Teoría de Lenguajes

Autómatas Push-Down

Gramáticas Libres de Contexto

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

Es un formalismo para especificar lenguajes

- V : conjunto finito de variables
- T : conjunto finito de terminales ($T \equiv \Sigma$)
- P : conjunto de reglas de producción
- S : símbolo inicial $S \in V$

$$\boxed{\begin{array}{c} A \rightarrow \alpha \;/\; \alpha \in (V \cup T)^* \\ A \in V \end{array}}$$

Lenguaje Libre de Contexto

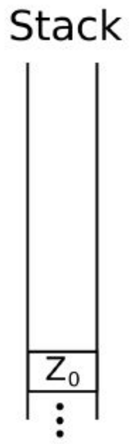
Es aquél que puede ser generado por una gramática libre de contexto
G:(V,T,P,S)

$$L(G) = \{ x \in T^* \ / \ S \overset{*}{\Rightarrow} x \}$$

Autómata Push-Down

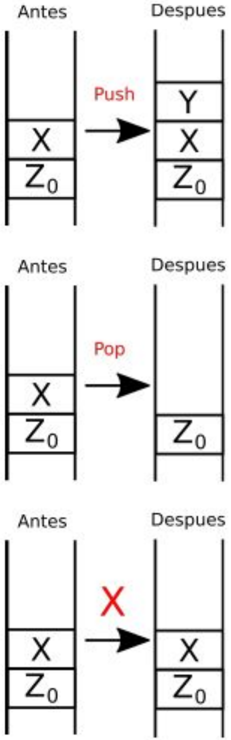
Es un autómata $M : (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ donde :

- Q - Conjunto finito de estados
- Σ - Alfabeto de entrada
- Γ - Alfabeto del stack
- δ - Función de transición
- q_0 - Estado inicial
- Z_0 - Simbol inicial de la stack
- F - Conjunto de estados finales



Autómata Push-Down

- $\delta : Q \times (\Sigma \cup \varepsilon) \times \Gamma \to 2^{Q \times \Gamma^*}$
- $\delta(q, a, X) = \{\{p_1, \alpha_1\}, \{p_2, \alpha_2\}, ..., \{p_n, \alpha_n\}\}$
- α_i es de la forma:
  - YX
  - ε
  - X (Queda como estaba)



Autómata Push-Down

$$DI_0 := (q_0, w, Z_0)$$

Descripción Instantánea inicial

Descripción instantánea

$$(q, w, \alpha)$$

estado entrada stack

antes de la transición

Ejemplos:

- $\delta(q,a,X) = \{(p_1,\beta_1), \dots ,(p_k,\beta_k)\} \Rightarrow \boxed{(q, ay, X\theta) \vdash (p_2, y, \beta_2\theta)}$

después de ejecutarla

- $\delta(q,\varepsilon,X) = \{(p_1,\beta_1), \dots ,(p_k,\beta_k)\} \Rightarrow \boxed{(q, ay, X\theta) \vdash (p_2, ay, \beta_2\theta)}$

$\theta \in \Gamma^*$

Autómata Push-Down

Dado $M : (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

Lenguaje aceptado por estado final

$$L(M) = \{x, x \in \Sigma^* / (q_0, x, Z_0) \overset{*}{\vdash} (q, \varepsilon, \alpha), q \in F, \alpha \in \Gamma^*\}$$

Lenguaje aceptado por stack vacío

$$L(M) = \{x, x \in \Sigma^* / (q_0, x, Z_0) \overset{*}{\vdash} (p, \varepsilon, \varepsilon), p \in Q\}$$

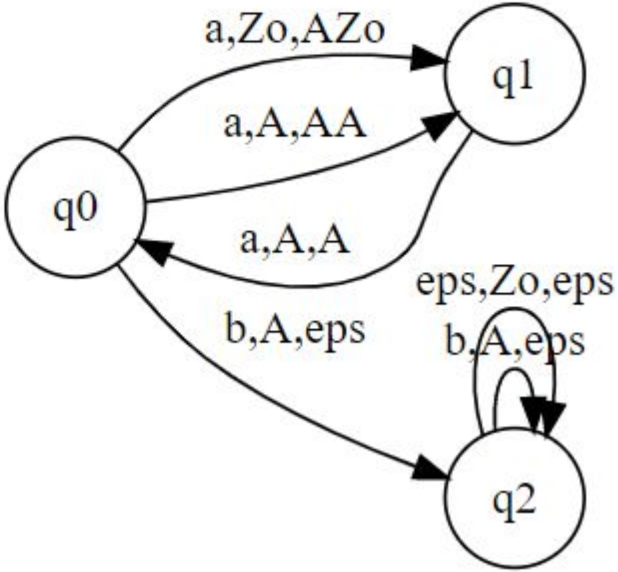
Autómata Push-Down Determinista

Dado $M : (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

- $\forall q \in Q \ y \ Z \in \Gamma : \ si \ \delta(q, \varepsilon, Z) \neq \emptyset \Rightarrow \forall a \in \Sigma, \ \delta(q, a, Z) = \emptyset$
- $\forall \ q \in Q, \ Z \in \Gamma \ y \ a \in \Sigma \cup \{\varepsilon\} \ / \ \#(\delta(q, a, Z)) \leqslant 1$

