

Zadanie 1. \mathcal{P} vs \mathcal{NPC} w NWP

Podstawy Analizy Algorytmów

Michał Małafiejski

Streszczenie

Celem realizacji zadania jest zbadanie granicy pomiędzy przypadkami łatwymi (\mathcal{P}) i trudnymi (\mathcal{NPH}) na przykładzie wybranego problemu. Dla niektórych szczegółowych wariantów ogólnie sformułowanego problemu należy zaproponować algorytmy o złożoności wielomianowej oraz pokazać \mathcal{NP} -zupełność dla wybranych przypadków szczegółowych.

Przez Σ będziemy rozumieli skończony alfabet złożony z liter (dowolnych symboli), np. $\Sigma = \{a, b, c\}$ lub $\Sigma = \{0, 1\}$. Przez *słowo* nad alfabetem Σ będziemy rozumieli dowolny skończony ciąg elementów z alfabetu, np. *aaba* lub 001100. Zbiór wszystkich skończonych słów nad alfabetem oznaczymy przez Σ^+ . Wprowadzimy tzw. *słowo puste* oznaczane przez ε ($\varepsilon \notin \Sigma$) oraz oznaczenie $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$. Dla dwóch słów $u, w \in \Sigma^*$ przez uw będziemy rozumieli słowo powstałe przez *konkatenację* (sklejenie) słów u oraz v . Zdefiniujemy słowo ε przez własności $w\varepsilon = \varepsilon w = w$, dla dowolnego $w \in \Sigma^*$. Długością słowa w , ozn. $|w|$, będziemy nazywali ilość kolejnych liter w słowie, np. $|aabad| = 5$. Niech $u, w \in \Sigma^*$, $u = u_1 \dots u_k$ oraz $w = w_1 \dots w_n$, gdzie $k = |u|$ oraz $n = |w|$. Słowo u nazwiemy *podciągami* słowa w , jeżeli $u_i = w_{j_i}$ dla $i \in \{1, \dots, k\}$, gdzie $1 \leq j_1 < j_2 < \dots < j_k \leq n$, np. *abbc* jest podciągami *xyababac*. Wprowadzimy oznaczenie $u \sqsubset w$, o ile u jest podciągami w .

Zdefiniujemy problem znajdowania najdłuższych wspólnych podciągów dla danego zbioru słów. Niech $u, w \in \Sigma^*$, przez $nwp(u, w)$ będziemy rozumieli $\max\{|v| : v \sqsubset u \wedge v \sqsubset w\}$, czyli długość najdłuższego wspólnego podciągu słów u oraz w . Analogicznie możemy zdefiniować $nwp(w_1, \dots, w_k)$ dla zbioru słów $\{w_1, \dots, w_k\}$.

Niech $L \subset \Sigma^+$. Problem NWP zdefiniujemy jako problem znalezienia najdłuższego wspólnego podciągu słów ze zbioru L (wersja optymalizacyjna). Wersja decyzyjna ma na wejściu dodatkowo liczbę $d \geq 0$ i pytamy czy $nwp(L) \geq d$. Wprowadzimy teraz wersję sparametryzowaną liczbami całkowitymi $p, q, r \geq 0$. Przez $NWP(p, q, r)$ będziemy rozumieli problem znajdowania najdłuższego wspólnego podciągu wszystkich słów ze zbioru $L \subset \Sigma^+$, gdzie $|L| \leq p$, $|\Sigma| \leq q$ oraz r oznacza ograniczenie na długość zwartych jednoliterowych ciągów w słowach ze zbioru L . Np. zapisując słowo *aabbabcccdaa* w postaci skompresowanej z krotnościami $a^2b^2abc^3da^2$ najdłuższy ciąg jednoliterowy to c^3 i ciąg ten należy do instancji dla $r = 3$, ale nie dla $r = 2$. W oznaczeniach przyjmujemy symbol „ \cdot ” oznaczający brak wybranej restrikcji.

Zadania punktowane (podstawowe 10pkt + dodatkowe 12 pkt)

- A. Pokaż, że $NWP(2, \cdot, \cdot) \in \mathcal{P}$ (algorytm wielomianowy, +4 pkt podstawowe)
- B. Zweryfikuj status problemu $NWP(\cdot, 2, 1)$ (algorytm wielomianowy lub α -redukcja, +2pkt podstawowe)
- C. Pokaż, że $NWP(\cdot, 2, \cdot) \in \mathcal{NPC}$ (α -redukcja, +4 pkt podstawowe)
- D. Pokaż, że $NWP(\cdot, 3, 1) \in \mathcal{NPC}$ (α -redukcja, +4 pkt dodatkowe)
- E. Pokaż, że $NWP(\cdot, 2, 2) \in \mathcal{NPC}$ (α -redukcja, +8 pkt dodatkowe)

Studenci, którzy sięgną po punkty dodatkowe będą poproszeni o zreferowanie dowodu \mathcal{NPC} przy tablicy (po stwierdzeniu poprawności). Pozostali studenci mogą zostać poproszeni o omówienie algorytmu lub α -redukcji w czasie konsultacji. Proszę nie ryzykować z pracą niesamodzielną (np. przepisywanie rozwiązania od innej osoby). Kara będzie dotkliwa: niezaliczenie przedmiotu i ewentualne konsekwencje administracyjne związane ze stwierdzeniem plagiatu. Można natomiast korzystać z literatury, książek i innych zasobów, ale warunkiem skorzystania jest podanie stosownej informacji w bibliografii.

Terminy: punkty podstawowe (72h, wtorek 23.01, godz. 12:00), punkty bonusowe (31.01, 24:00)