AGH

AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE

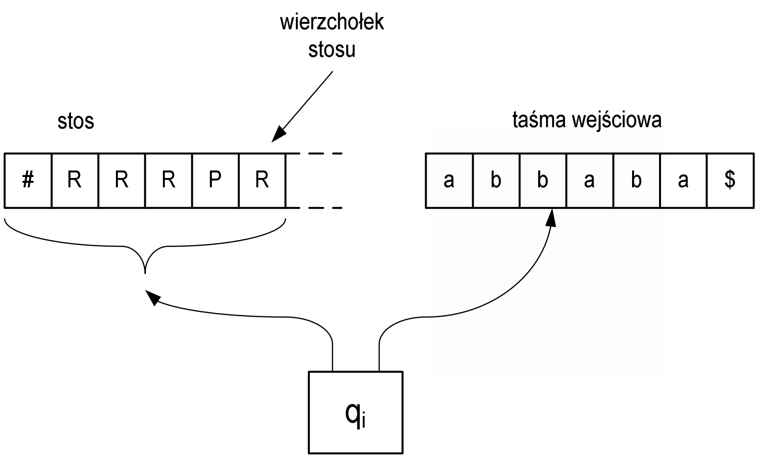
Automat ze stosem

Teoria automatów i języków formalnych

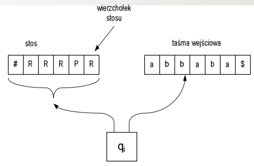
Dr inż. Janusz Majewski
Katedra Informatyki

AGH

Automat ze stosem (1)



Automat ze stosem (2)



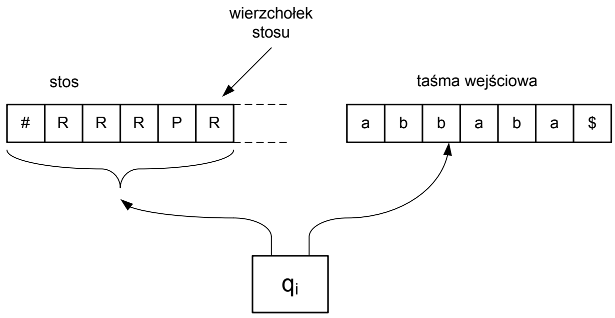
A = < Σ, Q, F, q_0 , Γ, Z_0 , δ, $ >

- Σ – zbiór symboli terminalnych
- Q – zbiór stanów $\#Q < \infty$
- $F \subseteq Q$ – zbiór stanów końcowych
- $q_0 \in Q$ – stan początkowy
- Γ – zbiór symboli stosowych
- $Z_0 \in \Gamma \cup \{\varepsilon\}$ – symbol początkowy stosu
- δ - funkcja przejścia

$$\delta : Q \times (\Sigma \cup \{\varepsilon, \$\}) \times \Gamma^* \mapsto 2^{Q \times \Gamma^*}$$

$ - ogranicznik końca słowa wejściowego

Konfiguracja automatu (opis chwilowy)

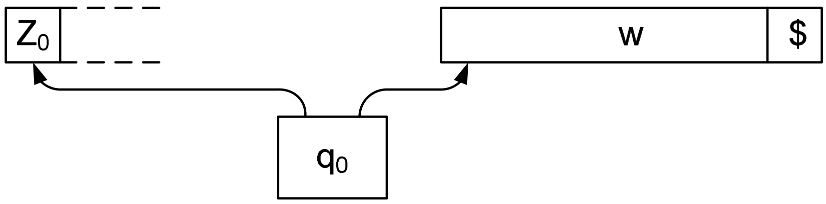


Konfiguracja automatu: (#RRRPR, q_i , baba$)

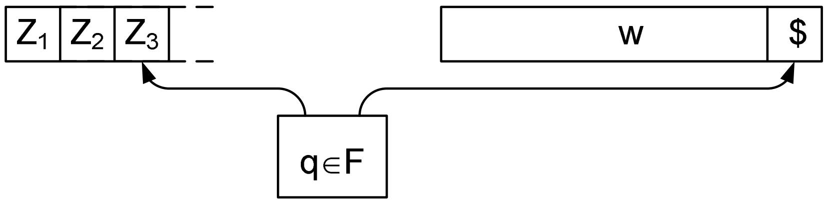
stos (wierzchołek stosu) stan nieprzeczytana część taśmy wejściowej

Akceptacja przez stan końcowy

Konfiguracja początkowa:

