
Termín odevzdání: 19. 3. 2014 do 9:55 hod.
Podrobné informace v interaktivní osnově předmětu v ISu.
Řešení každého příkladu na samostaném listu papíru.

Příklad 1 *30 bodů*
Je dané pole obsahující n vzájemně různých čísel $x_1, x_2, \ldots, x_n$. Ke každému číslu je asociována kladná hodnota $w_1, w_2, \ldots, w_n$ taková, že platí $\sum_{i=1}^{n} w_i = 1$. *Optimálním prvkem* posloupnosti nazveme číslo x_k, pro které platí

$$\sum_{i:x_i<x_k} w_i < 1/2$$

a současně

$$\sum_{x_i>x_k} w_i < 1/2.$$

Navrhněte algoritmus, který najde optimální prvek s časovou složitostí $\Theta(n)$.
Nezapomeňte zdůvodnit korektnost svého algoritmu a uvést analýzu jeho časové složitosti.

Příklad 2 *30 bodů*
Je daná $n \times n$ matice vzájemně různých čísel. Číslo v matici nazveme *esem* právě když je větší než všechna s ním sousedící čísla, tj. čísla, které se v matici nachází o řádek výš, o řádek níž, napravo a nalevo. Skoro všechna čísla v matici mají čtyři sousedy. Čísla v krajních sloupcích a v prvním a posledním řádku mají jenom tři sousedy; čísla v rozích mají jenom dva sousedy.
Navrhněte rekurzivní algoritmus, který najde eso v matici. Jedinou povolenou operací nad prvky matice je jejich vzájemné porovnání. Požadujeme, aby algoritmus vykonal jenom $\mathcal{O}(n)$ porovnání.
Návod: uvažte, které typy rekurentních rovnic splňují požadované omezení.
Nezapomeňte zdůvodnit korektnost svého algoritmu a uvést analýzu jeho časové složitosti.

Příklad 3 *30 bodů*
234 strom je strom, ve kterém každý vnitřní vrchol má 2, 3, nebo 4 syny a všechny listy mají stejnou výšku. 234 strom použijeme na reprezentaci množiny n klíčů (přirozených čísel). Klíče jsou uloženy v listech stromu, v každém listu x jeden klíč $key[x]$. Každý vnitřní vrchol x obsahuje hodnotu $small[x]$ která je rovna nejmenšímu klíči uloženému v listu podstromu s kořenem x. Navrhněte implementaci následujících operací.

- MINIMUM vrátí ukazatel na list obsahující nejmenší klíč.
- INSERT(k) vloží do stromu nový list s klíčem k.
- DECREASE KEY(x, k) změní klíč uložený v listu x na danou hodnotu $k \leq key[x]$.
- DELETE(x) odstraní ze stromu list x.
- EXTRACT MIN odstraní ze stromu list obsahující nejmenší klíč.

Požadujeme, aby složitost všech uvedených operací byla pro 234 strom obsahující n klíčů nejvýše $\mathcal{O}(\log n)$.
Nezapomeňte zdůvodnit korektnost svého řešení a uvést analýzu jeho časové složitosti.

Příklad 4 *20 bodů*

Uvažujme seznam S přirozených čísel. Nad seznamem vykonáváme následující operace.

- INSERT(S, i) vloží na začátek seznamu číslo i; cena operace je 1.
- MIN-ALL(S) vymaže ze seznamu S všechna čísla kromě minimálních čísel; cena operace je rovná délce seznamu.
- MIN-ONE(S) vymaže ze seznamu S všechna čísla kromě jednoho minimálního čísla; cena operace je rovná délce seznamu.
- DELETE(S, i) — vymaže ze seznamu S všechna čísla kromě čísel i; cena operace je rovná délce seznamu.

Na začátku výpočtu je seznam prázdný. O následujících tvrzeních rozhodněte, zda jsou korektní.

1. Libovolná posloupnost n operací typu INSERT a MIN-ALL má složitost $\mathcal{O}(n)$.
2. Libovolná posloupnost n operací typu INSERT a MIN-ONE má složitost $\mathcal{O}(n)$.
3. Libovolná posloupnost n operací typu INSERT a DELETE má složitost $\mathcal{O}(n)$.
4. Libovolná posloupnost n operací typu INSERT a DELETE taková, že při každém volání se operace DELETE volá s jiným parametrem i, má složitost $\mathcal{O}(n)$.

V případě, že tvrzení platí, proveďte analýzu složitosti technikou účtů nebo technikou potenciálových funkcí. V případě, že tvrzení neplatí, uveďte protipříklad.

Příklad 5 *Prémiový příklad 5 bodů*

Lokalizační problém je definovaný následovně. Je daných n lokací $p_1, p_2, \ldots, p_n$, každá lokace má asociovanou cenu $w_1, w_2, \ldots, w_n$. Cílem je najít lokaci p (nemusí to být nutně jedna z lokací $p_1, p_2, \ldots, p_n$), která minimalizuje součet $\sum_{i=1}^{n} w_i d(p, p_i)$, kde $d(x, y)$ je vzdálenost mezi lokacemi x, y.

a) Dokažte, že optimální prvek z příkladu 1 je korektním řešením pro jednorozměrný lokalizační problém, ve kterém lokace jsou reálná čísla a vzdálenost mezi lokacemi x, y je $d(x, y) = |x - y|$.

b) Najděte řešení problému dvourozměrného lokalizačního problému. Lokací je dvojice čísel (x, y) a vzdálenost mezi lokacemi (x_1, y_1) a (y_1, y_2) je $|x_1 - x_2| + |y_1 - y_2|$.