AUTÓMATAS A PILA (AP)

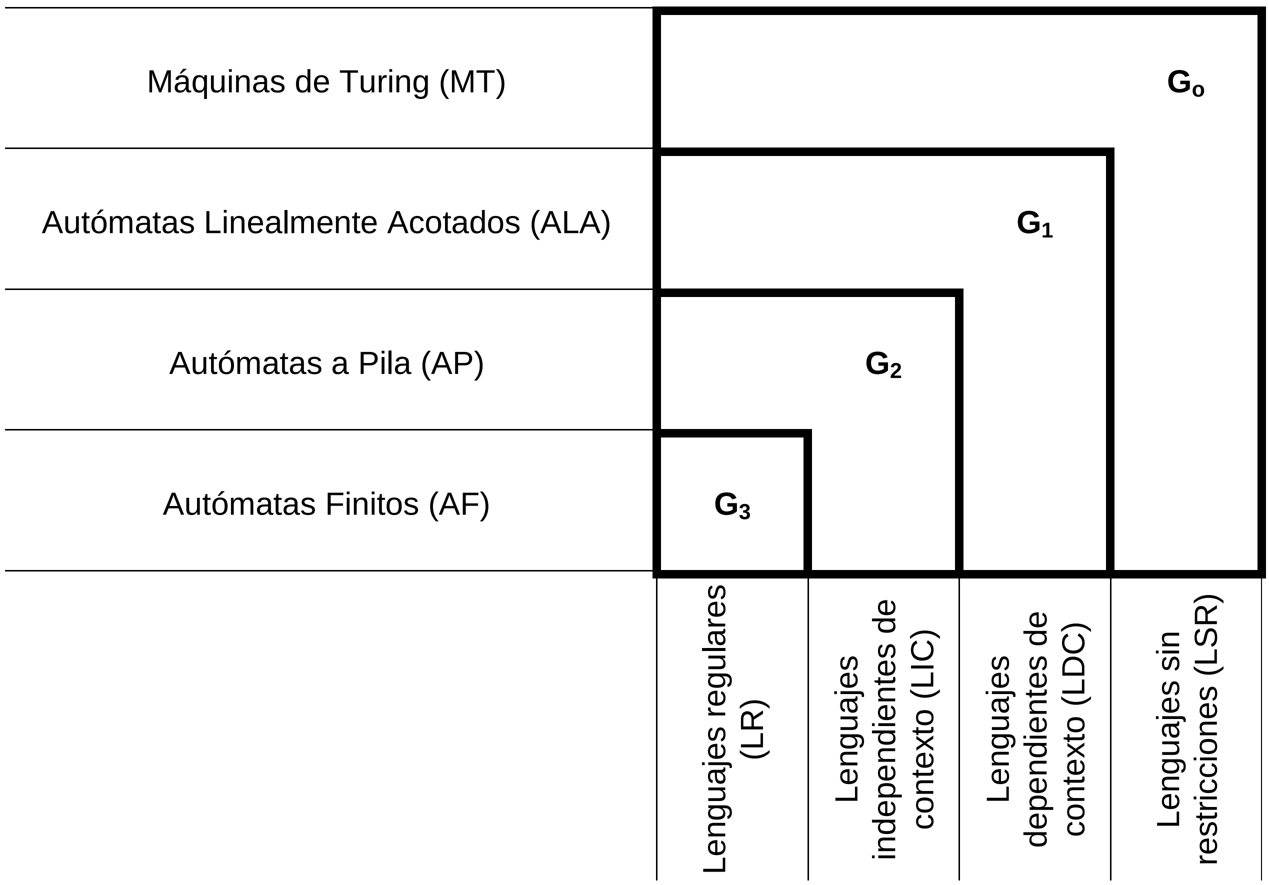
(Tema 5)

