

# ADS 1 příklady na cvičení

29. 5. 2020

1. Kritická hrana budiž taková, která leží na všech nejkratších cestách. Tedy ta, jejíž prodloužení by ovlivnilo vzdálenost. Navrhněte algoritmus, který najde všechny kritické hrany.
2. Navrhněte operaci  $\text{Join}(X,Y)$ , která dostane dva  $(a,b)$ -stromy  $X$  a  $Y$  a sloučí je do jednoho. Může se přitom spolehnout na to, že všechny klíče z  $X$  jsou menší než všechny z  $Y$ . Zkuste dosáhnout složitosti  $O(\log|X| + \log|Y|)$ .
3. Naleznete 2. nejmenší kostru. Kromě postupu jak takovou kostru nalézt je potřeba i dokázat, že tento postup funguje.
4. Na cvičení jsme dokázali dokonale vyvážit BVS v lineárním čase tím, že jsme ho vypsalí do pole (seřazeného) a toto pole jsme přetvořili na dokonale vyvážený BVS. Vyřešte toto cvičení tak, aby vám kromě zadaného stromu stačilo logaritmické množství paměti. (Nebo ještě lépe konstantní)
5. Pro graf  $G$  máte nalezenou minimální kostru. Váha jedné hrany v  $G$  se změní. Jak nalézt co nejrychleji minimální kostru?
6. Koupili jste na inzerát dvojici skvělých robotů. Lacino, neboť jsou právě uvězněni v bludišti (čtvercová síť s některými políčky blokovanými). Znáte jejich polohy a můžete jim rádiem vysílat povely pro posun o políčko na sever, jih, východ či západ, abyste je dostali na okraj bludiště. Háček je ale v tom, že na každý povel reagují oba roboti. Vymyslete algoritmus, který najde nejkratší posloupnost povelů, jež vysvobodí oba roboty. Dodejme ještě, že robot ignoruje povel, který by způsobil okamžitý náraz do zdi, a že jakmile se robot dostane na okraj, odchytíme ho a další povely neposlouchá.
7. Ve skriptech je ukázáno, že dokonale vyvážený strom o  $2^{k-1}$  vrcholech je úplný – všech  $k$  hladin obsahuje nejvyšší možný počet vrcholů. Dokažte, že ostatní dokonale vyvážené stromy mají podobnou strukturu, jen na poslední hladině mohou některé vrcholy chybět.
8. něco k asymptotické notaci (tam je mi asi jedno, jestli úkol, nějaká úloha ze skript nebo cokoliv jiného) - TODO
9. Šroubky a matičky: Na stole leží  $n$  různých šroubků a  $n$  matiček. Každá matička pasuje na právě jeden šroub a my chceme zjistit, která na který. Umíme ale pouze porovnávat šroub s matičkou – tím získáme jeden ze tří možných výsledků: matička je příliš velká, příliš malá, nebo správně velká. Naleznete co nejefektivnější algoritmus.
10. Okénková minima: Na vstupu postupně přicházejí čísla. Kdykoliv přijde další, vypište minimum z posledních  $k$  čísel. Na rozdíl od cvičení na okénkový medián existuje i řešení pracující v amortizované konstantním čase na operaci.
11. Vymyslete algoritmus, který nalezne všechny hrany, jež leží na alespoň jedné nejkratší cestě.
12. Trojúhelníky: Spočítejte v rovinném grafu trojúhelníky, tedy trojice vrcholů spojené každý s každým. Řešte v  $O(n+m)$ .
13. Navrhněte operaci  $\text{Split}(T,x)$ , která zadaný  $(a,b)$ -strom  $T$  rozdělí na dva stromy. V jednom budou klíče menší než  $x$ , v druhém ty větší. Pokuste se o logaritmickou časovou složitost.
14. Okénkový medián: Na vstupu postupně přicházejí čísla. Kdykoliv přijde další, vypište medián z posledních  $k$  čísel. Dosáhněte časové složitosti  $O(\log k)$  na operaci. Jako bonusovou variantu (bez odměny navíc) zkuste dokázat, že čas  $\Theta(\log k)$  je nejlepší možný, pokud umíme čísla pouze porovnávat.
15. Minimový strom pro posloupnost  $x_1, \dots, x_n$  navzájem různých prvků je definován takto: v kořeni leží prvek  $x_j$  s nejmenší hodnotou, levý podstrom je minimovým stromem pro  $x_1, \dots, x_{j-1}$ , pravý podstrom pro  $x_{j+1}, \dots, x_n$ . Navrhněte algoritmus, který sestrojí minimový strom v čase  $O(n)$ .

16. Inverze matice: Navrhněte algoritmus typu Rozděl a panuj na výpočet inverze trojúhelníkové matice  $n \times n$  v čase lepším než  $\Omega(n^3)$ . Jako podprogram se může hodit Strassenovo násobení matic. Můžete předpokládat, že  $n$  je mocnina dvojky.
17. Upravte funkci LinearSelect tak, aby si vystačila s konstantně velkou pomocnou pamětí. Prvky ve vstupním poli můžete libovolně přeskupovat.
18. Jak bude vypadat strom rekurzivních volání funkce LinearSelect? Kolik bude mít listů? Jak dlouhá bude nejkratší a nejdelší větev?