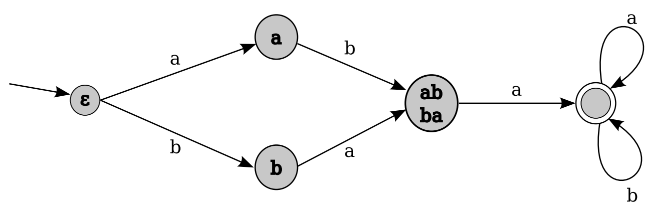
JAO: Języki, automaty i obliczenia, egzamin popr. 11 Września 2014

(każdą odpowiedź należy uzasadnić, nie dotyczy to zadania 1)

1. Narysować **minimalny** automat deterministyczny dla następującego języka, pomijajac stan "śmietnik".

$$(ab \cup ba)a(a \cup b)^*$$

Rozwiązanie Stany automatu przedstawiają klasy abstrakcji kongruencji wyznaczonej przez powyższy język.



2. Dla skończonego automatu deterministycznego A i jego dwóch wyróżnionych stanów p, q niech $\tilde{L}(A)$ będzie zbiorem słów, dla których w obliczeniu automatu stan p występuje tyle samo razy co stan q. Czy $\tilde{L}(A)$ jest regularny dla każdego deterministycznego automatu skończonego A ?

Rozwiązanie Nie. Weźmy automat $A = (\Sigma, Q, \delta, Q_0, F)$, gdzie

- $\Sigma = \{0, 1\}$,
- $Q = \{q_0, q_1\}$,
- $\delta = \{q_0 \xrightarrow{0} q_0, q_0 \xrightarrow{0} q_1, q_1 \xrightarrow{1} q_1\}$,
- $Q_0 = \{q_0\}$,
- $F = \{q_0, q_1\}$.

Rozpoznaje on język 0^*1^*. Jedyna możliwość wyboru pary różnych stanów tutaj to q_0, q_1. Dla takiego wyboru łatwo widzieć, że $\tilde{L}(A) = \{0^i 1^{i-1} \ : \ i > 0\}$. Pokażemy, że nie jest to język regularny, korzystając z lematu o pompowaniu dla języków regularnych. Niech C będzie stałą z tego lematu. Weźmy słowo $0^C 1^{C-1}$. W takim razie fragment, który można pompować, znajduje się w ramach podsłowa 0^C, a zatem można zwiększyć liczbę symboli 0 przy stałej liczbie symboli 1, pozostając w języku, co przeczy podanej postaci języka $\tilde{L}(A)$.

3. Napisać jak najprostszą gramatykę bezkontekstową generującą język

$$\{\, a^i b^{|j-i|} c^j \ : \ i, j \geqslant 0 \,\}.$$

Rozwiązanie Język generowany jest przez gramatykę:

$$S \to aSc, \quad S \to A, \quad S \to C, \quad A \to aAb, \quad A \to \varepsilon, \quad C \to bCc, \quad C \to \varepsilon$$

Weźmy słowo postaci $a^i b^{|j-i|} c^j$, gdzie $i, j \geqslant 0$. Niech $k = \min(i, j)$. Wyprowadzimy to słowo, używając k razy reguły $S \to aSc$, a następnie jeśli $i \geqslant j$, stosując $S \to A$, i dalej $i - j$ razy regułę $A \to aAb$, by zakończyć wyprowadzenie jednym zastosowaniem reguły $A \to \varepsilon$. Jeśli natomiast $j > i$, to korzystamy z $S \to C$, a następnie używamy $j - i$ razy reguły $C \to bCc$, by zakończyć wyprowadzenie regułą $C \to \varepsilon$.

Z kolei wyprowadzenie w powyższej gramatyce można przedstawić jako zastosowanie $k \geqslant 0$ razy reguły $S \to aSc$, po czym zastosowanie reguły (1) $S \to A$ lub (2) $S \to C$. W przypadku (1) dalsze wyprowadzenie składa się z $l \geqslant 0$ użyć reguły $A \to aAb$, po których następuje użycie reguły $A \to \varepsilon$. W wyniku powstaje słowo postaci $a^k a^l b^l c^k$, które należy do zadanego języka. W przypadku (2) dalsze wyprowadzenie składa się z $l \geqslant 0$ użyć reguły $C \to bCc$, po których następuje użycie reguły $C \to \varepsilon$. W wyniku powstaje słowo postaci $a^k b^l c^l c^k$, które również należy do zadanego języka.