ÍNDICE

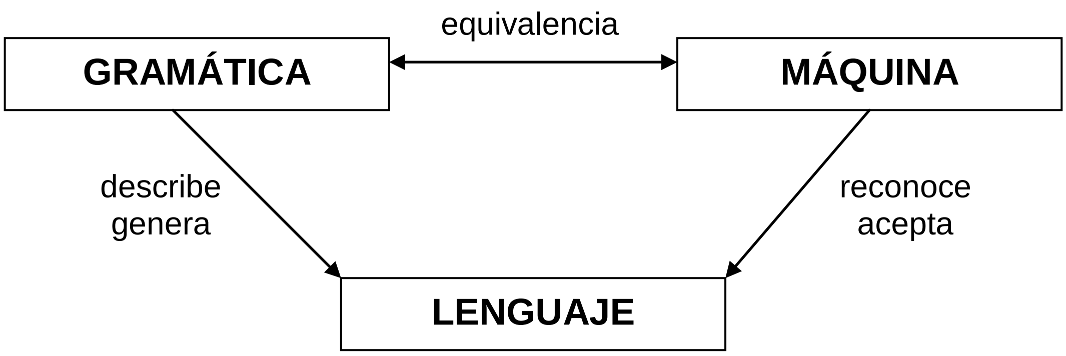
1. INTRODUCCIÓN

MAQUINÁS GRAMÁTICAS Y LENGUAJES:

Jerarquía de CHOMSKY



$$G_3 \subset G_2 \subset G_1 \subset G_0$$



Definición

Autómata Finito (AF) que tiene acceso (controla) a una memoria intermedia (Pila) y que lee símbolos en una cinta de entrada (Σ).

PILA: tiene el tamaño que sea necesario (incluso ∞).

LIFO: Last Input Firt Ouput

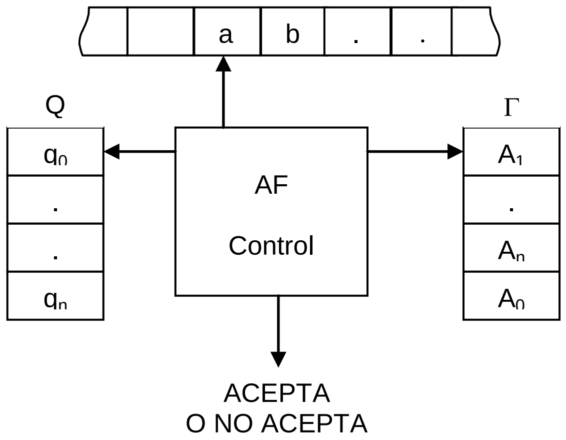
AP No Deterministas: Lenguajes Independientes de Contexto (Tipo 2)

$A ::= v \text{ , } v \in (\Sigma_T \cup \Sigma_N)^*$

AP Deterministas: Subconjunto de los Lenguajes de Tipo 2

$L(APD) \subset L(APND)$

ESQUEMA GRÁFICO de un AP



Operaciones en la pila:

Escribir ——— Introducir

Leer ——— Extraer (la lectura es <u>destructiva</u>)

Autómata a Pila (DEF. FORMAL)

Un Autómata a Pila es una SEPTUPLA.

$AP = \{ \Sigma , \Gamma , Q , A_0 , q_0 , f , \varnothing \}$

Σ = alfabeto de entrada: $a, b, c \in \Sigma$

$x, y, z \in \Sigma^*$

Γ = alfabeto de pila: $A, B, C \in \Gamma$

$X, Y, Z \in \Gamma^*$

Q = Conjunto de estados (finito) $q_0, q_1, q_2, \ldots, q_n \in Q$

A_0 = Símbolo inicial de pila ($A_0 \in \Gamma$)

q_0 = estado inicial $q_0 \in Q$

f = función de transición de estados:

$$f(Q \times \Sigma \times \Gamma) \longrightarrow \mathscr{P}(Q \times \Gamma^*)$$

F = Conjunto de estados finales ($F \subset Q$)

1.1. Funcionamiento (Movimientos)

En un AP existen 2 tipos de movimientos:

Movimientos de tipo 1:

f (q a A) = { (q_1 Z_1), (q_2 Z_2), (q_3 Z_3),…………….. (q_n Z_n) }

$q, q_1, q_2, q_3 \ldots, q_n \in Q$

$A \in \Gamma; Z_1, Z_2, Z_3 \ldots, Z_n \in \Gamma^*$

$a \in \Sigma$

Movimientos de tipo 2:

f (q λ A) = { (q_1 Z_1), (q_2 Z_2), (q_3 Z_3),…………….. (q_n Z_n) }

$q, q_1, q_2, q_3 \ldots, q_n \in Q$

$A \in \Gamma; Z_1, Z_2, Z_3 \ldots, Z_n \in \Gamma^*$

1er EJEMPLO:

