



AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE

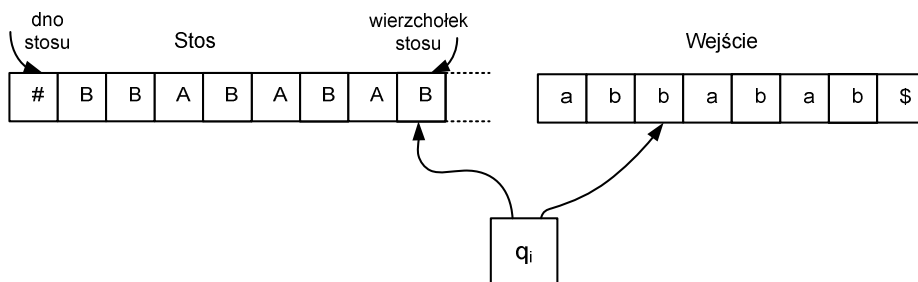
Automat ze stosem

Języki formalne i automaty

Dr inż. Janusz Majewski
Katedra Informatyki

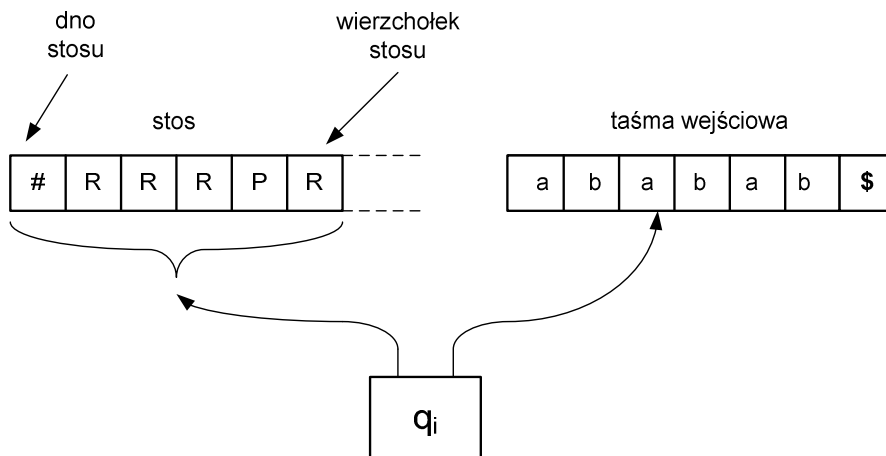


Automat ze stosem (1)

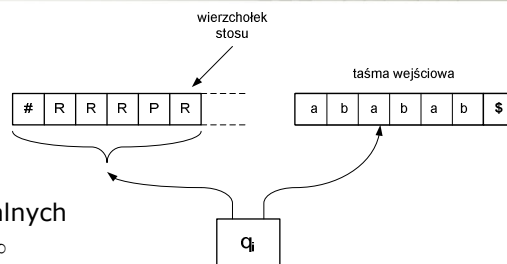




Automat ze stosem (1) c.d.



Automat ze stosem (2)



$A = \langle \Sigma, Q, F, q_0, \Gamma, Z_0, \delta, \$ \rangle$

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

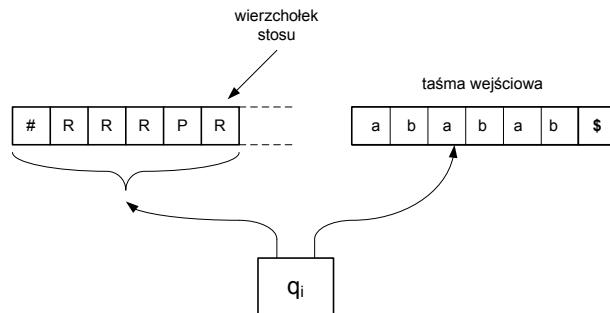
δ - funkcja przejścia

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

$\$$ - ogranicznik końca słowa wejściowego



Konfiguracja automatu (opis chwilowy)



