2. cvičení

Úloha 1 (Perfectly balanced, as all things should be)
Navrhněte algoritmus, který ze seřazeného pole v lineárním čase vytvoří perfektně vyvážený BVS. (Tedy pro každý vrchol musí platit, že počet vrcholů v levém podstromu se od počtu vrcholů v pravém podstromu smí lišit maximálně o 1.)

Řešení
Rozděl a panuj: vezmeme prostředek (délku pole víme předem, nebo si ji jedním průchodem spočítáme) a zarekurzíme se na levé podpole a pravé podpole (už s informací o délce).

Úloha 2 (Líně vyvažované stromy)
Připomeňte si, jak fungují líně vyvažované BVS a jakým potenciálem se analyzují.

a) Jak dlouho může trvat provedení k operací provedených na libovolném BB[α] stromu s n vrcholy?

b) Proč je v definici potenciálu výjimka pro rozdíl 1, tedy co by se pokazilo, kdybychom ji neudělali?

Řešení
Potenciál je součet přes potenciál vrcholů, kde každý vrchol má potenciál podle $|s(\ell(v)) - s(r(v))|$, kde s je velikost podstromu, $\ell(v)$ je levý podstrom a $r(v)$ je pravý podstrom. Je-li tento rozdíl 0 nebo 1, pak $\varphi(v) = 0$, jinak $\varphi(v) = |s(\ell(v)) - s(r(v))|$. Analýza pak probíhá tak, že díky vyvážení máme garantovanou $\mathcal{O}(\log n)$ hloubku, a držíme si invariant, že $|s(\ell(v)) - s(r(v))| \leq (2\alpha - 1)s(v)$. Když ten invariant porušíme, tak ale máme v potenciálu vrcholu dost na to, že zvládneme zaplatit přestavění celého podstromu v lineárním čase. Amortizovaná cena tedy bude $\mathcal{O}(\log n)$, protože zbytek se vyřeší potenciálem.

a) $\mathcal{O}(k \log n + n \log n)$

b) Ne každý strom umíme vyvážit přesně, a kdybychom měli rozdíl 1 v každém vrcholu, tak musíme platit $\Omega(s(v))$ pro všechny nevyvážené vrcholy a nemusíme nic ušetřit.

Úloha 3 (Hloubka splay stromů nemusí být vždy logaritmická)
Navrhněte posloupnost operací, která vytvoří splay strom s lineární hloubkou a to jak pro naivní splay tak pro správný splay.

Řešení
Postupně pustíme operace $\textsc{Splay}(1)$, $\textsc{Splay}(2)$,…, $\textsc{Splay}(n)$, nebo místo Splayů Findy.
Alternativně můžeme takhle Insertovat do prázdného stromu.

Úloha 4 (Naivita se ne vždy vyplácí)
Co se pokazí, když operaci Splay implementujeme naivně, tedy jen pomocí jednoduchých rotací jedné hrany? (Tedy implementace naivního Splay je „dokud x není kořen, Rotate(x)".)

Řešení
Pokud budeme dvakrát za sebou volat $\textsc{Splay}(1)$,…,$\textsc{Splay}(n)$, tak nejdříve vytvoříme levou cestu, a potom s jednoduchými operacemi budeme stavět druhou levou cestu nad první, takže celková cena bude $\mathcal{O}(n^2)$.

Úloha 5 (Splay stromy mají potenciál)
Ujistěte se, že chápete, jak je definovaný potenciál ve splay stromech a že rozumíte hlavním myšlenkám analýzy amortizované složitosti splaye.

a) Jak je definován potenciál splay stromu?

b) Jaký je potenciál perfektně vyváženého stromu? (Stačí nám rozumný horní a dolní odhad, pro jednoduchost předpokládejte $n = 2^k - 1$.)

c) Jaký je potenciál cesty? (Opět stačí rozumné odhady.)

d) Jaká je amortizovaná cena rotace (dvojité a jednoduché)?

e) Jaká je amortizovaná cena celého splaye a jak plyne z amortizovaných cen rotace?

f) Jaká je *reálná* cena celého splaye (a v jakých jednotkách ji vlastně počítáme)?