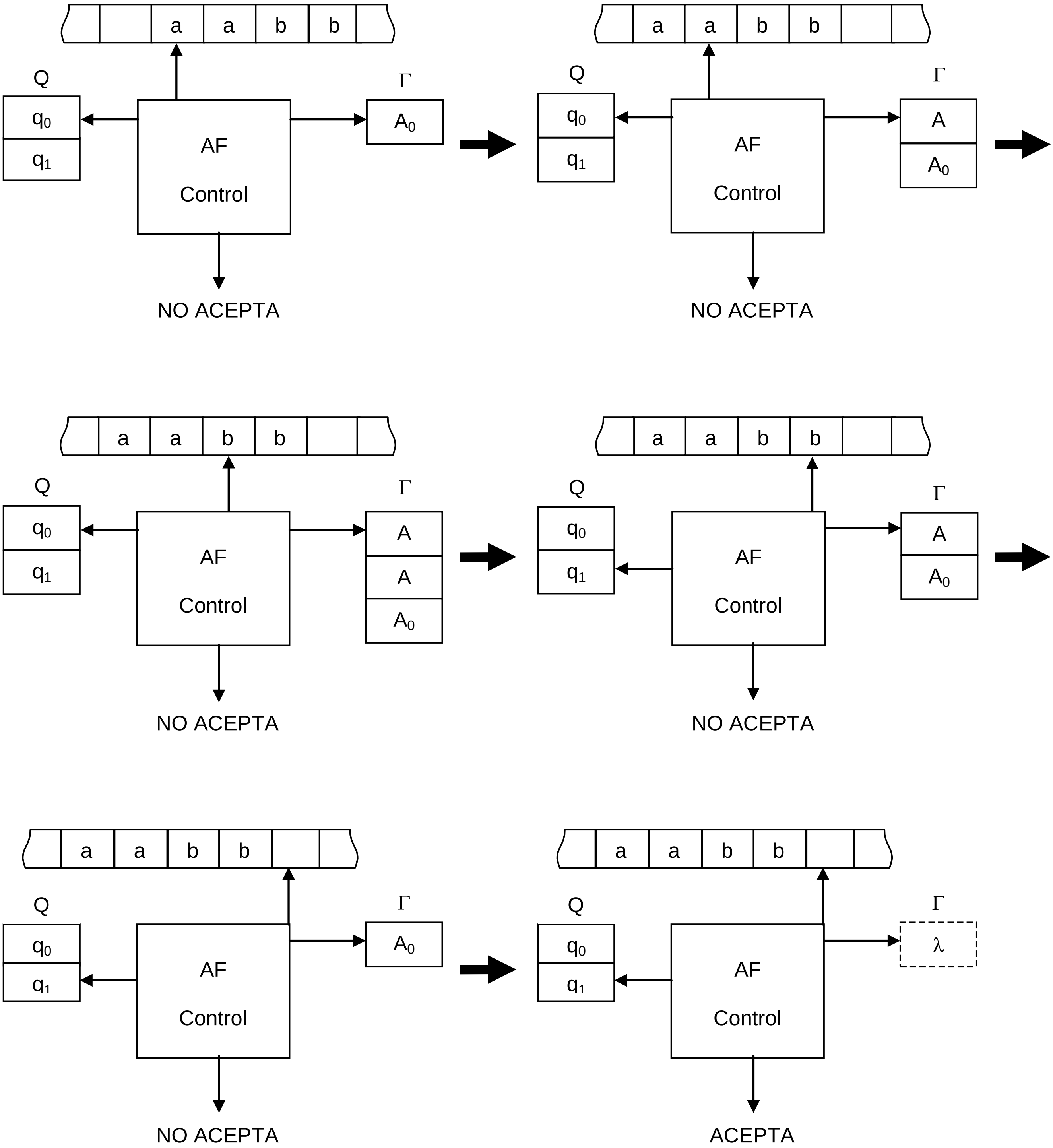
Construir un AP que reconozca el lenguaje $L = \{ a^n b^n / n \geq 1 \}$

AP = { Σ , Γ , Q , A_0 , q_0 , f , $\varnothing$ }

AP = { { a, b }, { A, A_0 } , { q_0, q_1 } , A_0 , q_0 , f , $\varnothing$ }

1) f (q_0 a A_0) = (q_0 AA_0)	Lee a mete A
2) f (q_0 a A) = (q_0 AA)	Lee a mete A (resto a leídas)
3) f (q_0 b A) = (q_1 λ)	Lee b borra A (pasa a q_1)
4) f (q_1 b A) = (q_1 λ)	Lee b borra A (resto b leídas)
5) f (q_1 λ A_0) = (q_1 λ)	Borra A_0 (vacía la pila) | Acepta | Para

PALABRA: aabb



1.2. Configuraciones (Descripciones Instantáneas):

Describen la aceptación o rechazo de una palabra $x \in \Sigma$.