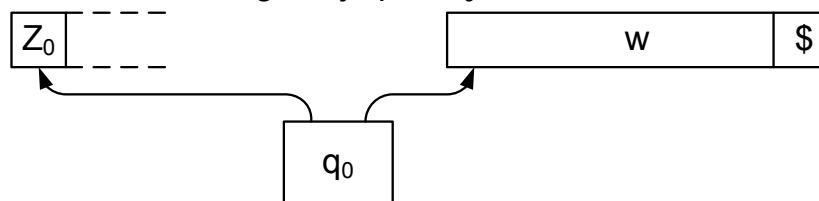
Konfiguracja automatu: $(\#RRRPR, q_i, abab\$)$

stos (wierzchołek stosu) stan nieprzeczytana część taśmy wejściowej

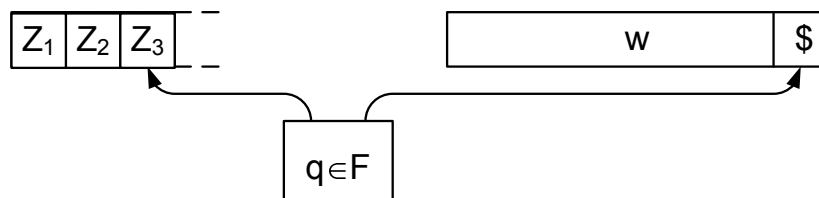


Akceptacja przez stan końcowy

Konfiguracja początkowa:

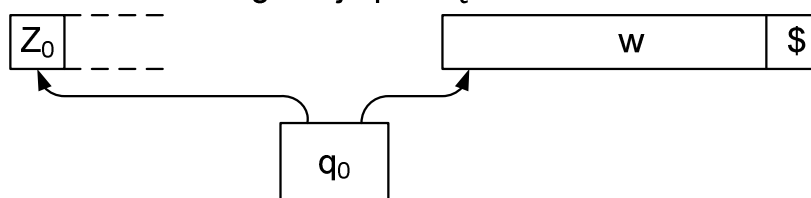


Konfiguracja akceptująca – akceptacja przez stan końcowy

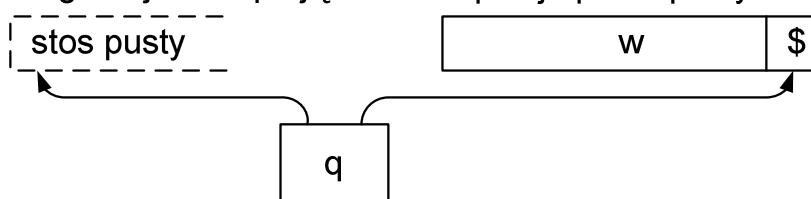


Akceptacja przez pusty stos

Konfiguracja początkowa:



Konfiguracja akceptująca – akceptacja przez pusty stos



Pojedynczy krok automatu

Funkcja przejścia:

$$\delta(\text{aktualny_stan}, \text{wejście}, \text{aktualny_wierzchołek_stosu}) = (\text{nowy_stan}, \text{nowy_wierzchołek_stosu})$$

Kroki automatu:

$\delta(q_n, a, XYZ) = (q_k, ST)$ — zmiana stanu z q_n na q_k , przeczytanie a z wejścia z przesunięciem głowicy w prawo, podmiana XYZ na ST na wierzchołku stosu

$\delta(q_n, a, XYZ) = (q_k, \epsilon)$ — zmiana stanu z q_n na q_k , przeczytanie a z wejścia z przesunięciem głowicy w prawo, zdjęcie XYZ z wierzchołka stosu

$\delta(q_n, a, \epsilon) = (q_k, ST)$ — zmiana stanu z q_n na q_k , przeczytanie a z wejścia z przesunięciem głowicy w prawo, dopisanie ST na wierzchołku stosu

$\delta(q_n, \epsilon, XYZ) = (q_k, ST)$ — zmiana stanu z q_n na q_k , podmiana XYZ na ST na wierzchołku stosu bez czytania wejścia i bez przesuwania głowicy wejściowej

$\delta(q_n, \epsilon, XYZ) = (q_k, \epsilon)$ — zmiana stanu z q_n na q_k , zdjęcie XYZ z wierzchołka stosu bez czytania wejścia i bez przesuwania głowicy wejściowej

