ZŁO* — ćwiczenia 9

Alternacja, powtórzenie

Zadanie 1 ($\diamondsuit$). Rozważmy następujący problem obliczeniowy: mamy dany niedeterministyczny automat na słowach $\mathcal{A}$ oraz słowo w nad alfabetem wejściowym $\mathcal{A}$. Pytamy się, czy $\mathcal{A}$ akceptuje w. Wykazać, że ten problem da się rozstrzygnąć w deterministycznej pamięci $\mathcal{O}(\log|\mathcal{A}| \cdot \log|w|)$.

Zadanie 2 ($\diamondsuit$). Wykaż, że jeśli $\mathsf{P} = \mathsf{NP}$, to $\mathsf{EXPTIME} = \mathsf{NEXPTIME}$.

Maszyna alternująca to maszyna Turinga o stanach podzielonych na $Q_\forall$ i $Q_\exists$. Przejścia albo prowadzą z jednej konfiguracji do drugiej, albo wywołują instrukcję akceptacji/odrzucenia. Maszyna akceptuje z konfiguracji ze stanem z $Q_\exists$ jeśli *istnieje* przejście, które albo natychmiast akceptuje, albo prowadzi do konfiguracji, z której maszyna akceptuje. Analogicznie dla konfiguracji ze stanami z $Q_\forall$ wymagamy by *każde* przejście miało tę własność. Maszyna działa w czasie $t(n)$ jeśli dla każdego słowa wielkości n można określić, czy maszyna akceptuje czy odrzuca na podstawie konfiguracji osiągalnych w co najwyżej $t(n)$ krokach. Maszyna używa pamięci $s(n)$ jeśli dla każdego słowa wielkości n można określić, czy maszyna akceptuje czy odrzuca na podstawie konfiguracji o pamięci roboczej $s(n)$. Klasy $\mathsf{ATIME}(t(n))$ oraz $\mathsf{ASPACE}(s(n))$ są zdefiniowane naturalnie. Oznaczamy

$$\mathsf{AL} = \bigcup_{c\in\mathbb{N}} \mathsf{ASPACE}(c\log n) \qquad \text{oraz} \qquad \mathsf{AP} = \bigcup_{c\in\mathbb{N}} \mathsf{ATIME}(n^c).$$

Zadanie 3. Wykaż, że jeśli język L jest w klasie $\mathsf{ATIME}(t(n))$, to również dopełnienie L jest w tej klasie. Udowodnij również taki sam fakt dla klasy $\mathsf{ASPACE}(s(n))$.

Zadanie 4. Wykaż, że $\mathsf{AP} = \mathsf{PSPACE}$.

Zadanie 5. Wykaż, że $\mathsf{AL} = \mathsf{P}$.

Zadanie 6. Rozważmy język nad alfabetem $\{0,1\}$ złożony z takich słów długości n^2 (dla $n \in \mathbb{N}$), które — zinterpretowane jako ciąg n liczb $m_1, \ldots, m_n$ reprezentowanych binarnie, każda przez ciąg n bitów — mają tę własność, że suma $m_1 + \ldots + m_n$ jest podzielna przez 3. Udowodnij, że ten język jest w klasie NC^1.

Zadanie 7. Wykaż, że następujące problemy są NP-zupełne:

(a) Vertex Cover: Dany jest graf G i liczba k. Pytamy się o istnienie zbioru wierzchołków X wielkości co najwyżej k, że każda krawędź ma co najmniej jeden koniec w X.

(b) Dominating Set: Dany jest graf G i liczba k. Pytamy się o istnienie zbioru wierzchołków X wielkości co najwyżej k, że każdy wierzchołek jest albo w X albo ma sąsiada w X.

(c) Set Cover: Dane jest uniwersum U, rodzina jego podzbiorów $\mathcal{F} \subseteq 2^U$, i liczba k. Pytamy się o istnienie podrodziny $\mathcal{G} \subseteq \mathcal{F}$ mocy co najwyżej k, dla której $\bigcup \mathcal{G} = U$.

Zadanie 8. Niech $x_1, \ldots, x_n$ będzie zbiorem zmiennych o wartościach w $\mathbb{Z}_2$. *Jednomianem stopnia* 1 nazywamy każdą pojedynczą zmienną x_i, zaś *jednomianem stopnia* 2 nazywamy iloczyn dwóch, niekoniecznie różnych zmiennych, np. $x_i x_j$. *Równanie kwadratowe nad* $\mathbb{Z}_2$ to przyrównanie pewnej sumy (nad $\mathbb{Z}_2$) jednomianów stopnia co najwyżej 2 do 0 lub 1, zaś *układ równań kwadratowych* to zbiór równań nad tym samym zbiorem zmiennych. Wartościowanie $\eta\colon \{x_1, \ldots, x_n\} \to \mathbb{Z}_2$ *spełnia* dany układ równań jeśli każde równanie jest spełnione po podstawieniu wartości w η pod odpowiadające im zmienne. Wykaż, że problem istnienia spełniającego wartościowania dla danego układu równań kwadratowych nad $\mathbb{Z}_2$ jest NP-zupełny.