Teoria Automatów i Języków Formalnych

Ćwiczenia 11: Automat ze stosem

dr inż. Marcin Luckner
mluckner@mini.pw.edu.pl

Wydział Matematyki i Nauk Informacyjnych

Wersja 1.3
3 marca 2021

Fundusze Europejskie Wiedza Edukacja Rozwój
Rzeczpospolita Polska
Politechnika Warszawska
Unia Europejska Europejski Fundusz Społeczny

Projekt „NERW 2 PW. Nauka – Edukacja – Rozwój – Współpraca" współfinansowany jest ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego.

Zadanie 10 pn. „Modyfikacja programów studiów na kierunkach prowadzonych przez Wydział Matematyki i Nauk Informacyjnych", realizowane w ramach projektu „NERW 2 PW. Nauka – Edukacja – Rozwój – Współpraca", współfinansowanego jest ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego.

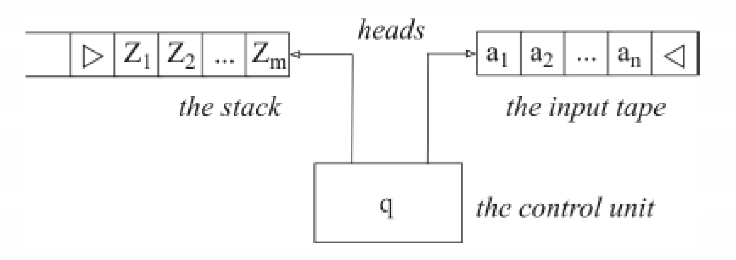
Fundusze Europejskie Wiedza Edukacja Rozwój

Rzeczpospolita Polska

Politechnika Warszawska

Unia Europejska Europejski Fundusz Społeczny

Automat ze stosem



Rysunek 1: Automat ze stosem

Definicja

$$A = (Q, \Sigma, \Gamma, \delta, q0, \triangleright, \triangleleft, F)$$

Automat składa się z następujących elementów:

- Q skończony zbiór stanów
- Σ zbiór symboli wejściowych,
- Γ alfabet stosu, zbiór symboli, które można umieścić na stosie,
- δ funkcja przejścia z $Q \times \Sigma \times \Gamma$ w $Q \times \Gamma$,
- q_0 stan początkowy,
- $\triangleright$ symbol początkowy, początkowo stos zawiera tylko ten symbol,
- $\triangleleft$ symbol końca wejścia,
- F zbiór stanów końcowych $F \subseteq Q$.

Operacje

polecenie	argument	stos przed	stos po
stay	A	$\triangleright A$	$\triangleright A$
pop	ϵ	$\triangleright A$	$\triangleright$
push	AB	$\triangleright A$	$\triangleright AB$

Tabela przejścia

- W przeciwieństwie do maszyny Turinga, funkcja przejścia automatu ze stosem ma trzy argumenty.
- Dlatego nie może być reprezentowana przez dwuwymiarową tablicę.
- Tworzymy osobną tablicę dla każdego stanu z wierszami oznaczonymi symbolami stosu i kolumnami oznaczonymi symbolami wejściowymi.
- Wiersze i kolumny zawierają odpowiednio symbole $\triangleright$ i $\triangleleft$.

Akceptacja

- Słowo wejściowe jest akceptowane kiedy automat osiągnie stan finalny.
- Jednakże można też akceptować pustym stosem.
  - Słowo wejściowe jest akceptowane jeżeli osiągnęliśmy symbol końca wejścia i stos jest pusty.
- Akceptacja przez pusty stos i poprzez stan końcowy są sobie równoważne.

Przykład I

Zaprojektuj deterministyczny automat ze stosem nad alfabetem $\Sigma = \{0, 1\}$ słów z równą liczbą zer i jedynek.

Przykład I - idea

- Możemy zliczać symbole poprzez składowanie ich reprezentacji na stosie.
- Jeżeli reprezentant symbolu wejściowego jest na stosie to dokładamy go (push).
- Jeżeli drugi symbol jest reprezentowany na stosie usuwamy go (pop).
- Automat będzie akceptował pustym stosem.

