Zadanie 75

GdziejestNemo?

4 kwietnia 2020

Treść zadania

W zadaniu obowiązują następujące oznaczenia:

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

P będzie najmniejszą symetryczną relacją spełniającą

$$\forall_{w \in \Sigma^*} \; P(w, \epsilon)$$
$$\forall_{a \in \Sigma} \forall_{w,\ v \in \Sigma^*} \; P(w, v) \Rightarrow P(aw, av).$$

Dla $L \subseteq \Sigma^*$ definujemy :

$$L_{p/q} = \{w \in \Sigma * : \exists_{v \in L} \; P(w, v) \wedge |w|/|v| = p/q\}$$

Niech $L \in \Sigma^*$ będzie regularny. Czy wynika z tego, że:

A język $L_{3/2}$ jest regularny?

B język $\cup_{i=1}^{\infty} L_{1/i}$ jest regularny?

Rozwiązanie

a) Czy język $L_{3/2}$ jest regularny?

Ten język to język słów w, których prefiks o długości $\frac{2}{3}|W|$ należy do L. Pokażę, że język ten nie jest regularny.

Dowód. Weźmy język $L = \mathcal{L}(a^*)$. Załóżmy nie wprost, że istnieje automat rozpoznający $L_{3/2}$ DFA $A =< \Sigma, \delta, q_o, Q, F >$. Niech $|Q| = n$.
Spójrzmy na zbiór słów $\{\epsilon, a^2, a^4, ..., a^{2n}\}$. Zbiór ten ma $n + 1$ elementów. Z zasady szufladkowej Dirichleta wiemy, że istnieją $0 \leqslant i < j \leqslant n$ takie, że $\hat{\delta}(q_0, a^{2i}) = \hat{\delta}(q_0, a^{2j})$.
Z tego wynika, że $\hat{\delta}(q_0, a^{2i}b^j) = \hat{\delta}(q_0, a^{2j}b^j)$. Łatwo zauważyć, że słowo $a^{2j}b^j \in$

$L_{3/2}$.
Aby $a^{2i}b^j$ należało do $L_{2/3}$, słowo $a^{2i}b^{\frac{2}{3}j-\frac{2}{3}i}$ musiałoby należeć do L, a ponieważ $j > i$ nie bedzie to prawda. Zatem nasz automat nie rozpoznaje $L_{2/3}$. $\square$

b) Czy jezyk $\cup_{i=1}^{\infty} L_{1/i}$ jest regularny?

Ten język, to język słów w dla których istnieje i-razy dłuższe słowo należące do L, którego prefiksem jest w. Niech DFA $A =< \Sigma, \delta, q_o, Q, F >$ będzie automatem rozpoznającym L. Niech $|Q| = k$. Skonstruuję automat NDFA rozpoznający ten język.
Nasz automat rozpoznający $\cup_{i=1}^{\infty} L_{1/i}$ będzie się składał z dwóch automatów pomocniczych:

1. automatu DFA A
2. automatu NDFA C, który dla dowolnego stanu $q \in Q$, słowa w oraz dowolnego i będzie liczył w jakich stanach może się znaleźć A, jeżeli znajdował się w stanie q i wczytał dowolne słowo długości $i|w|$.

Aby skonstruować C skonstruujemy ciąg automatów NDFA $C_0, C_1, ... C_i$, który liczy dla dowolnego stanu $q \in Q$ i słowa w, w jakich stanach może się znaleźć A, jeżeli znajdował się w stanie q i wczytał dowolne słowo długości $i|w|$. Konstrukcję C_i można stworzyć w podobny sposób jak w zadaniu 23. Nasz automat będzie wykonywał krok epsilonowy do każdego możliwego stanu q startowego i zapamięta ten stan. Następnie będzie sprawdzał do jakich stanów może dojść ze stanu q wykonując po i dowolnych kroków dla każdej wczytanej litery.

$$C_i =< \Sigma; \delta_i; q_{i,0}; Q_i; \emptyset >$$
$$Q_i \subseteq Q \times Q \cup q_{i,0}$$

Funkcję przejścia zdefiniujemy w sposób indukcyjny.

$$\delta_i(q_{i,0}, \epsilon) = \{< q, q >: q \in Q\}$$
$$\delta_0(< q_1, q_2 >, a) = \{< q_1, q_2 >\}$$
$$\delta_i(< q_1, q_2 >, a) = \{< q_1, q_4 >: \exists_{q_3 \in Q} \ (< q_2, q_3 > \in \delta_{i-1}(< q_2, q_2 >, a) \wedge \exists_{b \in \Sigma} \ \delta(q_3, b) = q_4)\}$$

Z tej definicji wynika, że $< q_1, q_2 > \in \hat{\delta}_i(< q_1, q >, a)$, gdy istnieje słowo długości i, takie że automat A po wczytaniu go zmieni stan z q na q_2
$< q_1, q_2 > \in \hat{\delta}_i(< q_1, q >, a) \Leftrightarrow \exists_{v \in \Sigma^*} |v| = i \wedge \hat{\delta}(q, v) = q_2$

