador.Identificador
4. Where → WHERE Dato OpRacional Identificador =
Identificador.Identificador
5. Dato → Numero | Cadena | TRUE | FALSE
6. OpRacional → < | > | <= | >= | = | <>

Símbolo inicial

- S = Actualizacion

No Terminal	Primero	Siguiente
Actualizacion	UPDATE	\$
Set	Identificador	WHERE
Where	WHERE	;
Dato	Numero, Cadena, TRUE, FALSE	
OpRacional	<, >, <=, >=, =, <>	AND

