Rozwiązania niektórych z około dwustu łatwych zadań z języków formalnych i złożoności obliczeniowej (i jednego nie aż tak trudnego, jak się o nim mówi)

Numeracja zadań jak w zbiorze z 2017 roku

Wrocław, 3 kwietnia 2017

1 Wskazówki

Patrząc perspektywicznie w stronę zbliżających się egzaminów, ta sekcja wydaje mi się ważnejsza i zachęcam do spędzania z nią czasu podczas samodzielnych wieczornych rozmyślań.

1.1 Języki i automaty

Zadanie 63. *Warto zastanowić się, jak miałby wyglądać niedeterministyczny automat ze stosem rozpoznający taki język. Pomocny może się też okazać pewien lemat.*

1.1.1 Języki rodzynkowe

Zadanie 71. *Probem sprowadza się do rozpoznawania liter języka L^*. Gdyby miał istanieć taki język L, to pewnie musiałby być nieskończony (dlaczego?). Może istnieje jakaś prosta, nieskończona rodzina słów postaci a^*ba^*, którą automat ze stosem może łatwo rozpoznawać, ale taki bez stosu będzie miał trudniej? Formalnie, jak to bywa, warto używać lematów.*

Zadanie 72. *Wskazówka z treści zadania wydaje się bardzo dobra. Tu cytat: "Warto rozważyć słowa postaci $a^k(ba^{2n})^l$, dla odpowiednich liczb k i l."*

Zadanie 73. *Raczej chodzi o oszacowanie $i(L_nL_n) \geqslant c2^{i(L_n)}$, przy założeniu że $i(L_n) \geqslant n$, inaczej to nie ma sensu. Możnaby najpierw pomyśleć, dlaczego w przypadku NDFA nie można udowodnić takiego oszacowania. A potem zmusić DFA, żeby pamiętał dużo tego, co NDFA pamiętać nie musi, bo może zgadnąć.*

1.2 Obliczalność

Zadanie 92. *Jak coś ma nie być r.e. to na pewno chodzi o redukcję z* $\overline{K}$. *Delikatna modyfikacja 89.*

Zadanie 99. *i) Jak coś ma nie być r.e. to na pewno chodzi o redukcję z* $\overline{K}$. *Delikatna modyfikacja 89.*
ii) Czy $\Phi_m \neq \Phi_n \iff \exists k \Phi_m(k) \neq \Phi_n(k)$*?*

Zadanie 101. *Trzeba wprost zdefiniować redukcję z* B *do* A.

Zadanie 102. *a. założenie o nietrywialności zadania b. implikuje odpowiedź w zadaniu a.. Pewna użyteczna redukcja i w tym zadaniu okaże się pomocna. b. formalne sformułowanie tego, co to znaczy że* $n \in B$*, używająca 3 zmiennych, wprost prowadzi do co-r.e. zbioru* A.

Zadanie 103. *Należałoby się zastanowić, co wspólnego mają te warunki z monotonicznością* f.

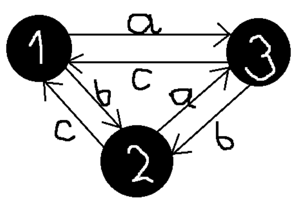
2 Szkice rozwiązań

2.1 Języki i automaty

2.1.1 Synchronizacja automatów częściowych

Zadanie 40.

Odp: NIE.



Rysunek 1: $csync(\{1,2\}) \supset \{a\}$; $csync(\{1,3\}) \supset \{b\}$; $csync(\{1,2\}) \supset \{c\}$. Natomiast $csync(Q) = \emptyset$.

Zadanie 41.

W obydwu podpunktach wystarczy zbadać funkcję

$$F : 2^Q \times \Sigma \longrightarrow 2^Q$$
$$F(S,a) = \{\delta(q,a) : \ q \in S\}.$$

Oczywiście $s \in csync(S) \iff |\widehat{F}(S,s)| = 1$, gdy zdefiniujemy $\widehat{F}$ w naturalny sposób. Ponadto $1 \leqslant |F(A,a)| \leqslant |A|$, zatem $1 \leqslant |\widehat{F}(S,p)| \leqslant |S|$ dla dowolnego prefiksu p słowa s, czyli $\widehat{F}(S,p)$ może przyjmować co najwyżej $\sum_{k=1}^{|S|} \binom{|Q|}{k}$ różnych wartości. Oznaczmy tę liczbę jako M.