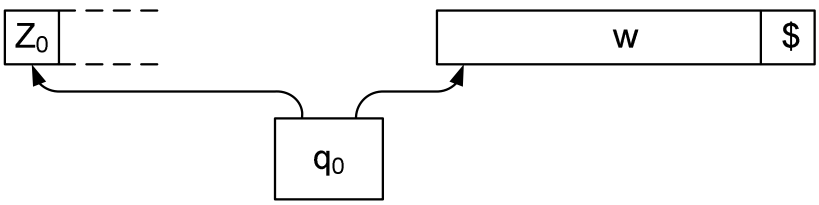


Konfiguracja akceptująca – akceptacja przez stan końcowy

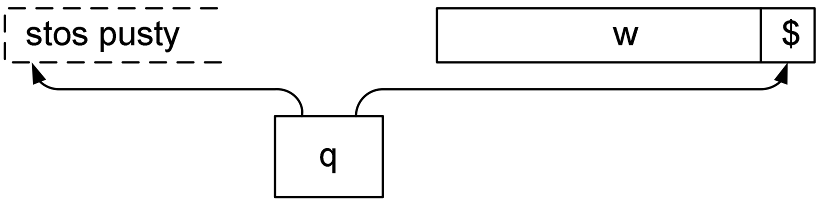


Akceptacja przez pusty stos

Konfiguracja początkowa:



Konfiguracja akceptująca – akceptacja przez pusty stos



Wyprowadzenie bezpośrednie - pojedynczy krok automatu

Wyprowadzenie bezpośrednie

$(\partial, q, a\tau\$) \vdash_A (\partial', q', \tau\$)$

$\partial, \partial' \in \Gamma^*$

$q, q' \in Q$

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

$\tau \in \Sigma^*$

to są początkowe symbole stosu, stos przyrasta "w prawo"

$(q', \text{SUFFIX}(\partial')) \in \delta(q, b, \text{SUFFIX}(\partial))$

gdzie:

$\text{SUFFIX}(\gamma)$ – przyrostek łańcucha γ

$$b = \begin{cases} a & -\text{gdy } a \in \Sigma \\ \varepsilon & -\text{gdy } a = \varepsilon \\ \$ & -\text{gdy } a = \varepsilon, \tau = \varepsilon \end{cases}$$

$\rightarrow$ wtedy wejście jest czytane, głowica przesuwa się w prawo

$\rightarrow$ wtedy wejście nie jest brane pod uwagę

$\rightarrow$ oznacza całkowite przeczytanie wejścia

Przykład: Automat ze stosem akceptujący język $L = \{ 0^n 1^n \mid n = 0,1,\ldots\}$

$A = < \Sigma, Q, F, q_0, \Gamma, Z_0, \delta, \$ >$

$\Sigma = \{ 0, 1\}$

$Q = \{ q_0, q_1, q_2\}$

$F = \{q_0\}$

q_0 – stan początkowy

$\Gamma = \{R, 0 \}$

$Z_0 = R$

(1) $\delta (q_0, 0, R\} =\{(q_1, R0)\}$

(2) $\delta (q_1, 0, 0\} =\{(q_1, 00)\}$

(3) $\delta (q_1, 1, 0\} =\{(q_2, \varepsilon)\}$

(4) $\delta (q_2, 1, 0\} =\{(q_2, \varepsilon)\}$

(5) $\delta (q_2, \$, R\} =\{(q_0, \varepsilon)\}$

(6) $\delta (q_0, \$, R\} =\{(q_0, \varepsilon)\}$

Analizowane słowo: 000111$

$(R, q_0, 000111\$)$

$\vdash^{(1)} (R0, q_1, 00111\$)$

$\vdash^{(2)} (R00, q_1, 0111\$)$

$\vdash^{(2)} (R000, q_1, 111\$)$

$\vdash^{(3)} (R00, q_2, 11\$)$

$\vdash^{(4)} (R0, q_2, 1\$)$

$\vdash^{(4)} (R, q_2, \$)$

$\vdash^{(5)} (\varepsilon, q_0, \$)$

Akceptacja języka przez automat ze stosem

$x \in \Sigma^*$ jest słowem akceptowanym przez automat A (ze stosem) przy stanie końcowym $\Leftrightarrow$

$$(\exists q \in F) \; ((Z_0, q_0, x\$) \vdash_A^* \; (s, q, \$); s \in \Gamma^*)$$

Język L jest akceptowany przez automat A przy stanie końcowym (co oznaczamy L(A)) $\Leftrightarrow$

$$L = L(A) = \{x \in \Sigma^* \mid x \text{ jest akceptowane przez A przy stanie końcowym}\}$$

