

Egzamin z teorii obliczeń, 5 lutego 2013

1. Dwugłowicowy (niedeterministyczny) automat skończony ma skończony zbiór stanów i dwie głowice czytające słowo wejściowe. W zależności od aktualnego stanu i dwóch obecnie oglądanych liter, każda z głowic przesuwa się w prawo lub pozostaje w miejscu. Formalnie, *dwugłowicowy automat skończony* nad alfabetem Σ definiujemy jako czwórkę postaci $\langle Q, q_0, q_a, \delta \rangle$, gdzie $q_0, q_a \in Q$ oraz

$$\delta \subseteq (Q \times \Sigma^2) \times (Q \times \{0, 1\}^2).$$

Obliczenie takiego automatu dla słowa wejściowego $w = a_1 \dots a_n$, to ciąg konfiguracji postaci $\langle q, i, j \rangle$, gdzie $q \in Q$ oraz $1 \leq i, j \leq n$. Zmiana konfiguracji $\langle q, i, j \rangle$ na $\langle p, i', j' \rangle$ jest możliwa wtedy, gdy $\langle \langle q, a_i, a_j \rangle, \langle p, \varepsilon_1, \varepsilon_2 \rangle \rangle \in \delta$, oraz $i' = i + \varepsilon_1$ i $j' = j + \varepsilon_2$ (jeśli np. $i = n$ oraz $\varepsilon_1 = 1$, to krok jest niemożliwy). Automat akceptuje słowo w , gdy istnieje obliczenie rozpoczynające się w konfiguracji $\langle q_0, 1, 1 \rangle$ i kończące się w $\langle q_a, n, n \rangle$. Uzasadnić (pomijając szczegóły) nierozstrzygalność problemu niepustości:

Czy dany dwugłowicowy automat skończony akceptuje co najmniej jedno słowo?

2. Mówimy, że zbiór $A \subseteq \mathbb{N}$ jest *monotonicznie przeliczalny*, gdy A jest zbiorem wartości pewnej niemalejącej funkcji rekurencyjnej $f : \mathbb{N} \rightarrow \mathbb{N}$. Udowodnić, że niepusty zbiór jest monotonicznie przeliczalny wtedy i tylko wtedy, gdy jest rekurencyjny.
3. Czy każdy nieskończony zbiór
- (a) klasy Σ_1^0
 - (b) klasy Π_1^0
- ma nieskończony podzbiór rekurencyjny?
4. Czy istnieje maszyna Turinga M , która dla dowolnego słowa wejściowego w wypisuje na wyjściu słowo w powtórzone dokładnie tyle razy, ile maszyna M ma stanów?

Rozwiązania:

Zadanie 1 : Dla dowolnej instancji $P = \{(x_i, y_i) \mid i = 1, \dots, n\}$ problemu odpowiedniości Posta konstruujemy automat dwugłowicowy, który akceptuje język rozwiązań dla P , tj. język wszystkich słów postaci $w = x_{i_1}x_{i_2}\dots x_{i_m} = y_{i_1}y_{i_2}\dots y_{i_m}$. Automat uruchomiony w stanie q_0 niedeterministycznie przechodzi do jednego ze stanów $q_i \in \{q_1, \dots, q_n\}$. Następnie (z pomocą pewnej liczby stanów pomocniczych) automat czyta pierwszą głowicą słowo x_i , a drugą głowicą słowo y_i , po czym przechodzi do stanu akceptującego q_a . (Jeśli jakiś symbol na taśmie nie zgadza się z oczekiwaniami, to automat przechodzi do stanu $q_\perp$, w którym nie może wykonać żadnego kroku.) Kolejne fazy pracy są analogiczne. Taki automat akceptuje co najmniej jedno słowo wtedy i tylko wtedy, gdy P ma rozwiązanie, ponadto otrzymaliśmy go efektywnie z P . W ten sposób zredukowaliśmy PCP do problemu niepustości.

Zadanie 2: Załóżmy, że A jest niepustym zbiorem rekurencyjnym; niech $f(0) = \min A$, oraz

$$f(n+1) = \begin{cases} n+1, & \text{jeśli } n+1 \in A; \\ f(n), & \text{w przeciwnym przypadku.} \end{cases}$$

Funkcja f jest rekurencyjna i niemalejąca, oraz $A = \text{Rg}(f)$.

Niech teraz A będzie monotonicznie przeliczalny i niech f będzie odpowiednią funkcją. Zbiory skończone są rekurencyjne, więc bez straty ogólności można założyć, że A jest nieskończony. Aby stwierdzić, czy dana liczba a należy do A obliczamy kolejne wartości $f(n)$ dla $n = 0, 1, 2, \dots$, aż uzyskamy $f(n) = a$ lub $f(n) > a$. Ta procedura się zakończy, bo A jest nieskończony i ma element większy bądź równy a .

Zadanie 3a: Tak. Niech A będzie rekurencyjnie przeliczalny i nieskończony. Jest taka rekurencyjna funkcja f , że $A = \text{Rg}(f) = \{f(n) \mid n \in \mathbb{N}\}$. Niech $B = \{f(m) \mid \forall i (i < m \rightarrow f(i) < f(m))\}$. Zbiór B jest monotonicznie przeliczalny i nieskończony; oczywiście $B \subseteq A$.

Zadanie 3b: Nie, na przykład zbiór $-X$ z dowodu twierdzenia 6.2 w skrypcie.

Zadanie 4: To zadanie jest „udramatyzowaną” wersją pierwszej części wniosku 4.5 ze skryptu. Niech M_m oznacza maszynę Turinga (nad ustalonym alfabetem) o numerze m i niech $q(m)$ będzie liczbą stanów maszyny M_m . Funkcja q jest obliczalna, podobnie jak funkcja $\lambda m w. w^{q(m)}$. Niech teraz $f(m)$ będzie numerem maszyny, obliczającej funkcję $\lambda w. w^{q(m)}$. Funkcja f też jest obliczalna, więc z twierdzenia o rekursji istnieje taka liczba n , że maszyna M_n oblicza to samo, co $M_{f(n)}$. A więc maszyna M_n robi to co trzeba.