Spis treści

Rozdział 1

Egzamin 2008

1.1. Część 2

1.1.1. Zadanie 4

Czy istnieje algorytm rostrzygający dla danych dwóch deterministycznych automatów ze stosem, czy istnieje niepuste słowo, które zostanie zaakceptowane przez oba te automaty?

Przez deterministyczny automat ze stosem rozumiemy tu automat, którego każdy krok polega na wczytaniu jedneo symbulu ze słowa wejściowego, jednego symbolu ze stosu i w zależności od wczytanych danych oraz od aktualnego stanu, na zmianie stanu i odłożeni na stos ciągu (być może pustego) symboli (w szczególności taki automat nie wykonue ε-przejść). Automat akceptuje słowo, jeśli po wczytaniu tego słowa stos jest pusty (tzn. zaiwera jedynie symbol dna stosu).

Taki algorytm nieistnieje. W celu pokazania tego, stworzymy redukcję, przedstawioną poniżej, problemu PCP (nierostrzygalnego) do naszego problemu z automatami.

$$f(\langle(l_1, r_1), \cdots, (l_k, r_k)\rangle) = (A_1, A_2) \text{ gdzie } A_1 \text{ i } A_2 \text{ to automat opisane w zadaniu} \tag{1.1}$$

Automaty A_1 i A_2 zbudujemy tak by akceptoway języki opisane poniżej:

$$L(A_1) = L(\{i_{j_1} \cdots i_{j_s} l^R_{j_s} \cdots l^R_{j_1}\}) \tag{1.2}$$

$$L(A_2) = L(\{i_{j_1} \cdots i_{j_s} r^R_{j_s} \cdots r^R_{j_1}\}) \tag{1.3}$$

Idea automaty A_1: Automat wczytuje tylke słowa postaci ciąg indeksów, po którym występuje ciąg literek, w przeciwnym przypadku zatrzymuje sie z niepustym stosem.

1. automat wczytuje kolejne indeksy i_j i wrzuca na stos słowa $l^R_{i_j}$
2. gdy zobaczy literke to ściąga ją ze stosu jeżeli taka sama leży na wierzchu lub się zacina

1.1.2. zadanie 5

Niech $A \subset N$ (nie jest r.e.) będzie zbiorem tych liczb naturalnych, ktore są numerami programów zatrzymujących się dla wszystkich danych. Niech $E \subset N$ (nie jest r.e.) bedzie zbiorem tych liczb naturalnych, które sa numerami programów nie zatrzymujujących się dla żadnych danych. Udowodnij, że $E \leqslant_{rek} A$.

Wskazówka: To jest łatwe zadanie. Przypomnij sobie co to jest redukjca. Pomyśl jakiego typu muszą byc argumenty redukcji, która masz zbudować i jakiego typu maja być wartości. Nie gap sie w pustą kartkę, napisz sobie równoważność, ktora ma zachodzić.

$$A = \{i : \forall_x \varphi_i(x) \neq \perp\} \tag{1.4}$$

$$E = \{j : \forall_x \varphi_j(x) = \perp\} \tag{1.5}$$

Mamy pokazać że:

$$E \leqslant_{rek} A \ \Leftrightarrow (\exists_f \forall_{x \in N} x \in E \Leftrightarrow f(x) \in A) \tag{1.6}$$

Pokażemy że istnieje takie f. W tym celu stworzymy odpowiednią redukcję, której kod przedstiony jest poniżej:

Niech $b : N \to N \times N$ będzie bijekcją (np taką przekątniową $b(1) = (1,1)$, $b(2) = (1,2)$, $b(3) = (2,1)$, ...).

```
wczytaj n
t = numer takiego programu:
        wczytaj m
        x, k = b(m)
        uruchom fi_n(x) na k krokow
                jesli fi_n(x) zatrzyma sie i zwroci wynik to zapetl sie
                w.p.p. zwroc 1
zwroc t
```

$n \in E$

$$n \in E \ \to \ \forall_x \varphi_n(x) = \perp \to \ \forall_x \varphi_t(x) = 1 \neq \perp \to t = f(n) \in A$$

$n \notin E$

$$n \notin E \ \to \ \exists_x \varphi_n(x) \neq \perp \to \ \exists_m b(m) = (x,k) \wedge \varphi_n(x) \text{ uruchomione na k kroków} \neq \perp \to \exists_m \varphi_t(m) = \perp \to n \notin A$$

1.1.3. zadanie 6

> Niech A i E będą takie, jak w poprzednim zadaniu. Czy zachodzi nierówność $A \leqslant_{rek} E$?
> Wskazówka: Co pamiętasz z ćwiczeń na temat nierówności $K \leqslant_{rek} \overline{K}$?

Korzystając ze wskazówki wiemy że $K \nleqslant_{rek} \overline{K}$. Pokażemy, że:

$$K \leqslant_{rek} A \ \wedge \ E \leqslant_{rek} \overline{K} \ \Rightarrow \ A \nleqslant_{rek} E \tag{1.7}$$

Najpierw pokażemy, że:

$$K \leqslant_{rek} A \Leftrightarrow \exists_f \forall_x x \in K \Leftrightarrow f(x) \in A \tag{1.8}$$

W tym celu skonstrujemy następująca redukcje f:

```
wczytaj n
t <- znajdz numer nastepujacego programu:
        wczytaj m
        zwroc fi_n(n)
zwroc t
```

$n \in K$

$$n \in K \Rightarrow \forall_x \varphi_t(x) = \varphi_n(n) \neq \perp \Rightarrow n \in A$$

$n \notin K$

$$n \notin K \Rightarrow \forall_x \varphi_t(x) = \varphi_n(n) = \perp \Rightarrow n \notin A$$

Teraz pokażemy, że:

$$E \leqslant_{rek} \overline{K} \Leftrightarrow \exists_f \forall_x x \in K \Leftrightarrow f(x) \in \overline{K} \tag{1.9}$$

W tym celu skonstrujuemy następującą redukcje f:

```
wczytaj n
t <- znajdz numer nastepujacego programu:
        wczytaj m
        dla kazdego i od 1 do ....
                dla kazdego j od 1 do ...
                        jezeli fi_n(i) zwroci wyniki po j krokach zwroc 1
        zwroc 1
zwroc t
```

$n \in E$

$$n \in E \Rightarrow \forall_x \varphi_n(x) = \perp \Rightarrow \neq \exists_{i,j} \varphi_n(i) \text{ zatrzyma sie po j krokach } \Rightarrow \forall \varphi_t(x) = \perp \Rightarrow \varphi_t(t) = \perp \Rightarrow t = f(n) \in \overline{K}$$

$n \notin E$

$$n \notin E \Rightarrow \exists_{i,j} \varphi_n(i) \text{ zatrzyma sie po j krokach } \Rightarrow \forall_x \varphi_t(x) = 1 \Rightarrow \varphi_t(t) = 1 \Rightarrow t = f(n) \notin \overline{K}$$

Bibliografia

[1] test reference