

Cvičení 8 – AVL, (a, b) -stromy

Z minula

Úloha 1 (*Dokonalé vyvážení*) Navrhněte algoritmus, který ze seřazeného pole vyrobí v lineárním čase dokonale vyvážený BVS.

Úloha 2 (*Operace*) Ukažte, jak s BVS implementovat následující operace, a určete časovou složitost:

- Nalezení minimálního/maximálního klíče
- Nalezení následníka pro zadaný vrchol
- Nalezení k -tého nejmenšího prvku

Strom si můžete upravit, aby udržoval nějaké informace navíc.

Úloha 3 (*Iterovaný následník*) Dokažte, že projdeme-li celý strom opakovaným hledáním následníka, strávíme tím čas $\Theta(n)$.

AVL stromy

Úloha 4 Odsimulujte postupné přidání čísel 10, 20, 15, 25, 30, 16, 18 a 19 do na začátku prázdného AVL stromu. Co se stane, pokud na konci odstraníme číslo 30?

Úloha 5 V AVL stromu je potřeba udržovat v každém vrcholu znaménko $\{-, 0, +\}$ (udávající rozdíl hloubek podstromů), na což jsou potřeba dva bity. Ukažte, že existuje způsob, jak si vystačit s jedním bitem na vrchol.



Jakub Komárek

komarek+ads1@iuuk.mff.cuni.cz

<https://jakoma02.cz/teaching/ls24/ads1/>

Aplikace

Úloha 6 Sestrojte (složenou) datovou strukturu, která v zadaném čase zvládne následující operace:

- $\text{INIT}()$ inicializuje datovou strukturu v $\mathcal{O}(1)$,
- $\text{INSERT}(X)$ vloží prvek X pokud v datové struktuře ještě není v $\mathcal{O}(\log n)$,
- $\text{DELETE}(X)$ smaže X pokud je ve struktuře v $\mathcal{O}(\log n)$,
- $\text{DELETEINPLACE}(I)$ smaže I -tý přidaný prvek v $\mathcal{O}(\log n)$,
- $\text{GETPLACE}(X)$ vrátí číslo I , kolikátý byl vložený prvek X v $(\log n)$.

Úloha 7 (*Intervalový update*) Mějme BVS reprezentující slovník dvojic (klíč, hodnota), přičemž hodnoty jsou číselné. Upravte jej, aby podporoval operaci $\text{ADD}(x, y, \delta)$, která k hodnotám všech klíčů v intervalu $[x, y]$ přičte δ . Tato operace má běžet v $\mathcal{O}(\log n)$, takže nemusíme hned provést aktualizaci hodnot všech klíčů v daném intervalu, stačí když $\text{FIND}(k)$ vrátí správnou hodnotu klíče k .

(a, b) -stromy

Úloha 8 (*Volba parametrů*) Určete, jak závisí složitost operací s (a, b) -stromy na parametrech a a b . Z toho odvoďte, že se nikdy nevyplatí volit b výrazně větší než $2a$.

Úloha 9 Navrhněte operaci $\text{JOIN}(X, Y)$, která dostane dva (a, b) -stromy X a Y a sloučí je do jednoho. Můžete se přitom spolehnout na to, že všechny klíče z X jsou menší než všechny z Y . Za tohoto předpokladu zkuste dosáhnout složitosti $\mathcal{O}(\log |X| + \log |Y|) = \mathcal{O}(\max\{\log |X|, \log |Y|\})$.

Úloha 10 (*Jednosměrné operace*) Navrhněte úpravu operací INSERT a DELETE do (a, b) -stromů, aby operace procházely stromem pouze od kořene dolů a nebylo nutné jít zpět. Předpokládejte $b \geq 2a$.



Jakub Komárek

komarek+ads1@iuuk.mff.cuni.cz

<https://jakoma02.cz/teaching/ls24/ads1/>