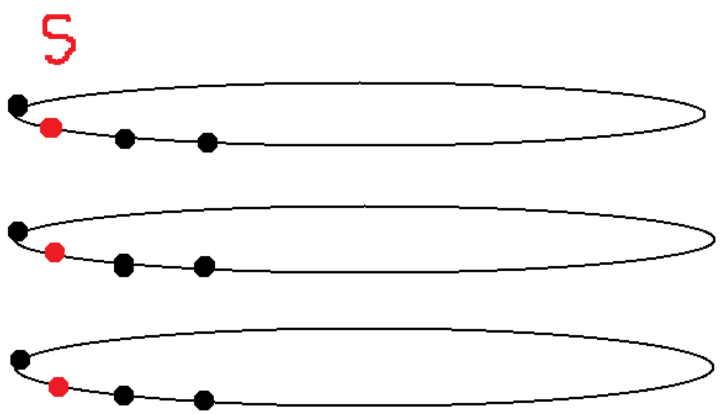
Dla $|s| > M$ istnieją prefiksy p_1 i p_2 słowa $s = p_1 s_1 = p_2 s_2$, $|p_1| < |p_2|$, takie że $\widehat{F}(S,s_1) = \widehat{F}(S,s_2)$ (Zasada Szufladkowa). Wtedy oczywiście $\widehat{F}(S,s) = \widehat{F}(S,p_1 s_2)$, przy czym $|p_1 s_2| < |s|$.

W zwiazku z powyższym $csync(Q) \neq \emptyset \iff \exists s \in S |s| \leqslant M$.
Dokładne odpowiedzi wynikają z tego wprost, po podstawieniu za S a) dowolnego trzyelementowego zbioru stanów b) Q.

Zadanie 42.

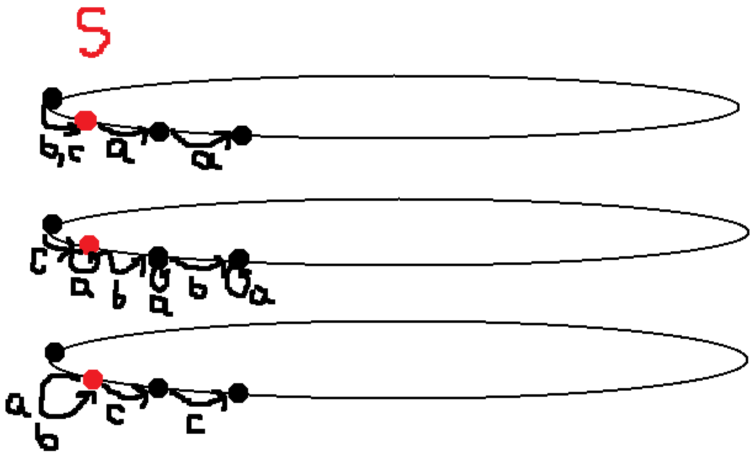
Wystarczy rozwiązać M, L i XL wynikają w prosty sposób. Wskazówka jest myląca.

Zbudujmy automat (częściowy) z trzech cykli, ułożonych jeden nad drugim, każdy długości m. Trzy stany, ułożene jeden nad drugim, będą stanowiły nasz początkowy zbiór S

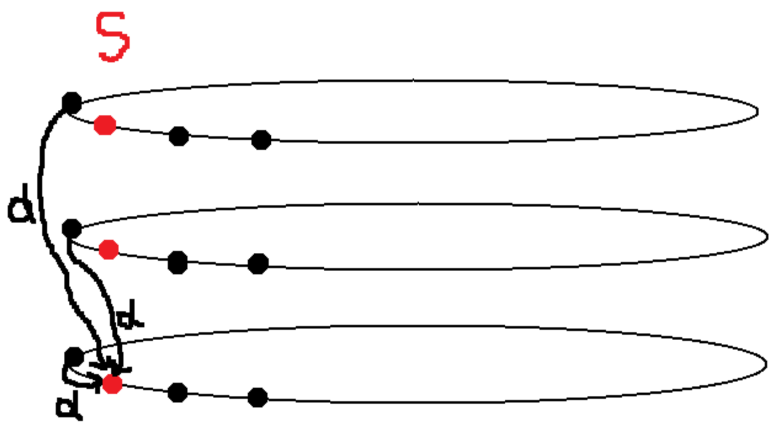


Naszym celem jest, aby co jedną literę zmieniał się stan na górnym cyklu, co m liter stan na drugim, a co m^2 na trzecim. Zapenimy też, że synchro-

