

PŘÍKLAD PRVNÍ

Nejprve rozvička. Připomeňte si argument, proč dynamicky nafukované pole má amortizovanou složitost $O(1)$ pro vložení.

PŘÍKLAD DRUHÝ

Mějme algoritmus, který pracuje s textem délky n následovně. Převezme písmeno z textu a buď jej uloží do zásobníku a posune se dále, nebo vyhodí ze zásobníku nějaký počet písmen, které v něm byly, a posune se dále.

Dokažte, že takový algoritmus pracuje v čase $O(n)$. Mimochodem, ačkoli jsme vynechali detaily, tak toto je odhad složitosti algoritmu Knuth-Morris-Pratt pro vyhledávání v textu, který ještě potkáte.

PŘÍKLAD TŘETÍ

Mějme setříděné pole čísel. Jak rychle z něj umíte postavit perfektní BVS? Perfektní BVS je takový strom, kde kromě pravidel pro vyhledávací stromy pro každý vrchol platí, že počet jeho vrcholů v levém a pravém podstromu se liší nejvýše o jedna.

PŘÍKLAD ČTVRTÝ

Restaurujeme stromy. Představme si, že máme BVS, a to takovýto: neobsahuje žádné rotace, vrcholy vkládá tam, kam patří (jako ostatní BVS). V každém vrcholu si navíc pamatuje velikost celého jeho podstromu. Jakmile poměr velikostí levého ku pravému přesáhne $2 : 1$, tak celý podstrom vezme a přestaví ho v perfektní BVS. (Při malých velikostech od 0 do řekněme 4 nedělá samozřejmě nic.)

Zřejmě složitost vkládání nebo mazání v nejhorším případě bude lineární, nicméně dokažte, že složitost vyhledávání bude $O(\log n)$ a amortizovaná složitost vkládání nebo mazání je také $O(\log n)$.

Může mít BVS amortizovanou složitost vkládání menší, než $O(\log n)$?

PŘÍKLAD PÁTÝ

Binomiální haldy jsou na vlastnosti trochu bohatší haldy, které silně využívají rankových vlastností a při jejich analýze se hodí amortizovaná složitost.

Nejprve, co je to binomiální strom řádu r : Binomiální strom řádu 0 je strom o jednom vrcholu, řádu 1 je strom s kořenem a jedním synem, a řádu r je strom ranku $r - 1$, který má navíc u kořene (jako syna) zapojen druhý binomiální strom ranku $r - 1$. Viz obrázek na tabuli.

Všimněte si, kolik má binomiální strom řádu r celkem vrcholů a jaký má maximální stupeň.

Nyní, co to je binomiální halda. To je seznam binomiálních stromů, pro které platí dvě vlastnosti:

- Stromy jsou seřazeny podle rostoucího řádu, ale každý řád je tam nejvýše jednou (nicméně může chybět).
- V jednotlivých stromech jsou ve vrcholech uložena čísla, a to podle klasické haldové vlastnosti – v otci je menší číslo, než ve všech jeho synech.

Pro pořádek ještě upozorňuji, že mezi jednotlivými stromy není žádné uspořádání, a tedy minimum celé haldy může být v libovolném z kořenů stromů. Z toho důvodu si mimo tenhle seznam udržujeme ještě speciální odkaz na minimum celé haldy.

Nyní už k samotnému cvičení:

1. Máme-li v binomiální haldě n vrcholů, jaký je největší řád, který se tam může objevit?
2. Jak byste sloučili dvě haldy v jednu? Zkuste to stihnout v čase $O(\max\{\log n, \log m\})$.

Nápověda: Souvisí slučování dvou hald se sčítáním binárních čísel?

3. Vymyslete, jak pomocí myšlenky v předchozím bodě vkládat do binomiální haldy v čase $O(\log n)$ a amortizovaně v čase $O(1)$.
4. Nakonec si rozmyslete, jak pomocí myšlenky v bodě 2 vyjmout minimum v čase $O(\log n)$.