Řešení a) Označíme si $T(v)$ podstrom s kořenem v, velikost $s(v) = |T(v)|$, hodnost $r(v) = \log_2(s(v))$, potenciál pak je $\Phi = \sum_{v \in V} r(v)$, a (tradičně) značíme n jako počet vrcholů ve stromě.

b) Dvojitá $3r'(x) - 3r(x)$, kde r' je rank po, r je rank před, a jednoduchá $3r'(x) - 3r(x) + 1$.

c) Nejvýše $3r'(x) - 3r(x) + 1 \in \mathcal{O}(\log n)$ - tohle vyjde, protože součty amortizovaných cen rotací teleskopují.

d) Počet provedených rotací - v rotacích.

e) $\sum_{i=0}^{k-1} 2^i \cdot (\log_2(2^{k-i} - 1)) \leq \sum_{i=0}^{k-1} 2^i \cdot (\log_2(2^{k-i})) = \sum_{i=0}^{k-1} 2^i \cdot (k - i) = \sum_{\ell=1}^{k} \sum_{i=0}^{\ell-1} 2^i = \sum_{\ell=1}^{k} 2^\ell - 1 = 2^{k+1} - 1 - (k+1) = 2^{k+1} - k - 2 = 2(2^k - 1) - k = 2n - \log n$. Zároveň máme triviální spodní odhad $n/4$, protože úplný binární strom má $2^{k-2} \approx n/4$ vrcholů v úrovni těsně nad listy.

f) $\sum_{i=1}^{n} \log(i) = \log(n!) \geq \frac{n}{2} \log\left(\frac{n}{2}\right)$, zároveň můžeme odhadnout $\log(n!) \leq n \log n$, protože $n! \leq n^n$.

Bonusové úlohy

Úloha 6 (Potenciál pro následníka)
Na prvním cvičení jsme si ukázali, že použití $n-1$ operací následníka na libovolném BVS, když začneme ve vrcholu s nejmenším klíčem, má složitost $\mathcal{O}(n)$. Jak to můžeme dokázat pomocí potenciálu?

Řešení
Zadefinujeme si potenciál $\Phi(x)$ jako $h + s_L(x) - s_R(x)$ pro x aktuální vrchol, kde h je maximální hloubka stromu, $s_L(x)$ je počet levých synů na cestě z kořene do x a $s_R(x)$ je počet pravých synů na cestě z kořene do x. Speciálně nalezení nejlevějšho prvku trvá i amortizovaně jeho hloubku, ale to v analýze všech kroků nevadí. Podíváme se na jednotlivé případy použití operace následníka na x:

1. pokud x má pravého syna a y je následníkem x, pak $s_R(x) - s_R(y) = -1$, $s_L(y) - s_L(x) =$ hloubka vrcholu y pod pravým a tedy $\Phi(y) - \Phi(x)$ je řádově reálná cena operace,

2. pokud x pravého syna nemá, postupujeme po otcích, dokud nevystoupíme poprvé jako levý syn - v tom případě pro vrchol y, který je následníkem x, máme $s_R(x) - s_R(y) =$ počet hran, které jsme vystoupali nahoru $-$ 1 a $s_L(y) - s_L(x) = -1$, což je opět řádově reálná cena operace.

Zároveň snadno zpozorujeme, že hodnota potenciálu je nejvýše $2h$.
Speciálně tedy až na první operaci v čase h máme všechny ostatní operace amortizovaně v $\mathcal{O}(1)$, a tedy celková složitost je $\mathcal{O}(n + h) = \mathcal{O}(n)$, neboť $h \leq n$.

Úloha 7 (Pro fajnšmekry)
Proveďte přestavění BVS na perfektně vyvážený BVS v lineárním čase *a s konstantní pamětí.*

Řešení 1. ze stromu uděláme pravou cestu

2. vhodně pozkomprimujeme listy