nizacja będzie mogła nastąpić dopiero po przejściu przez dolny stan całego cyklu (czyli m^3 krokach).
Możemy to wymusić w następujący sposób:



przy czym pętelki z literą a są przy każdym stanie na drugim dysku, a pętelki z literami a,b są przy każdym stanie na trzecim dysku.
Możliwość synchronizacji zapeniamy przez dodanie przejść z przedostatnich stanów na każdym dysku do pierwszego stanu dolnego dysku. Oznaczenie ich specjalną "literką synchronizacji", jak d, zapewni nam, że skorzystać z niej będzie można dopiero gdy dolny stan dojdzie do przedostatniego miejsca na dolnym dysku (co następuje dopiero po m^3 krokach:



Zauważmy teraz, że ten automat (zanim dojedzie do synchronizacji) jednoznacznie wyznacza słowo, dla którego funkcja przejścia jest określona dla wszystkich stanów z S:

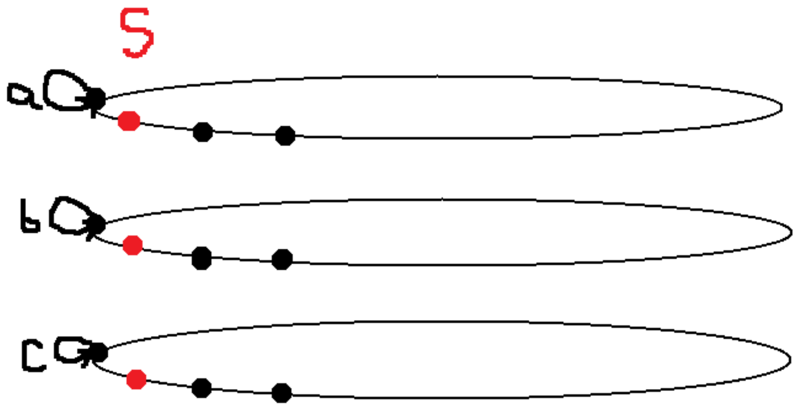
$$s = ((a^{m-1}b)^{m-1}c)^{m-1}d$$

Takie s synchronizuje S i nie istnieje żadne krótsze od niego. Nietrudno wyliczyć, że jest ono odpowiednio długie.

Co do wersji L i XL, wystarczy zauważyć, że:

1. To samo rozumowanie można zastosować dla dowolnie wielu cykli, otrzymując wielomian dowolnie dużego stopnia.
2. Potrzebny nam alfabet, który ma $k+1$ liter (gdzie k jest liczbą cykli). Ale dowolny alfabet można zamienić na dwuliterowy, zapisując jego l-tą literę w postaci 0^l1. To działa.

Dodatkowo, dodając pętle z literką przejścia na przedostatnich stanach każdego cyklu, zamienimy nasz automat na taki, który synchronizuje wszystkie stany (pierwszy stan każdego cyklu "zjada" wszystkie kolejne w pierwszym przejściu cyklu):



2.1.2 Transducery

Zadanie 77.

Podpunkt 1: definiujemy $\sigma_{Mealy} = \sigma_{Moore} \circ \delta$. Reszta zostaje.
Podpunkt 2: definiujemy (dla transducera Mealy'ego $\langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma_{Mealy} \rangle$)

1. $Q' = Q \times \Sigma \cup q_0'$. Stan (q, a) = stan do którego doszlibyśmy w starym automacie ze stanu q wczytując literę a. Stan q_0' - dodatkowy stan początkowy.