Case I - tabela przejścia

q_0	0	1	$\lhd$
O	(q_0, OO)	(q_0, ϵ)	q_R
I	(q_0, ϵ)	(q_0, II)	q_R
$\rhd$	$(q_0, \rhd O)$	$(q_0, \rhd I)$	q_A

Przykład II

Zaprojektuj deterministyczny automat ze stosem nad alfabetem $\Sigma = \{0, 1\}$ słów z dwukrotnie większą liczbą zer niż jedynek.

Przykład II - idea

- Zadanie wygląda na podobne do poprzedniego, jednak istnieje pewna ważna różnica.
- Odkładając reprezentacje 0 na stos potrzebujemy usuwać dwa symbole używając jednego symbolu 1.
- Nie można tego osiągnąć pojedyńczą akcją pop. Dlatego automat powinien być niedeterministyczny lub podwójne 0 powinno być symulowane w inny sposób.
- W proponowanym rozwiązaniu będzie ono symulowane poprzez stan.

Przykład II - tabele przejścia

q_0	0	1	$\triangleleft$
O	(q_1, OO)	(q_0, ϵ)	q_R
I	(q_1, ϵ)	(q_0, II)	q_R
$\triangleright$	$(q_1, \triangleright O)$	$(q_0, \triangleright I)$	q_A

q_1	0	1	$\triangleleft$
O	(q_0, O)	(q_0, ϵ)	q_R
I	(q_0, I)	(q_1, II)	q_R
$\triangleright$	$(q_0, \triangleright)$	$(q_1, \triangleright I)$	q_R

Case II - Przykład obliczeń

q_0	0	1	$\lhd$
O	(q_1, OO)	(q_0, ϵ)	q_R
I	(q_1, ϵ)	(q_0, II)	q_R
$\rhd$	$(q_1, \rhd O)$	$(q_0, \rhd I)$	q_A

q_1	0	1	$\lhd$
O	(q_0, O)	(q_0, ϵ)	q_R
I	(q_0, I)	(q_1, II)	q_R
$\rhd$	$(q_0, \rhd)$	$(q_1, \rhd I)$	q_R

- Słowo wejściowe: 101000

1. $\mathbf{1}01000\lhd, q_0, \rhd \rightarrow 01000\lhd, q_0, \rhd I$
2. $\mathbf{0}1000\lhd, q_0, \rhd \mathbf{I} \rightarrow 1000\lhd, q_1, \rhd$
3. $\mathbf{1}000\lhd, q_1, \rhd \rightarrow 000\lhd, q_1, \rhd I$
4. $\mathbf{0}00\lhd, q_1, \rhd \mathbf{I} \rightarrow 00\lhd, q_0, \rhd I$
5. $\mathbf{0}0\lhd, q_0, \rhd \mathbf{I} \rightarrow 0\lhd, q_1, \rhd$
6. $\mathbf{0}\lhd, q_1, \rhd \rightarrow 0\lhd, q_0, \rhd$
7. $\lhd, q_0, \rhd \rightarrow q_A$

Zadania I

1. Czy następujące zadanie można rozwiązać używając jednostanowego deterministycznego automatu ze stosem?
   - Zaprojektuj deterministyczny automat ze stosem nad alfabetem $\Sigma = \{0, 1\}$ słów z dwukrotnie większą liczbą zer niż jedynek.
2. Zaprojektuj deterministyczny automat ze stosem dla następujących języków.
   - Język nad alfabetem $\Sigma = \{0, 1\}$ słów z nie większą liczbą jedynek niż zer w każdym prefiksie.
   - Język nad alfabetem $\Sigma = \{0, 1\}$ słów z nieparzystą różnicą wystąpień zer i jedynek.
   - Język nad alfabetem $\Sigma = \{a, b, c\}$ słów postaci $a^i b^j c^k$ gdzie $k < i + j$ i $k > i$.