Lemat 1. $< q_1, q_4 > \in \hat{\delta}_i(< q_1, q_2 >, w)$, *wtedy i tylko wtedy gdy istnieje słowo długości* $i|w|$, *takie że automat A po wczytaniu go zmieni stan z* q_2 *na* q_4. *Innymi słowy:*
$< q_1, q_4 > \in \hat{\delta}_i(< q_1, q_2 >, w) \Leftrightarrow \exists_{v \in \Sigma^*} |v| = i|w| \wedge \hat{\delta}(q_2, v) = q_4$

Dowód. Udowodnię zależność indukcyjnie względem długości słowa $|w|$ na wejściu.
1. $w \in \Sigma$ - wynika z konstrukcji δ_i. (Można przeprowadzić naturalny dowód indukcyjny po indeksach kolejnych automatów dla tego faktu.)
2. Z.I. Dla każdego słowa w, takiego że $|w| \leqslant n$, spełnione jest $< q_1, q_4 > \in \hat{\delta}_i(< q_1, q_2 >, w) \Leftrightarrow \exists_{v \in \Sigma^*} |v| = i|w| \wedge \hat{\delta}(q_2, v) = q_4$.
Weźmy dowolne słowo $w \in \Sigma^{n+1}$. Niech $w = ma$, gdzie $a \in \Sigma$.
$\Rightarrow$
Wiemy, że $< q_1, q_4 > \in \hat{\delta}_i(< q_1, q_2 >, w)$.
Z tego wynika, że istnieje takie q_3, że $< q_1, q_3 > \in \hat{\delta}_i(< q_1, q_2 >, m)$ i $< q_1, q_4 > \in \hat{\delta}_i(< q_1, q_3 >, a)$.
Korzystając z założenia indukcyjnego dla słowa m istnieje słowo v' długości in, takie że $\hat{\delta}(q_2, v') = q_3$.
Korzystając z przypadku 1 $< q_1, q_4 > \in \hat{\delta}_i(< q_1, q_3 >, a)$ to istnieje u o długości i, takie że $\hat{\delta}(q_3, u) = q_4$. Zatem $\hat{\delta}(q_2, v'u) = q_4$.
$\Leftarrow$
Wiemy, że $\exists_{v \in \Sigma^*} |v| = i|w| \wedge \hat{\delta}(q_2, v) = q_4$.
Niech $v = yz$, $y \in \Sigma^{in}$, $z \in \Sigma^i$, $q_3 = \hat{\delta}(q_2, y)$.
Z założenia indukcyjnego dla słowa m otrzymujemy:
$< q_1, q_3 > \in \hat{\delta}_i(< q_1, q_2 >, m)$.
Wiemy także z przypadku 1, że $< q_1, q_4 > = < q_1, \hat{\delta}(q_3, z) > \in \hat{\delta}_i(< q_1, q_3 >, a)$, ponieważ z jest długości i. Zatem $< q_1, q_4 > \in \hat{\delta}_i(< q_1, q_2 >, w)$.

□

Teraz zdefiniujemy na tej podstawie C.

$$C = < \Sigma; \delta_\infty; q_{\infty,0}; Q_\infty; \emptyset >$$
$$Q \subseteq Q \times Q \cup q_{\infty,0}$$
$$\delta_\infty(q_{\infty,0}, \epsilon) = \{< q, q >: q \in Q\}$$
$$\delta_\infty(< q_1, q_2 >, a) = \cup_{i=0}^{\infty} \delta_i(< q_1, q_2 >, a)$$

Zauważmy, że sumy częściowe po zbiorach stanów kolejnych automatów będą ciągiem zbiorów monotonicznych względem inkluzji i ograniczonych z góry więc od pewnego momentu bedą stałe. Zatem δ_∞ będzie dobrze określona i można ją policzyć przez policzenie sumy częściowej skończonej liczby zbiorów stanów.

Teraz zdefiniujemy automat NDFA $A' = < \Sigma, \delta', q_0', Q', F' >$ rozpoznający $\cup_{i=1}^{\infty} L_{1/i}$.

$$Q' \subseteq Q \times Q_\infty$$
$$q_0' = < q_0, q_{\infty,0} >$$
$$\hat{\delta}'(< q_0, q_{\infty,0} >, w) = \{< \hat{\delta}(q_0, w), q_\infty >: q_\infty \in \hat{\delta}_\infty(q_{\infty,0}, w)\}$$
$$F = \{< q, < q, q_f >>: q_f \in F\}$$

Stany akceptujacy to stany oznaczające, że ze stanu końcowego da się dojść wielokrontością dotychczas wykonanych kroków do stanu akceptujacego w A, co jest równoważne należeniu do $\cup_{i=1}^{\infty} L_{1/i}$.