

Příklad 1.

Dokažte, že perfektně vyvážený strom má všechny hladiny až na poslední zaplněné. Rozmyslete si, zda může ta poslední už vypadat jakkoliv, či zda musí splňovat nějaké další požadavky.

Příklad 2.

Představte si, že BVS použijeme na ukládání k -tic přirozených čísel, které porovnáváme lexikograficky. Jaká bude časová složitost jednotlivých operací? Kolik paměti tento strom zabere?

Příklad 3.

Zkusme uvážit Poměrově vyvažovaný vyhledávací strom, ve kterém pro každý vrchol v platí $1/2 \leq \frac{|l(v)|}{|r(v)|} \leq 2$. Dokažte, že takový strom má logaritmickou hloubku.

Příklad 4.

Zkusme naimplementovat vkládání vrcholu do Poměrově vyvažovaného BVS následně: Přidáme list a poté, pokud se pro libovolný v poruší vyváženost, tak zvolíme v nejbližší ke kořeni a celý jeho podstrom předěláme na perfektně vyvážený.

Ukažte, že takové vkládání vždy udrží BVS Poměrově vyvážený.

Příklad 5.

Navrhněte co nejrychlejší algoritmus, který pro dané k rozdělí BVS na dva tak, aby jeden obsahoval klíče menší než k a ten druhý větší nebo rovno k .

Příklad 6.

Mějme dva BVS velikostí n, m takové, že je chceme spojit do jednoho. Navrhněte algoritmus, který to udělá v čase $\mathcal{O}(n + m)$. Půjde to v čase $\mathcal{O}(\min(n, m))$?

Příklad 7.

Zkuste vymyslet způsob, jak pro určité volby m a n spojit dva stromy v čase lepším, než $\mathcal{O}(n + m)$.

Příklad 8.

AVL strom pro svou práci potřebuje mít v každém vrcholu značky tří druhů. Ukažte, že ale existuje způsob, jak udržovat AVL strom s jedním bitem na vrchol.