

9. CVIČENÍ Z ADS 1

Viktor Němeček 15. 4. 2019

<https://kam.mff.cuni.cz/~viki/vyuka/ads11819/>

Příklad 1. Převedte libovolný strom na perfektně vyvážený v čase $\mathcal{O}(n)$.

* **Příklad 2.** Vyřešte předchozí příklad pouze pomocí rotací a v paměti $\mathcal{O}(1)$ (mimo samotného stromu). Můžete počítat s tím, že $n = 2^k - 1$ pro nějaké $k \in \mathbb{N}$.

** **Příklad 3.** Mějme model, v němž je pointer absolutně amorfní, tedy nelze s ním dělat nic jiného, než dereferencovat (například nemůžete vzít dva pointery a jejich adresy porovnávat menšítkem). V tomto modelu navrhnete datovou strukturu binárního stromu, tak, aby každý vrchol obsahoval nejvýše dva pointery, ale abyste v konstantním čase pro daný vrchol aniž byste využili klíčů:

- Zjistili, zda se jedná o kořen.
- Zjistili, zda má jednoho syna, dva či nula.
- Má-li dva, našli pravého a levého (musíte být schopní poznat, který je který).
- Má-li jednoho syna, našli tohoto syna (nemusíte poznat, zda je pravý či levý).
- Není-li vrchol kořen, našli jeho otce.

Příklad 4. Dokažte, že merge dvou BVS o velikostech n a m nemusí jít provést rychleji, než v čase $\Theta(\min(m, n))$ (pro libovolně vysokou hodnotu $\min(m, n)$), a to ani v případě, že vstupní stromy jsou dokonale vyvážené a výstupní být nemusí. Ukažte, že v $\mathcal{O}(n + m)$ to naopak možné je vždy.

Příklad 5. Navrhnete co nejrychlejší algoritmus, který z jednoho BVS udělá dva tak, že v jednom z nich budou všechny hodnoty menší než nějaká konstanta, a ve druhém ty větší nebo rovný.

Příklad 6. V AVL stromu máme standardně v každém vrcholu uloženou informaci o vyvážení, která může nabývat tří různých hodnot. Ukažte, jak si vystačit pouze s jedním bitem v každém vrcholu.

Příklad 7. Dokažte, že každý AVL strom lze obarvit tak, abychom dostali validní červenočerný (ne nutně left-leaning) strom.

Příklad 8. Který strom může být hůře vyvážený (tj. mít horší poměr mezi délkou nejdelší větve, a hloubkou perfektně vyváženého stromu o stejném počtu prvků), AVL nebo LLRB?