$\delta(q_n, \epsilon, \epsilon) = (q_k, ST)$ — zmiana stanu z q_n na q_k , dopisanie ST na wierzchołku stosu bez czytania wejścia i bez przesuwania głowicy wejściowej



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

$A = \langle \Sigma, Q, F, q_0, \Gamma, Z_0, \delta, \$ \rangle$

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

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

$F = \{ q_0 \}$

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\$)

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

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

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

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

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

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

$\mapsto^{(5)}$ (ε , q_0 , \$)



Przykład konstrukcji automatu ze stosem (1)

Budujemy automat ze stosem akceptujący wszystkie łańcuchy zbudowane z liter **a** oraz **b**, w których liczba liter **a** jest równa liczbie liter **b**.

Początkowo na stosie umieścimy symbol **#**.

Jeżeli na wierzchołku stosu nie ma żadnej z liter **a** lub **b**, to przeczytaną z wejścia literę umieścimy na stosie:

$\delta(q_0, a, \#) = \{(q_0, \#a)\}$

$\delta(q_0, b, \#) = \{(q_0, \#b)\}$

Jeżeli kolejna czytana litera jest identyczna, jak ta znajdująca się na wierzchołku stosu, to dokładamy ją na stos:

$\delta(q_0, a, a) = \{(q_0, aa)\}$

$\delta(q_0, b, b) = \{(q_0, bb)\}$



Przykład konstrukcji automatu ze stosem (2)

Jeżeli kolejna czytana litera jest różna od tej znajdującej się na wierzchołku stosu, to zdejmujemy literę z wierzchołka stosu:

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

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

Jeżeli widzimy na wejściu znacznik końca \$ i na wierzchołku stosu jest #, to akceptujemy:

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

(Stan q_1 będzie stanem akceptującym).



Przykład konstrukcji automatu ze stosem (3)

Pełna definicja skonstruowanego automatu będzie następująca:

$$A = \langle \Sigma, Q, F, q_0, \Gamma, Z_0, \delta, \$ \rangle$$

$$\Sigma = \{a, b\}$$

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

$$F = \{q_1\}$$

q_0 – stan początkowy

$$\Gamma = \{\#, a, b\}$$

$$Z_0 = \#$$

$$\delta(q_0, a, \#) = \{(q_0, \#a)\}$$

$$\delta(q_0, b, \#) = \{(q_0, \#b)\}$$

$$\delta(q_0, a, a) = \{(q_0, aa)\}$$

$$\delta(q_0, b, b) = \{(q_0, bb)\}$$

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

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

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



Akceptacja języka przez automat ze stosem

$x \in \Sigma^*$ jest słowem akceptowanym przez automat A (ze stosem) przy stanie końcowym jeżeli **istnieje** taka sekwencja kroków automatu, która powoduje przejście automatu z konfiguracji początkowej do konfiguracji akceptującej przez stan końcowy.

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 jeżeli **istnieje** taka sekwencja kroków automatu, która powoduje przejście automatu z konfiguracji początkowej do konfiguracji akceptującej przez pusty stos.

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, których lewe strony produkcji są pojedynczymi symbolami nieterminalnymi
- Języki akceptowane przez automaty ze stosem przy stanie końcowym
- Języki akceptowane przez automaty ze stosem przy pustym stosie



Przykład automatu ze stosem odtwarzającego wywód lewostronny metodą top-down dla gramatyki wyrażeń

Przykład:

$G = \langle \{E, T, F\}, \{\underline{id}, +, *, (,)\},$

$P = \{ E \rightarrow E+T \mid T$

$T \rightarrow T*F \mid F$

$F \rightarrow (E) \mid \underline{id}\rangle, E >$

$A = \langle \{\underline{id}, +, *, (,)\}, \{q\}, \emptyset, q, \{E, T, F, \underline{id}, +, *, (,)\}, E, \delta, \$ \rangle$

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

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

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

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



Analiza słowa: „ id + id * id ”