Ejemplo

Sea $L = \{ x / x \in \{a,b\}^*$ y es de la forma $a^{2k}b^k$, $k > 0 \}$

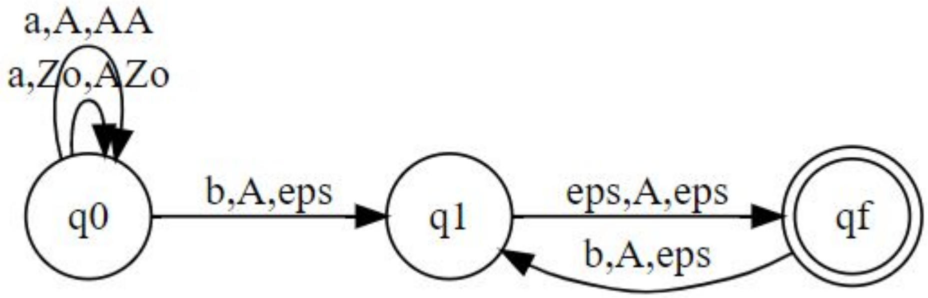


Transiciones épsilon

Son aquellas que me permiten modificar el stack y eventualmente cambiar de estado SIN consumir un símbolo de la tira

Ejemplo de uso:

L = { x / x $\in$ {a,b}* y es de la forma $a^{2k}b^k$, k >0}



Equivalencia de modelos

Todo APD que acepta un lenguaje por estado final, tiene un APD equivalente (acepta el mismo lenguaje) por stack vacío y recíprocamente

Equivalencia de modelos

APD M: $(Q, \Sigma, \Gamma, \delta, q_o, Z_o, F)$ ➔ APD M': $(Q', \Sigma, \Gamma', \delta', q'_o, Z'_o, \varnothing)$

- Definimos otro stack (un nuevo tope: Z'_o) [$\Gamma' = \Gamma \cup \{Z'_o\}$]
- Hacemos "push" del Z_o en este nuevo stack: $\delta'(q'_o, \varepsilon, Z'_o) = (q_o, Z_o Z'_o)$
- Entramos en las transiciones de δ: $\delta'(q,a,X) = \delta(q,a,X)$ [$Q' = Q \cup \{q'_o, q'_s\}$]
- $\forall\ q \in F$: $\delta'(q, \varepsilon, X) = (q'_s, \varepsilon)$ $\forall\ X \in \Gamma$ (incluído Z_o)

  $\delta'(q'_s, \varepsilon, X) = (q'_s, \varepsilon)$ $\forall\ X \in \Gamma'$ (incluído Z'_o)

Equivalencia de modelos

APD M: $(Q, \Sigma, \Gamma, \delta, q_o, Z_o, \varnothing)$ → APD M': $(Q', \Sigma, \Gamma', \delta', q'_o, Z'_o, \{q'_f\})$

- Definimos otro stack (un nuevo tope: Z'_o) [$\Gamma' = \Gamma \cup \{Z'_o\}$]
- Hacemos "push" del Z_o en este nuevo stack: $\delta'(q'_o, \varepsilon, Z'_o) = (q_o, Z_o Z'_o)$
- Entramos en las transiciones de δ: $\delta'(q,a,X) = \delta(q,a,X)$ [$Q' = Q \cup \{q'_o, q'_f\}$]
- $\forall q \in Q$: $\delta'(q, \varepsilon, Z'_o) = (q'_f, \varepsilon)$ $\forall q \in Q$

Equivalencia entre Gramáticas y Autómatas

Todo lenguaje generado por una gramática libre de contexto tiene asociado un APD (por stack vacío) que lo reconoce y recíprocamente

($\Rightarrow$)

L=L(G), G:(V,T,P,S) Libre de Contexto (en FNG) $\Rightarrow$ L=L(M), M: $(Q, \Sigma, \Gamma, \delta, q_o, Z_o, \emptyset)$

- M simula derivaciones de más a la izquierda de G

Equivalencia entre Gramáticas y Autómatas

Todo lenguaje generado por una gramática libre de contexto tiene asociado un APD (por stack vacío) que lo reconoce y recíprocamente

($\Leftarrow$)

$\mathcal{L}$=L(M), M: (Q, Σ, Γ, δ, q_o, Z_o, ∅) $\Rightarrow$ $\mathcal{L}$=L(G), G:(V,T,P,S) Libre de Contexto

- Las reglas se construyen a partir de las transiciones