Se representan por una terna: $(q \quad x \quad Z)$

q = Estado actual del AP

x = palabra (o subpalabra) por leer

Z = Contenido actual de la pila

Movimientos de tipo 1 con descripciones instantáneas:

$(q \ \ ax \ \ AZ) \vdash (p \ \ x \ \ YZ)$

Cuando el movimiento del AP es:

$(p \ \ Y) \in f (q \ \ a \ \ A)$

$q, p \in Q$; $A \in \Gamma$; $Y, Z \in \Gamma^*$; $a \in \Sigma$; $x \in \Sigma^*$

Movimientos de tipo 2 con descripciones instantáneas:

$(q \ \ ax \ \ AZ) \vdash (p \ \ ax \ \ YZ)$

Cuando el movimiento del AP es:

$(p \ \ Y) \in f (q \ \ \lambda \ \ A)$

$q, p \in Q$; $A \in \Gamma$; $Y, Z \in \Gamma^*$; $a \in \Sigma$; $x \in \Sigma^*$

El símbolo $\vdash$ significa PRECEDE

$(DI_1) \vdash (DI_2) \vdash (DI_3) \vdash (DI_4) \ldots\ldots\ldots\ldots \vdash (DI_n)$

$(DI_1) \vdash_* (DI_n)$

El símbolo $\vdash_*$ significa SUCESIÓN DE MOVIMIENTOS

1.3. Lenguajes aceptados por un AP:

Son de 2 tipos:

- Estados finales

$$L = \{ x \ / \ (q_0 \ x \ A_0) \vdash^{*} (p \ \lambda \ Z) \}$$

$q_0 \in Q$, $p \in F \subset Q$; $A_0 \in \Gamma$; $x \in \Sigma \cup \{ \lambda \}$

- Vaciado de Pila

$$L = \{ x \ / \ (q_0 \ x \ A_0) \vdash^{*} (p \ \lambda \ \lambda) \}$$

$q_0 , p \in Q$; $A_0 \in \Gamma$; $x \in \Sigma \cup \{ \lambda \}$

1.4. Autómatas a Pila Deterministas (APD):

Debe cumplir 2 condiciones:

1. $\forall \ q \in Q , A \in \Gamma$

   Si $| f (q \ \lambda \ A) | > 0$

   entonces $\forall \ a \in \Sigma$, $f (q \ A \ q) = \varnothing$

2. $\forall \ q \in Q , A \in \Gamma$

   $| f (q \ a \ A) | < 2$; $\forall \ a \in \Sigma \cup \{ \lambda \}$

2. LENGUAJES INDEPENDIENTES DE CONTEXTO (LIC) Y AUTÓMATAS A PILA (AP)

2.1. Equivalencia entre aceptación de lenguajes por un AP, LV(AP) = LF(AP)

2.1.1. LV(AP) $\subset$ LF(AP)

2.1.2. LV(AP) $\supset$ LF(AP)

2.1.1. Caso "$\subset$": LV(AP) $\subset$ LF(AP)

A_1 es un AP, si $L = LV(A_1) \Rightarrow \exists\, A_2 \,/\, LF(A_2) = L$

$A_1 = \{ \Sigma, \Gamma, Q, q_0, A_0, f, \varnothing \}$,

$A_2 = \{ \Sigma, \Gamma \cup \{ A_0' \}, Q \cup \{ q_0', q_F \}, q_0', A_0', f', \{ q_F \} \}$

Paso 1: A_2 accede a DII de A_1 y comienza a simular a A_1

$f'(\, q_0' \;\; \lambda \;\; A_0') = (\, q_0 \;\; A_0A_0')$

Paso 2: A_2 realiza los mismos movimientos que A_1

$f'(\, q \;\; a \;\; A\,) = f\,(\, q \;\; a \;\; A\,) \;\; \forall\, q \in Q\, ;\, a \in \Sigma \cup \{ \lambda \}\, ;\, A \in \Gamma$

Con los pasos 1 y 2:

A_1 con una palabra $x \in \Sigma$ borra A_0 y acepta la palabra.

A_2 con una palabra $x \in \Sigma$ borra A_0 y aflora A_0' a la cima de la pila.

Paso 3: Para aceptar una palabra x, A_2 debe realizar un λ-movimiento borrar A_0' y acceder al estado final q_F.

