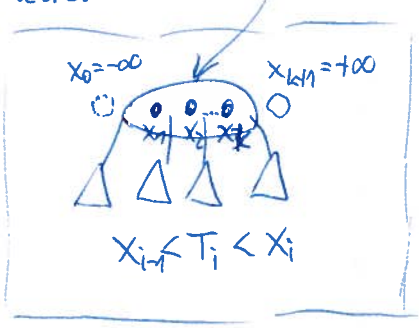
(a,b) - STROMY (Bayer, McCreight 1972)

Zobecníme vyhledávací stromy:

① Více klíčů ve vrcholech: k klíčů $\Rightarrow$ $k+1$ synů, klíče oddělují podstromy

② přidáme externí vrcholy (vnější): na místech NULL-pointerů, odpovídají intervalům mezi klíči

↳ často pouze myšlené



Df: (a,b)-strom pro $a\geq 2$, $b\geq 2a-1$ je obecný vyhl. strom, v němž platí:

① Kořen má 2 až b synů, ostatní vnitřní vrcholy a až b synů.

② Všechny vnější vrcholy jsou stejně hluboko.

↳ příklady: (2,3)-strom, (2,4)-strom

Varianty definice: Data pouze v listech, vnitřní vrcholy obsahují "navigační" kopie klíčů

Lemma: Hloubka (výška?) (a,b)-stromu s n vrcholy je nejvýše $1+\log_a \frac{n+1}{2}$. $\leq \log_a n$

$O(\log_a n)$ nezávisle na a

Dk: Počítejme, kolik nejméně klíčů obsahuje strom výšky h:

$$\underbrace{1}_{\text{kořen}}+\underbrace{(a-1)\cdot 2}_{\text{1. hladina}}+\underbrace{(a-1)\cdot 2a}_{\text{2. hladina}}+\dots+\underbrace{(a-1)\cdot 2a^{h-2}}_{(h-1)\text{-ní}}+\underbrace{0}_{\text{na }h\text{-té jsou listy}} =$$

$$= 1+2(a-1)\cdot\frac{a^{h-1}-1}{a-1} = 1+2\left(a^{h-1}-1\right) = 2a^{h-1}-1.$$

[analogicky: výška $\geq \log_b(n+1)$]

Find: jako v BVS ... $O(\log_a n)$ hladin, na každé $O(\log b)$ práce $\rightarrow O\left(\frac{\log b}{\log a}\cdot\log n\right)$

$\downarrow$ typicky $b\approx a$

$O(\log n)$. (z teorie informace: lépe to nejde)

Insert:

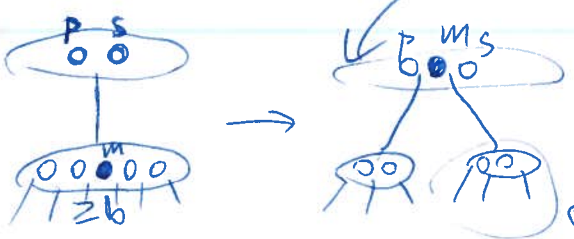
přidávám do nejhlubšího vnitřního vrcholu na cestě k přísluš. vnějšímu

↓

přibude klíč + vnější vrchol

↓

Pokud už je klíčů příliš mnoho ($\geq b$), štěpíme vrchol

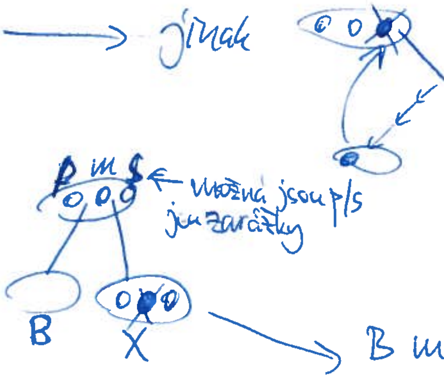


← zde přibude klíč ⇒ totéž o patro výš atd., při nejhorším štěpíme kořen?

← Nebudou příliš malé? Min. ok: $2(a-1)+1 = 2a-1 \leq b$ ✓

Čas: $O(b\cdot\log_a n) = O\left(\frac{b}{\log a}\cdot\log n\right)$.

Delete: BÚNO mažeme klíč z listu ⟶ jinak nahradíme minimem pravého podstromu



- smažeme klíč + vnější vrchol
- klíčů stále dost ⇒ hotovo
- jinak se podíváme na bratra: (BÚNO levého)

pokud B má $\geq a$ klíčů → "rotace" $\max(B) \to m \to \min(X)$

B má $a-1$ klíčů, my máme $a-2$
↓
sloučíme X s B, vytáhneme separující klíč z otce
↓
nový vrchol má #klíčů $= (a-1)+(a-2)+1 = 2a-2 \leq b-1$ ✓ OK
↓
delete o patro výš atd.
↓
možná smažeme kořen.