4. Czy jest bezkontekstowym język $\{\, a^n b^{n^3} \ : \ n \geqslant 1 \,\}$?

Rozwiązanie Nie, ten język nie jest bezkontekstowy. Skorzystamy z lematu o pompowaniu dla języków bezkontekstowych. Dowód przez sprzeczność. Załóżmy, że ten język, nazwijmy go L, jest bezkontekstowy. Niech C będzie zależną od języka liczbą z lematu o pompowaniu. Rozważmy słowo $w = a^N b^{N^3}$, gdzie $N > C$. Niech $uvwxy$ będzie dekompozycją tego słowa wynikającą z lematu o pompowaniu. Granica między literami a a literami b w słowie w nie może przypadać ani na v, ani na w, bo inaczej wynik pompowania $uvvwxxy$ będzie zawierał więcej niż jedno przejście od liter a do liter b. Jeśli całe podsłowo vwx wypada wyłącznie w ramach a lub wyłącznie w ramach b, to pompowanie będzie zwiększało liczbę wystąpień jednego symbolu, zostawiając tę liczbę dla drugiego bez zmian, co spowoduje popsucie zależności $a^n b^{n^3}$. Pozostaje przypadek, gdy podsłowo v znajduje się w ramach ciągu liter a, zaś podsłowo w w ramach ciągu liter b. Wtedy jednak pompowanie $uv^{k+1}wx^{k+1}y$ da słowo postaci $a^{N+kc_1} b^{N^3+kc_2}$, gdzie $c_1 = |v|$, a $c_2 = |x|$. Zauważmy teraz, że $(N + kc_1)^3 = N^3 + 3N^2kc_1 + 3Nk^2c_1^2 + k^3c_1^3 > N^3 + kc_2$ dla odpowiednio dużego k, ze względu na dominującą wagę członu $k^3c_1^3$ nad członem kc_2.

5. Czy następujący problem jest rozstrzygalny: dla danego niedeterministycznego automatu skończonego A sprawdzić, czy jest nieskończony zbiór $\{\, n > 0 \ : \ L(A) \cap \Sigma^n \ = \ \emptyset \,\}$.

Rozwiązanie Tak. Ten problem jest rozstrzygalny.

Na początku zauważmy, że problem ten jest równoważny szczególnemu przypadkowi, gdy alfabet jest jednoliterowy. Rzeczywiście możemy każdą produkcję postaci $q \xrightarrow{b} q'$, gdzie $b \neq a$ zastąpić przez $q \xrightarrow{a} q'$ i dostać w ten sposób z automatu A nad dowolnym alfabetem automat A' nad $\{a\}$. Oczywiście każdy przebieg A daje w wyniku przebieg A', będący translacją przejść według schematu podanego powyżej. Jednak również każdy przebieg A' da się podnieść do przebiegu A, zastępując przejście $q \xrightarrow{a} q'$ dowolnym jego przejściem źródłowym $q \xrightarrow{b} q'$.

Możemy teraz automat A' nad językiem jednoliterowym zdeterminizować algorytmem determinizacji, otrzymując automat A_d. Graf takiego automatu, ze względu na deterministyczny charakter automatu, ma postać ciągu wierzchołków $q_1, \ldots, q_n$, takich że $q_i \xrightarrow{a} q_{i+1}$ i, jeżeli język jest nieskończony, $q_n \xrightarrow{a} q_l$ dla pewnego $l \in \{1, \ldots, n\}$. Natychmiast widać, że $\{n > 0 \ : \ L(A) \cap \Sigma^n = \emptyset\}$ jest nieskończony wtedy i tylko wtedy, gdy w pętli $q_l, \ldots, q_n$ któryś stan nie jest akceptujący, którą to własność łatwo sprawdzić algorytmicznie.