$(\, q_F \;\; \lambda\,) \in f'(\, q \;\; \lambda \;\; A_0') \qquad \forall\, q \in Q$

2.1.2. Caso "$\supset$": LV(AP) $\supset$ LF(AP)

M_1 es un AP, si L = LF(M_1) $\Rightarrow \exists$ M_2 / LV(M_2) = L

$M_1 = \{ \Sigma, \Gamma, Q, q_0, A_0, f, F \}$,
$M_2 = \{ \Sigma, \Gamma \cup \{ A_0' \}, Q \cup \{ q_0', q_v \}, q_0', A_0', f', \varnothing \}$

Paso 1: M_2 accede a DII de M_1 y comienza a emular a M_1

$f'(q_0' \;\; \lambda \;\; A_0') = (q_0 \;\; A_0 A_0')$

Paso 2: M_2 realiza los mismos movimientos que M_1

$f'(q \;\; a \;\; A) = f (q \;\; a \;\; A) \;\; \forall\, q \in Q \,;\, a \in \Sigma \cup \{ \lambda \} \,;\, A \in \Gamma$

Paso 3: a) M_2 debe transitar al estado q_v cuando M_1 alcanza un estado final q. $(q_v \;\; \lambda) \in f'(q \;\; \lambda \;\; Z) \;\; \forall\, q \in F \,;\, Z \in \Gamma^* \cup \{ A_0' \}$

b) M_2 debe permanecer en q_v y borrar símbolo a símbolo hasta vaciar totalmente la pila. $(q_v \;\; \lambda) \in f'(q_v \;\; \lambda \;\; Z)$
$\forall\, q_v \in Q \,;\, Z \in \Gamma^* \cup \{ A_0' \}$

2.2. LOS LENGUAJES INDEPENDIENTES DE CONTEXTO (LIC) SON ACEPTADOS POR UN AUTÓMATA A PILA (AP) Y GENERADOS POR UNA GRAMÁTICA DE TIPO 2 (G_2): LIC = L(AP) = L(G_2)

2.2.1. L(AP) $\subset$ L(G_2)

2.2.2. L(AP) $\supset$ L(G_2)

2.2.1. Caso "$\subset$": L(AP) $\subset$ L(G_2)

$L = L(AP) \implies \exists\, G_2 \,/\, L(G_2) = L$

Se va a construir una gramática (G_2) obtenida a partir de un autómata a pila (AP) que acepte el mismo lenguaje L.

2.2.2. Caso "$\supset$": L(AP) $\supset$ L(G_2)

$L = L(G_2) \implies \exists\, AP \,/\, L(AP) = L$

Se va a construir un autómata a pila (AP) obtenido a partir de una gramática (G_2) que genere el mismo lenguaje L.

2.2.1. Caso "$\subset$": $L(AP) \subset L(G_2)$. CONSTRUCCIÓN DE UNA GRAMÁTICA (G_2) OBTENIDA A PARTIR DE UN AP.

Se debe construir una gramática que genere el lenguaje aceptado por un autómata a pila: $AP = \{ \Sigma , \Gamma , Q , A_0 , q_0 , f , \varnothing \}$

$G = \{ \Sigma_T , \Sigma_N , \mathscr{P} , S \}$

$\Sigma_T = \Sigma$ del AP

$\Sigma_N = \{ S \} \cup \{ [q \;\; A \;\; p] \;\; / \;\; q, p \in Q , A \in \Gamma \}$

S = Axioma

$\mathscr{P}$ = Producciones de G

<u>ALGORITMO</u> (para obtener las producciones $\mathscr{P}$ de G):

1. Desde el axioma S, se genera la siguiente producción:

   $S ::= [q_0 \;\; A_0 \;\; p] , \forall \; p \in Q$

2. Se construyen producciones de la forma:

   $[q \;\; A \;\; q_{m+1}] ::= a [q_1 \;\; B_1 \;\; q_2] , [q_2 \;\; B_2 \;\; q_3] , ... , [q_m \;\; B_m \;\; q_{m+1}]$

   Todas las posibles secuencias que llevan de q_1 a q_{m+1} para cada símbolo $B_1, B_2, ... , B_m$ introducido en la pila

   $\forall \; q_1, q_2, q_3, ... , q_m, q_{m+1} \in Q$

   $A, B_1, B_2, B_3, ... , B_m \in \Gamma$

   $a \in \Sigma \cup \{ \lambda \}$

   si los movimientos de la función de transición f son del tipo:
   $(q_1 \;\; B_1B_2B_3 ... B_m) \in f (q \;\; a \;\; A)$