- Čas: $O(b \cdot \log_a n) = O\left(\frac{b}{\log a} \cdot \log n\right)$

Volba a,b:

- Nemá smysl volit $b \gg a$ … Find $O(\log n)$, Ins/Del $O\left(\frac{a}{\log a} \cdot \log n\right)$
  ↳ obvykle $b = 2a-1$ nebo $b = 2a$
- Pro interní paměť chceme malé a
- Pro paměť s blokovou strukturou (disk, cache) volíme parametry podle velikosti bloků.
  [často používáno v DB a FS pod názvem B-stromy]
- Modifikace: přerozdělují se sousedy, zaplnění $\geq \frac{2}{3}$ } B^+-stromy

Amortizace

Věta: Posloupnost m Insertů na zprvu prázdném stromu provede $O(m)$ změn vrcholů.

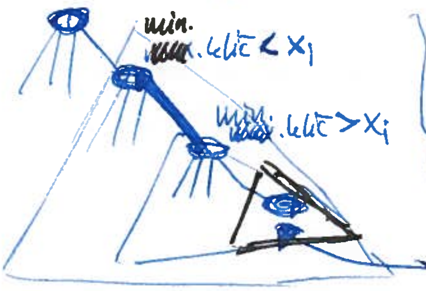
Dk: #změn na Insert $= O(1) + \underbrace{\#\text{štěpení}}_{=\#\text{vzniklých vnitřních vrcholů}}$

Celkem vznikne nejvýše m vnitřních vrcholů ⇒ celkem max. m štěpení.

Aplikace: A-sort (třídicí alg. pro částečně setříděná data) [Guibas et al. 1977]

↳ míra neuspořádanosti: $\underbrace{\#\text{inverzí}}_{I}$ $(\#\{(i,j): 1 \leq i < j \leq n, a_i > a_j\})$

- Postupně Insert-ujeme hodnoty do (2,4)-stromu [BÚNO všechny hodnoty různé]
- Pamatujeme si max. prvek (v "pravém dolním" vrcholu stromu), hledáme od něj směrem nahoru:



} insert do podstromu výšky v

→ $v \in O(\log(\underbrace{\#\text{inverzí s } x_i}_{I_i}))$

kdyby $I_i = 0$, dodefinujeme $\log = 0$

podstrom výšky $v-2$: všechny jeho klíče jsou v inverzi s x_i (~~nebo rovné x_i~~)

- Celkem hledáním strávíme čas

$$O\left(\sum_i (1+\log(I_i))\right) \le O\left(n + \underbrace{\sum_i \log(I_i)}\right) \le O(n + n\log I)$$

$$\log \prod_i (I_i) = n\cdot \log \sqrt[n]{\prod_i (I_i)} \overset{AG}{\le} n\cdot\log \frac{\sum_i (I_i)}{n} \le n\cdot \log I$$

Celkem tedy $O(n + n\log I)$

lze jednodušeji: jelikož $\forall i\ I_i \le I$, platí $\sum_i \log I_i \le \sum_i \log I = n\log I$.

- Všechny modifikace stromu trvají $O(n)$.

Silnější amortizace pro $b=2a$

← na rozdíl od $b=2a-1$ má "hysterezi"

Věta: Posloupnost m operací Insert+Delete na zpočátku prázdném $(a,2a)$-stromu způsobí $O(m)$ změn vrcholů.

Dk: Zavedeme potenciál $\Phi := \sum_{v \ne \text{kořen}} f(\#\text{klíčů ve vrcholu } v)$ ← započítávám i dočasné stavy mimo povolený rozsah

kořen uvažujeme zvlášť $\Rightarrow O(1)$ změn na Insert/Delete

- Co chceme od funkce f?

① $|f(i) - f(i+1)| \le c$ (běžné změny moc nemění Φ)

② $\underbrace{f(2a)}_{\text{starý vrchol}} \ge \underbrace{f(a)+f(a-1)}_{\text{nové vrcholy}} + \underbrace{c}_{\text{nový klíč v otci}} + \underbrace{1}_{\text{skutečná cena operace}}$ (štěpení je zdarma)

③ $\underbrace{f(a-2)+f(a-1)}_{\text{staré vrcholy}} \ge \underbrace{f(2a-2)}_{\text{nový vrchol}} + \underbrace{c}_{\text{smazání klíče v otci}} + \underbrace{1}_{\text{skutečná cena operace}}$ (slučování je zdarma)

tohle stačí, protože Ins/Del provede nějaká ②+③ a $O(1)\times$ ①

- Jak bude vypadat f?

