

Automate push-down (APD)

.

Automat Push Down (APD)

Definitie:

Un automat push-down (APD) este un ansamblu

$M = (Q, \Sigma, \Gamma, \delta, q_o, Z_o, F)$, unde:

- Q alfabetul starilor;
- Σ alfabetul de intrare;
- Γ alfabetul memoriei stivă; ;
- $q_o \in Q$ stare inițială;
- $Z_o \in \Gamma$ simbolul de start al memoriei stivă;
- $F \subseteq Q$ mulțimea stărilor finale;
- $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \rightarrow \mathcal{P}_o(Q \times \Gamma^*)$ funcția de tranziție

Reprezentare

- enumerare
- tabelara
- sub forma de graf

Reprezentare folosind enumerare

Exemplu:

- $M = (\{q_0, q_1\}, \{a, b\}, \{A, Z\}, \delta, q_0, Z, \{q_0\})$

- δ :

$$\delta(q_0, a, Z) = \{(q_0, AZ)\}$$

$$\delta(q_0, a, A) = \{(q_0, AA)\}$$

$$\delta(q_0, b, A) = \{(q_1, \varepsilon)\}$$

$$\delta(q_1, b, A) = \{(q_1, \varepsilon)\}$$

$$\delta(q_0, \varepsilon, Z) = \{(q_0, \varepsilon)\}$$

$$\delta(q_1, \varepsilon, Z) = \{(q_1, \varepsilon)\} \quad \dots \text{ si } \Phi \text{ in celelalte cazuri}$$

Reprezentare tabelara

Exemplu:

		a	b	ε	
q_0	Z	(q_0, AZ)		(q_0, ε)	1
	A	(q_0, AA)	(q_1, ε)		
q_1	Z			(q_1, ε)	0
	A		(q_1, ε)		

Care este limbajul acceptat dupa criteriul stivei vide?

Dar dupa criteriul starii finale?

Dar daca starea finala ar fi q_1 ?

Configuratie

- **formal:**

$$(q, x, \alpha) \in Q \times \Sigma^* \times \Gamma^*$$

- automatul se găsește în starea q , pe banda de intrare urmează să se citească (accepte) secvența x , iar în memoria stivă avem secvența α
- configuratie initiala

$$(q_0, w, Z_0)$$

Tranzitii

- \vdash *tranziție directă*

$$(q, aw, Z\alpha) \vdash (p, w, \gamma\alpha) \Leftrightarrow \delta(q, a, Z) \ni (p, \gamma)$$

sau

$$(q, aw, Z\alpha) \vdash (p, aw, \gamma\alpha) \Leftrightarrow \delta(q, \varepsilon, Z) \ni (p, \gamma) \quad \text{(\u03b5-tranziție)}$$

– unde $p, q \in Q$, $a \in \Sigma$, $Z \in \Gamma$, $w \in \Sigma^*$, α, γ din Γ^*

- \vdash^k — k *tranziția* (k tranziții directe) \sim AF
- \vdash^+ — $+$ *tranziția* \sim AF
- \vdash^* — $*$ *tranziția* \sim AF

Secvența acceptată de automat

- după criteriul stivei vide

$$L_{\varepsilon}(M) = \{ w \mid w \in \Sigma^*, (q_o, w, Z_o) \vdash^* (q, \varepsilon, \varepsilon), q \in Q \}$$

– $(q, \varepsilon, \varepsilon)$ - configurația finală după criteriul stivei vide

- după criteriul stării finale

$$L_f(M) = \{ w \mid w \in \Sigma^*, (q_o, w, Z_o) \vdash^* (q, \varepsilon, \gamma), q \in F, \gamma \in \Gamma^* \}$$

– (q, ε, γ) , $q \in F$ configurație finală după criteriul stării finale

$$L_{\varepsilon}(M) = \{ a^n b^n \mid n \in \mathbb{N} \}$$

		a	b	ε	
q ₀	Z	(q ₀ , AZ)		(q ₀ , ε)	1
	A	(q ₀ , AA)		(q ₁ , A)	
q ₁	Z			(q ₁ , ε)	0
	A		(q ₁ , ε)		

De ce? (justificare:)

in q₀ – se accepta oricate simboluri a

cu ramanere in q₀ si adaugare cate un A in stiva

adica: la fiecare citire de a adaug in stiva un A (1)

sau – se trece in starea q₁ (dupa ce am citit cel putin un a, adica am A in stiva)

obs.: se poate trece in q₁ oricand, fara modificarea stivei (2)

sau - se scoate Z din stiva

(acest lucru se poate intampla numai inainte de citirea unui simb)

=> **se accepta secventa vida**

in q₁ – cand in varful stivei este un A, se citeste un b

adica: fiecare citire de b scoate un A din stiva (3)

sau: daca in varful stivei este un Z, acesta se scoate (goleste stiva)

din (1) , (2) , (3) => nr(a) = nr(b)

(4) q₀ citeste a (oricati)

(5) q₁ citeste numai b; nu se poate trece inapoi in q₀

din (2)/(4) , (5) => simb. a citite inaintea simb b

Teoreme de echivalenta

Teoremă.

Fie automatul push-down \mathbf{M} . Există întotdeauna un automat push-down \mathbf{M}' astfel încât $L_{\varepsilon}(\mathbf{M}') = L_f(\mathbf{M})$; si reciproc.

Teoremă.

Oricare ar fi G – o gramatica independenta de context, există un automat push-down \mathbf{M} astfel încât $L_{\varepsilon}(\mathbf{M}) = L(G)$; si reciproc.

G.I.C. \Rightarrow APD echivalent

Fie: $G = (N, \Sigma, P, S)$ – gram. independenta de context

Cine este M - APD astfel incat $L(G) = L_{\varepsilon}(M)$?

constructia:

$$M = (\{q\}, \Sigma, N \cup \Sigma, \delta, q, S, \Phi)$$

1. dacă $(A \rightarrow \alpha) \in P$ atunci $(q, \alpha) \in \delta(q, \varepsilon, A)$;
2. $\delta(q, a, a) = \{(q, \varepsilon)\} \quad \forall a \in \Sigma$;
3. $\delta(.,.,.) = \Phi$ în celelalte cazuri.

Determinism

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

este *determinist* ddacă:

$$\forall Z \in \Gamma, \forall q \in Q, \forall a \in \Sigma$$

$$1) |\delta(q, \epsilon, Z)| = 0 \text{ si } |\delta(q, a, Z)| \leq 1$$

$$2) |\delta(q, \epsilon, Z)| = 1 \text{ si } |\delta(q, a, Z)| = 0$$

in caz contrar, automatul nu este determinist

- multimea limbajelor acceptate de APD nedeterministe este strict mai larga decat multimea limbajelor acceptate de APD deterministe