$x \in \Sigma^*$ jest słowem akceptowanym przez automat A (ze stosem) przy pustym stosie $\Leftrightarrow$

$$(\exists q \in Q) \; ((Z_0, q_0, x\$) \vdash_A^* (\varepsilon, q, \$))$$

Język L jest akceptowany przez automat A przy pustym stosie (co oznaczamy N(A)) $\Leftrightarrow$

$$L = N(A) = \{x \in \Sigma^* \mid x \text{ jest akceptowane przez A przy pustym stosie}\}$$

Języki akceptowane przez automaty ze stosem

Poniższe trzy klasy języków pokrywają się ze sobą:

- Języki bezkontekstowe, czyli języki generowane przez gramatyki bezkontekstowe
- Języki akceptowane przez automaty ze stosem przy stanie końcowym
- Języki akceptowane przez automaty ze stosem przy pustym stosie

Konstrukcja automatu ze stosem odtwarzającego wywód lewostronny w gramatyce $G \in \mathcal{G}_{BK}$ (top - down)

We: $G = < V, \Sigma, P, S > \in \mathcal{G}_{BK}$

Wy: $A = < \Sigma, Q, F, q_0, \Gamma, Z_0, \delta, \$ >$

taki, że **$N(A) = L(G)$**

Rozwiązanie:

$Q := \{q\};$

$F := \varnothing;$

$q_0 := q;$

$\Gamma := V \cap \Sigma;$

$Z_0 := S;$

Przykład dla gramatyki wyrażeń

Przykład:

G = <{E, T, F}, {id, +, *, (,)},

P = { E → E+T | T

T → T*F | F

F → (E) | id}, E >

A = <{id, +, *, (,)}, {q}, ∅, q, {E, T, F, id, +, *, (,)}, E, δ, $>

$\delta(q, \varepsilon, E) = \{_{(1)}(q, T+E), _{(2)}(q, T)\}$

$\delta(q, \varepsilon, T) = \{_{(3)}(q, F*T), _{(4)}(q, F)\}$

$\delta(q, \varepsilon, F) = \{_{(5)}(q,)E( ), _{(6)}(q, \underline{id})\}$

$\delta(q, b, b) = \{_{(pop)}(q, \varepsilon)\}$ dla wszystkich $b \in \{ \underline{id}, +, *, (,) \}$

Analiza słowa: „ id + id * id”

E,	q,	id + id * id $	⊢ (1)	E
T+E,	q,	id + id * id $	⊢ (2)	E + T
T+T,	q,	id + id * id $	⊢ (4)	T + T
T+F,	q,	id + id * id $	⊢ (6)	F + T
T+ id ,	q,	id + id * id $	⊢ (pop)	id + T
T+ ,	q,	+ id * id $	⊢ (pop)	id + T
T ,	q,	id * id $	⊢ (3)	id + T
F * T,	q,	id * id $	⊢ (4)	id + T * F
F * F ,	q,	id * id $	⊢ (6)	id + F * F
F * id,	q,	id * id $	⊢ (pop)	id + id * F
F * ,	q,	* id $	⊢ (pop)	id + id * F
F ,	q,	id $	⊢ (6)	id + id * F
id,	q,	id $	⊢ (pop)	id + id * id
ε,	q,	$		id + id * id

Wyprowadzenie lewostronne top-down

Akceptacja przez pusty stos

Symbole zdjęte przez (pop)

Konstrukcja automatu ze stosem odtwarzającego wywód prawostronny (bottom-up) w gramatyce $G \in \mathcal{G}_{BK}$

We: $G = < V, \Sigma, P, S > \in \mathcal{G}_{BK}$

Wy: $A = < \Sigma, Q, F, q_0, \Gamma, Z_0, \delta, \$ >$

taki, że $L(A) = L(G)$

Rozwiązanie:

$Q := \{q_0, q_1\};$

$F := \{q_1\};$

$\Gamma := V \cup \Sigma \cup \{\#\};$ /* # - dodatkowy symbol */

$Z_0 := \#;$

for $a \in \Sigma$ do

Przykład gramatyki wyrażeń

$G = \langle\{E, T, F\}, \{\underline{id}, +, *, (\ ,)\},$
$\{ E \to E+T \mid T$
$T \to T^* F \mid F$
$F \to (E) \mid \underline{id} \},\qquad E \rangle$

$A = \langle\{ \underline{id}, +, *, (\ ,)\}, \{q_0, q_1\}, \{q_1\}, q_0, \{E, T, F, \underline{id}, +, *, (\ ,), \#\}, \#, \$\rangle$