3. Si m = 0, es decir, no se introduce nada en la pila, se construyen producciones de la forma:

   $[q \;\; A \;\; q_i] ::= a \;\;, \forall \; q_i \in Q$

   si los movimientos de la función de transición f son del tipo:
   $(q_i \;\; \lambda) \in f (q \;\; a \;\; A)$

Símbolos no-terminales (TERNAS)

Los símbolos no-terminales $\in \Sigma_N$ de una gramática (G) construida a partir de un AP tienen la forma de TERNAS:

$$[\, q \quad A \quad p \,]$$

q = Estado actual del autómata a pila (AP)

A = Símbolo de la cima de pila

p = Estado al que llega el autómata a pila (AP) cuando se hayan suprimido todos los símbolos que se hayan podido introducir en la pila al borrar el símbolo A.

- Las TERNAS están asociadas a la supresión de un símbolo de pila A.
- Las TERNAS dan información de los estados en que se encuentra el AP antes y después de la supresión del símbolo A.

AUTÓMATA A PILA (AP)	GRAMATICA (G)
Elimina A de la pila	2º componente de la TERNA en la parte IZQUIERDA de la producción $[\, q \; \boxed{A} \; p \,] ::=$
Introduce $B_1B_2B_3 \ldots B_m$ en la pila (que luego se eliminan con sucesivos movimientos)	2º componente/s de las TERNA/S en la parte DERECHA de la producción $::= [\, q \; \boxed{B_1} \; p \,][\, q \; \boxed{B_2} \; p \,] \ldots [\, q \; \boxed{B_m} \; p \,]$
Se lee $a \in \Sigma$	Se genera $a \in \Sigma_T$

METODO REDUCIDO:

Se aplica el método anterior pero tras ver que símbolos no-terminales (TERNAS) son **válidos** para construir las producciones de la G_2 a partir de los movimientos del AP.

Se construyen todas las posibles ternas que se pueden formar con los estados (Q) y los símbolos de pila (Γ) y se analizan (validez).

2.2.2. Caso "⊃": L(AP) ⊃ L(G_2). CONSTRUCCIÓN DE UN AP A PARTIR DE UNA GRAMÁTICA (G_2).

Existen 2 métodos para construir un AP a partir de una gramática G:
MÉTODO 1 y MÉTODO 2

<u>MÉTODO 1</u>

Construir un AP que acepte (reconozca) el lenguaje generado por la gramática: G = { Σ_T, Σ_N, $\mathscr{P}$, S }. <u>La gramática ha de estar en FNG</u>.

AP = { Σ_T , Σ_N , { q } , S , q , *f* , ∅ }

Σ_T = alfabeto de entrada (Σ)

Σ_N = alfabeto de pila (Γ)

{ q } = Q conjunto de estados de AP

S = Símbolo de inicio de pila

q = estado inicial de AP

f = Función de transición (movimientos)

F = ∅

ALGORITMO (para obtener los movimientos del AP):

1. Si A : : = aZ , $a \in \Sigma_T$, $A \in \Sigma_N$, $Z \in \Sigma_N^*$

   entonces, (q Z) ∈ *f* (q a A)

2. Si S : : = λ

   entonces, (q λ) ∈ *f* (q λ S)

MÉTODO 2

Construir un AP que acepte (reconozca) el lenguaje generado por la gramática: G = { Σ_T, Σ_N, $\mathscr{P}$, S }. La gramática puede estar o no en FNG.

AP = { Σ_T , { $\Sigma_N \cup \Sigma_T$ } , { q } , S , q , *f* , $\varnothing$ }

Σ_T = Alfabeto de entrada (Σ)

{ $\Sigma_N \cup \Sigma_T$ } = Alfabeto de pila (Γ)

{ q } = Q (Conjunto de estados del AP)

S = Símbolo de inicio de pila

q = estado inicial del AP

f = Función de transición (movimientos)

F = $\varnothing$ (Conjunto de estados finales)

ALGORITMO (para obtener los movimientos del AP):

1. $X \in \{ \Sigma_N \cup \Sigma_T \}$, $A \in \Sigma_N$

   $\forall$ A : : = X producción de la gramática,

   en AP se hace: $(q \;\; X) \in f (q \;\; \lambda \;\; A)$

2. $\forall\, a \in \Sigma_T$

   entonces, $(q \;\; \lambda) \in f (q \;\; a \;\; a)$