14. CVIČENÍ Z ADS 1, ČTVRTEK 19.5. 10:40

Dynamické programování a Quick{Select,Sort}

1. *Nejdelší společná podposloupnost.* Navrhněte algoritmus pro nalezení nejdelší společné podposloupnosti daných posloupností $x_1, \ldots, x_n$ a $y_1, \ldots, y_m$. Jak tento problém souvisí s editační vzdáleností? Jaký je grafový pohled na výsledný algoritmus?

2. *Násobení mnoha matic.* Násobíme-li matice $X \in \mathbb{R}^{a \times b}$ a $Y \in \mathbb{R}^{b \times c}$ podle definice, počítáme $a \cdot b \cdot c$ součinů čísel (používáme přímočarý algoritmus z definice násobení). Pokud chceme spočítat maticový součin $X_1 \cdot X_2 \cdots X_n$, výsledek nezávisí na uzávorkování, ale časová složitost (měřená pro jednoduchost počtem součinů čísel) ano. Vymyslete algoritmus, který stanoví, jak výraz uzávorkovat, abychom složitost minimalizovali.

3. *Nejlevnější vrcholové pokrytí stromu.* Dáno městečko ve tvaru stromu (ulice odpovídají hranám a křižovatky vrcholům). Chceme na křižovatky umístit strážníky, aby každá ulice měla strážníka alespoň na jednom konci. Každá křižovatka má danou cenu, za kterou na ní lze umístit strážníka. Cílem je najít nejlevnější rozmístění strážníků.

   (Výběru podmnožiny vrcholů grafu takové, že každá hrana má alespoň jeden konec vybraný, se říká vrcholové pokrytí.)

4. *QuickSelect.* Co kdybychom v QuickSelectu používali jako pivota aritmetický průměr čísel na vstupu? Jaká bude časová složitost v nejhorším a nejlepším případě?

   A jak by to dopadlo, kdybychom vždy vybírali za pivot „skoroskoromedián“, což je prvek, který leží v prostředních šesti osminách vstupu?

5. *Proč pětice?* V deterministickém algoritmu pro výběr k-tého nejmenšího v lineárním čase používáme jako pivota pro QuickSelect medián z mediánů *pětic*. Co kdybychom místo pětic používali trojice? Nebo sedmice? Jak by to dopadlo s časovou složitostí?

6. *Triangulace mnohoúhelníku.* Konvexní mnohoúhelník můžeme triangulovat, tedy rozřezat neprotínajícími se úhlopříčkami na trojúhelníky. Nalezněte takovou triangulaci, aby součet délek řezů byl nejmenší možný

7. *Medián dvou setříděných polí.* Mějme dvě setříděné posloupnosti (ne nutně stejně dlouhé), reprezentované v poli. Jak najít jejich medián v sublineárním čase (tedy $o(n)$)?

8. *Editační vzdálenost rychleji.* Na první pohled se zdá, že čím podobnější řetězce dostaneme, tím by mělo být jednodušší zjistit jejich editační vzdálenost. Algoritmus z přednášky ovšem pokaždé vyplňuje celou tabulku dynamického programování. Ukažte, jak ho zrychlit, aby počítal v čase $O((n+m) \cdot (ED(s,t)+1))$, kde $|s| = n$, $|t| = m$ a $ED(s,t)$ je editační vzdálenost s a t.

Hint: Nejprve zkuste vymyslet algoritmus, který dostane číslo $D \geq 1$ a v čase $O((n+m) \cdot D)$ buď korektně ohlásí, že $ED(s,t) > D$, nebo vrátí $ED(s,t)$.

9. *Nejkratší společná **nad**posloupnost.* Jak najít nejkratší společnou nadposloupnost dvou posloupností $x_1, \ldots, x_n$ a $y_1, \ldots, y_m$? Tím se myslí nejkratší posloupnost, která obsahuje jako podposloupnosti jak $x_1, \ldots, x_n$, tak $y_1, \ldots, y_m$.