13. cvičení z ADS1

27. 5. 2021

Příklad 1.

Ve standardní úloze Hanoiských věží, kde přesouváme věž z tyče A na tyč B navíc zakážeme přesun částí přímo mezi tyčemi A a B. Pro jaká n bude úloha stále řešitelná? Vymyslete algoritmus řešící problém. Kolik kroků trvá optimální řešení?

Příklad 2.

Upravme algoritmus QuickSelect tak, že za pivoty budeme volit „skoroskoromediány“, které leží v prostředních česti osminách vstupu. Jaká bude časová složitost algoritmu?

Příklad 3.

QuickSort je sice hezký algoritmus, ale může potřebovat až $\Theta(n)$ pomocné paměti kvůli rekurzivním krokům na zásobníku. Proto místo toho uvážíme jinou variantu rekurze. Na zásobník vždy uložíme *větší* úsek a budeme pokračovat tříděním menšího úseku. Jakmile budeme hotoví se tříděním menšího úseku, vytáhneme ze zásobníku úsek, a ten začneme taky třídit.

Dokažte, že po této úpravě může být v libovolný okamžik na zásobníku jen $\mathcal{O}(\log n)$ úseků. Tím množství potřebné pomocné paměti podstatně kleslo.

Příklad 4.

Uvažme náhodně konstruovaný (nevyvažovaný) binární vyhledávací strom, do kterého chceme vložit množinu $[n]$. V každém kroku náhodně zvolíme jeden z ještě nepřidaných prvků, a následně jej vložíme do BVS. Takto postupujeme, dokud nepřidáme všechny vrcholy.

Dokažte, že průměrná hloubka takového stromu bude $\mathcal{O}(\log n)$.

Nápověda: Zkuste si rozmyslet, jak tato úloha souvisí s QuickSortem.

Příklad 5.

Vymyslete datovou strukturu, pomocí které půjde nalézt nejdelší rostoucí podposloupnost v čase $O(n \log n)$, kde n je délka posloupnosti.

Příklad 6.

Dešifrovali jsme tajnou depeši, ale chybí v ní mezery. Známe však slovník všech slov, která se v depeši mohou vyskytnout. Chceme tedy rozdělit depeši na co nejméně slov ze slovníku.