$a \le 2a-2$ ✓

dočasné $a-2$	$a-1$	a	$\dots$	$2a-2$	$2a-1$	$2a$ dočasné
2	1	0	$\dots$	0	2	4

Ověříme podmínky:

① $c=2$

② $4 \ge 0+1+2+1 = 4$ ✓

③ $2+1 \ge 0+2+1$ ✓

Štěpení shora dolů

- v $(a,2a)$-stromu mohu štěpit/slučovat "preventivně" při cestě shora dolů

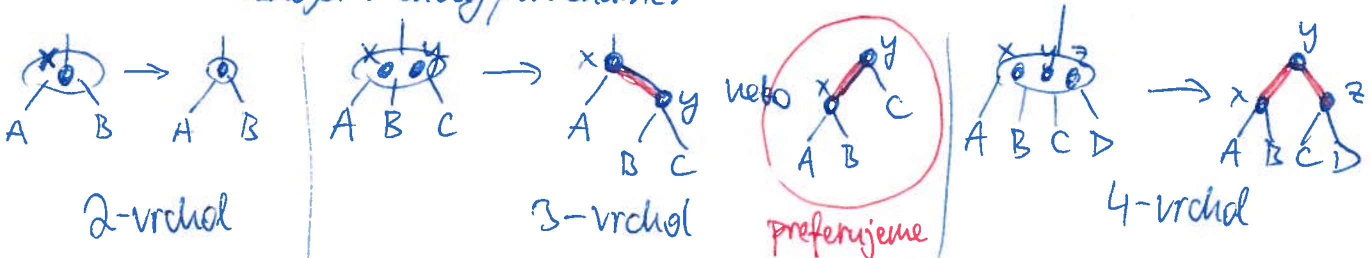
Štěpení: $2a-1$ klíčů $\to (a-1)+(a-1)+1$ — určitě se vejde o patro výš } Invariant: aktuální vrchol má méně než $2a-1$ klíčů

Slučování: $(a-1)+(a-1)+1 \to 2a-1$ — určitě můžeme smazat } Inv: aktuální má $\ge a$ klíčů

- Výhoda: při paralelním přístupu stačí zamykat 2 vrcholy pod sebou, nehrozí deadlocky ← nebo otec+syn+jiný syn

Červeno-Černé (RB)-stromy [Guibas, Sedgewick 1978]

Myšlenka: Vrcholy (2,4)-stromu zakódujeme do konfigurací vrcholů BVS
Červené hrany spojují vrcholy uvnitř konfigurace,
černé hrany vedou mezi konfiguracemi. ← to tedy jsou hrany (2,4)-stromu
Vnější vrcholy ponecháme.



Překlad axiomů (2,4)-stromů do řeči (LL)RB-stromů:

① Hrany obarveny červeně/černě,] barvu hrany ukládáme do spodního vrcholu

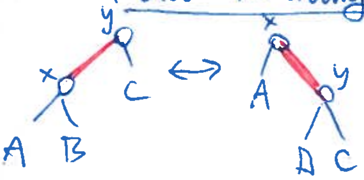
červené axiomy:
- a) nejsou 2 červené nad sebou
- b) vede-li z vrcholu dolů 1 červená, je to levá] "LL" podmínka

černé axiomy:
- c) hrany do listů jsou černé
- ② Na všech cestách kořen–vnější vrchol je stejný # černých hran.

Důsledky:
- bijekce (2,4) ⇔ LLRB
- hloubka LLRB-stromu ≤ 2 · hloubka (2,4)-stromu ≤ 2 log n.

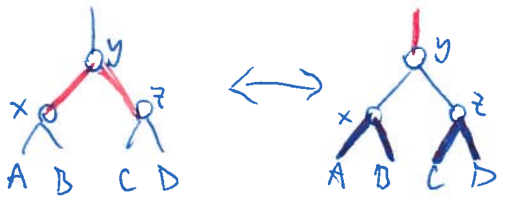
Vyvažovací operace:

Rotace R hrany



Zachová uspořádání + B axiomy
R axiomy může rozbít

Přebarvení 4-vrcholu



Odpovídá štěpení 4 → 2+2

Zachová uspořádání a B axiomy,
může porušit R axiomy o patro výš

Závěrem: Existují varianty, které zaručují O(1) rotací při Ins/Del v nejhorším případě

Insert:

① Hledáme místo, kam vložit, obvyklým způsobem.
Při tom štěpíme 4-vrcholy...
- aktuální vrchol není 4-vrchol
- nad námi mohou vznikat nekorektní 3-vrcholy a 4-vrcholy.

② Připojíme  červenou hranou namísto  → B axiomy OK,
možná vyrobíme nek 3/4-vrchol

③ Zdola nahoru opravujeme rotacemi R axiomy: 

trvá O(log n)