(shift) $\delta(q_0, b, \varepsilon\} = \{(q_0, b)\}$ dla wszystkich $b \in \{ \underline{id}, +, *, (\ ,)\}$

(1) $\delta(q_0, \varepsilon, E + T\} = \{(q_0, E)\}$

(2) $\delta(q_0, \varepsilon, T \} = \{(q_0, E)\}$

(3) $\delta(q_0, \varepsilon, T^*F \} = \{(q_0, T)\}$

(4) $\delta(q_0, \varepsilon, F\} = \{(q_0, T)\}$

(5) $\delta(q_0, \varepsilon, (E)\} = \{(q_0, F)\}$

(6) $\delta(q_0, \varepsilon, \underline{id} \} = \{(q_0, F)\}$

(acc) $\delta(q_0, \$, \#E \} = \{(q_1, \varepsilon)\}$

Analiza słowa: „ id + id * id”

#,	q_0,	id + id * id $	$\vdash$(shift)	id + id * id
# id,	q_0,	+ id * id $	$\vdash$(6)	
#F,	q_0,	+ id * id $	$\vdash$(4)	F + id * id
#T,	q_0,	+ id * id $	$\vdash$(2)	T + id * id
#E,	q_0,	+ id * id $	$\vdash$(shift)	E + id * id
#E +,	q_0,	id * id $	$\vdash$(shift)	
#E + id,	q_0,	* id $	$\vdash$(6)	
#E + F,	q_0,	* id $	$\vdash$(4)	E + F * id
#E + T,	q_0,	* id $	$\vdash$(shift)	E + T * id
#E + T*,	q_0,	id $	$\vdash$(shift))	
#E + T* id,	q_0,	$	$\vdash$(6)	
#E + T *F,	q_0,	$	$\vdash$(3)	E + T * F
#E + T ,	q_0,	$	$\vdash$(1)	E + T
#E ,	q_0,	$	$\vdash$(acc)	E
ε,	q_1,	$		

Wyprowadzenie prawostronne bottom-up.
Wierzchołek stosu przy redukcjach
to prawa granica osnowy

Deterministyczny automat ze stosem

A = < Σ, Q, F, q_0, Γ, Z_0, δ, $ > - automat ze stosem jest deterministyczny ⇔

(i) $(\forall q \in Q)\ (\forall a \in \Sigma \cup \{\varepsilon, \$\})\ (\forall \gamma \in \Gamma^*)\ (\#\delta(q, a, \gamma) \leq 1)$

(ii) $(\delta(q, a, \alpha) \neq \emptyset \ \wedge\ \delta(q, a, \beta) \neq \emptyset \ \wedge\ \alpha \neq \beta) \Rightarrow$ żaden z łańcuchów: α oraz β nie jest przyrostkiem drugiego łańcucha

(iii) $(\delta(q, a, \alpha) \neq \emptyset \ \wedge\ \delta(q, \varepsilon, \beta) \neq \emptyset) \Rightarrow$ żaden z łańcuchów: α oraz β nie jest przyrostkiem drugiego łańcucha

Twierdzenie: Klasa języków akceptowanych przez deterministyczne automaty ze stosem jest właściwą podklasą klasy języków akceptowanych przez automaty ze stosem.

Innymi słowy: nie dla każdego automatu ze stosem istnieje równoważny mu deterministyczny automat ze stosem.

$$L(A_{\text{Deterministyczny ze Stosem}}) \subset L(A_{\text{ze Stosem}})$$

Przykład

L={xx^R | x∈Σ*} – jest językiem nieakceptowalnym przez deterministyczny automat ze stosem.

Przypuśćmy, że A jest automatem ze stosem akceptującym język *L* i niech *y*∈*Σ** będzie dowolnym słowem przeznaczonym do analizy przez automat. Aby sprawdzić, czy *y* jest postaci xx^R trzeba przepisać lewą połowę słowa y na stos, tzn. przejść od konfiguracji (ε, q_0, $xx^R\$$) do konfiguracji (x, q, $x^R\$$), a następnie przystąpić do sprawdzenia, czy słowo na stosie jest zwierciadlanym odbiciem słowa pozostającego na wejściu. Takie postępowanie wymaga umiejętności odszukiwania połowy (środka) słowa *y*, co przy jednokrotnym jego czytaniu jest oczywiście niemożliwe.

Można pokazać, że deterministyczny automat akceptujący język *L* istnieje, nie jest to już jednak automat ze stosem, ale automat liniowo ograniczony.