E, q, <u>id</u> + <u>id</u> * <u>id</u> \$	↳ (1)	E	
T+E, q, <u>id</u> + <u>id</u> * <u>id</u> \$	↳ (2)	E + T	
T+T, q, <u>id</u> + <u>id</u> * <u>id</u> \$	↳ (4)	T + T	
T+F, q, <u>id</u> + <u>id</u> * <u>id</u> \$	↳ (6)	F + T	
T+ <u>id</u> , q, <u>id</u> + <u>id</u> * <u>id</u> \$	↳ (pop)	<u>id</u> + T	
T+, q, + <u>id</u> * <u>id</u> \$	↳ (pop)	<u>id</u> + T	
T, q, <u>id</u> * <u>id</u> \$	↳ (3)	<u>id</u> + T	
F * T, q, <u>id</u> * <u>id</u> \$	↳ (4)	<u>id</u> + T * F	
F * F, q, <u>id</u> * <u>id</u> \$	↳ (6)	<u>id</u> + F * F	
F * <u>id</u> , q, <u>id</u> * <u>id</u> \$	↳ (pop)	<u>id</u> + <u>id</u> * F	
F *, q, * <u>id</u> \$	↳ (pop)	<u>id</u> + <u>id</u> * F	
F, q, <u>id</u> \$	↳ (6)	<u>id</u> + <u>id</u> * F	
<u>id</u> , q, <u>id</u> \$	↳ (pop)	<u>id</u> + <u>id</u> * <u>id</u>	
ε, q, \$		<u>id</u> + <u>id</u> * <u>id</u>	

Wyprowadzenie lewostronne top-down

Akceptacja przez pusty stos

Symbole zdjęte przez (pop)



Przykład automatu ze stosem odtwarzającego wywód prawostronny metodą bottom-up dla gramatyki wyrażeń

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

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

(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”

#,	$q_0,$	<u>id</u> + <u>id</u> * <u>id</u> \$	\hookrightarrow (shift)	<u>id</u> + <u>id</u> * <u>id</u>
# <u>id</u> ,	$q_0,$	+ <u>id</u> * <u>id</u> \$	\hookrightarrow (6)	
#_F,	$q_0,$	+ <u>id</u> * <u>id</u> \$	\hookrightarrow (4)	F + <u>id</u> * <u>id</u>
#T,	$q_0,$	+ <u>id</u> * <u>id</u> \$	\hookrightarrow (2)	T + <u>id</u> * <u>id</u>
#E,	$q_0,$	+ <u>id</u> * <u>id</u> \$	\hookrightarrow (shift)	E + <u>id</u> * <u>id</u>
#E +,	$q_0,$	<u>id</u> * <u>id</u> \$	\hookrightarrow (shift)	
#E + <u>id</u> ,	$q_0,$	* <u>id</u> \$	\hookrightarrow (6)	
#E + F,	$q_0,$	* <u>id</u> \$	\hookrightarrow (4)	E + F * <u>id</u>
#E + T,	$q_0,$	* <u>id</u> \$	\hookrightarrow (shift)	E + T * <u>id</u>
#E + T*,	$q_0,$	<u>id</u> \$	\hookrightarrow (shift)	
#E + T* <u>id</u> ,	$q_0,$	\$	\hookrightarrow (6)	
#E + T*F,	$q_0,$	\$	\hookrightarrow (3)	E + T * F
#E + T,	$q_0,$	\$	\hookrightarrow (1)	E + T
#E,	$q_0,$	\$	\hookrightarrow (acc)	E
$\varepsilon,$	$q_1,$	\$		

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



Deterministyczny automat ze stosem

Automat ze stosem jest deterministyczny wtedy i tylko wtedy, gdy w każdej konfiguracji może wykonać co najwyżej jeden ruch.

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

$L = \{xx^R \mid x \in \Sigma^*\}$ – jest językiem nieakceptowalnym przez deterministyczny automat ze stosem.

Przypuśćmy, że A jest automatem ze stosem akceptującym język L i niech $y \in \Sigma^*$ 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 $(\varepsilon, q_0, xx^R\$)$ 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 (środką) 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 tzw. automat liniowo ograniczony lub maszyna Turinga.