2. $\delta'((q,a),b) = (\delta(q,a),b)$ dla pary $(q,a) \in Q \times \Sigma$

   $\delta'(q_0', a) = (q_0, a)$

3. $\sigma_{Moore}((q,a)) = \sigma_{Mealy}(q,a)$

   $\sigma_{Moore}(q_0') = \varepsilon$

Otrzymujemy transducer Moore'a $\langle \Sigma, \Sigma_1, Q', q_0', \delta', \sigma_{Moore} \rangle$ równoważny z pierwotnym t. Mealy'ego.
Dowód w obu przypadkach zapewne angażuje Zasadę Indukcji Matematycznej względem długości słowa.

Zadanie 78.

bso. (77) zajmijmy się transducerem Mealy'ego $\langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma_{Mealy} \rangle$, który świadczy że $A \leqslant_{reg} B$. Niech przy okazji $A_B = \langle \Sigma_1, Q^B, q_0^B, F^B, \delta^B \rangle$ będzie DFA rozpoznającym B. Definiujemy $\delta'(q,a) = \widehat{\delta^B}(q, \sigma_{Mealy}(q,a))$. Udajemy, że jesteśmy słowem z języka B i chodzimy po automacie A_B. Wtedy $\langle \Sigma, Q^B, q_0^B, F^B, \delta' \rangle$ jest DFA rozpoznającym A (d-d. indukcyjny względem długości słowa).

Zadanie 79.

Definiujemy transducer Mealy'ego T_{Mealy}:

1. $\Sigma = \{1, 2, 3, ..., n\}$
2. $Q = \Sigma$
3. $q_0 = 1$
4. $\delta(q,a) = a$
5. $\Sigma_1 = Q \times \Sigma$
6. $\sigma_{Mealy} = Id$

Niech $T_{Moore} = \langle \Sigma, \Sigma_1, Q', q_0', \delta', \sigma_{Moore} \rangle$ będzie transducerem Moore'a równoważnym z T_{Mealy}.

Obserwacja 1. *Możemy założyć, że każdy stan z Q' jest osiągany przez DFA stowarzyszony z T_{Moore}. W przeciwnym razie możemy usunąć te stany, a powstały T'_{Moore} wciąż będzie równoważny z T_{Mealy}.*

Obserwacja 2. $s \in Im(\sigma_{Mealy}) \Rightarrow |s| = 1$. *Zatem* $s \in Im(\sigma_{Moore}) \Rightarrow |s| = 1$.

Gdyby było $|Q'| < n^2$, to $Im(\sigma_{Mealy}) \nsubseteq Im(\sigma_{Moore})$. Niech $s \in Im(\sigma_{Mealy}) \setminus Im(\sigma_{Moore})$. $s = (k, l)$ dla pewnych $k, l \in \Sigma$. Rozważmy słowo $t = kl$. Wtedy $f_{T_{Mealy}}(t) = \sigma_{Mealy}(1, k)\sigma_{Mealy}(k, l) = (1, k)(k, l)$.
Załóżmy nie wprost, że $f_{T_{Moore}}(t) = f_{T_{Mealy}}(t)$. Jest to równoważne (obs. 2) z tym, że $\sigma_{Moore}(\delta'(q_0', k)) = \sigma_{Mealy}(1, k)$ oraz $\sigma_{Moore}(\widehat{\delta'}(q_0', kl)) = \sigma_{Mealy}(k, l) = (k, l)$. Druga równość stoi w jawnej sprzeczności z naszym założeniem, że $(k, l) \notin Im(\sigma_{Moore})$.

Zadanie 80.

Zdaje się, że świadczy o tym następujący transducer Mealy'ego:

1. $\Sigma = \{(,), [,], \langle, \rangle\}$
2. $Q = \{q_0\}$
3. $q_0 = q_0$
4. $\delta \equiv q_0$
5. $\Sigma_1 = \{(,), [,]\}$
6. $$\sigma_{Mealy}(q_0, (/)) = ((/))$$ $$\sigma_{Mealy}(q_0, [/]) = [[/]]$$ $$\sigma_{Mealy}(q_0, \langle/\rangle) = [(/)]$$

Dowód pozostawiamy Czytelnikowi jako ćwiczenie.