NB: АВЛ – Адельсон-Вельский и Ландис

Splay-tree

[Sleator, Tarjan]

Реализуют операцию Splay(x), которая перестраивает дерево так, что x – корень.

Лемма 1: Стоимость операции Splay(x) = $O(h)$

Лемма 2: ∃ Φ – некоторая функция потенциала, опред. на деревьях (бинарных), такая что

а) $\Phi(T) = O(n \log n)$

б) Учётная стоимость Splay(x) отн Φ будет $O(\log n)$

Учётная стоимость

∃ c_i – реальные стоимости операций

Определим $\tilde{c}_i = c_i + \Delta\Phi$ – учётная стоимость относительно потенциала Φ.

$$\sum_i c_i = \sum \tilde{c}_i - \Delta\Phi \underset{\Delta\Phi \geq 0}{\Longrightarrow} \sum_i c_i \leq \sum_i \hat{c}_i$$

Для Splay(x): $\sum_i c_i = \sum_i \hat{c}_i + O(n \log n) = O(n \log n + n \log n)$

m – # операций. $\Rightarrow$ при $m \geq n$

$m = \Omega(n)$

$\Downarrow$

Для m операций Splay() нам потребуется $O(m \log n)$ операций.

$\Downarrow$

В среднем Splay требует $O(\log n)$ операций

Find(x):

"найди x в дереве" // как в AVL

Splay(x)

Insert(x):

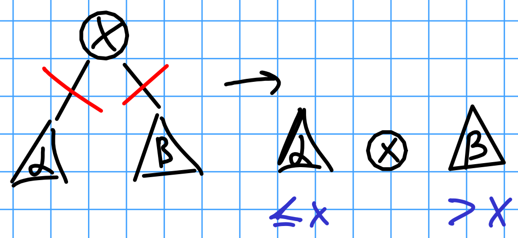
"вставь x в дерево" // как в AVL

Splay(x)

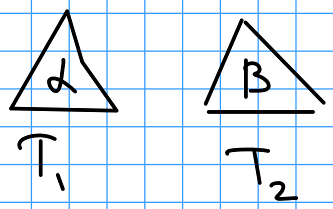
Split(x):

Find(x)

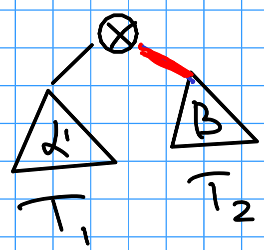
return $\{left(x), x, right(x)\}$



Merge(T_1, T_2): // $\forall x \in T_1 \ \forall y \in T_2 \ \ x \le y$



$\longrightarrow$ Splay(max T_1)



Remove(x):

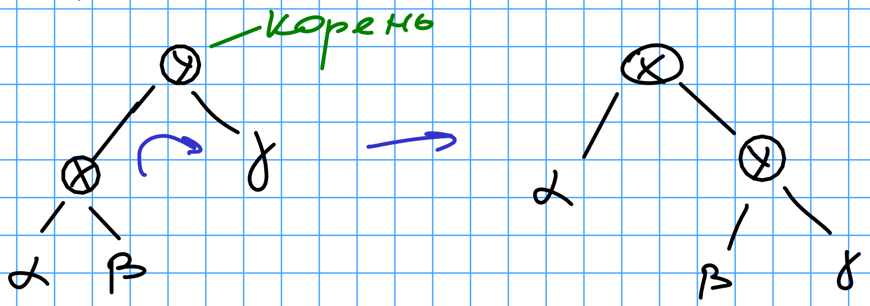
$\{T_1, x, T_2\}$ = Split(x)

return Merge(T_1, T_2)

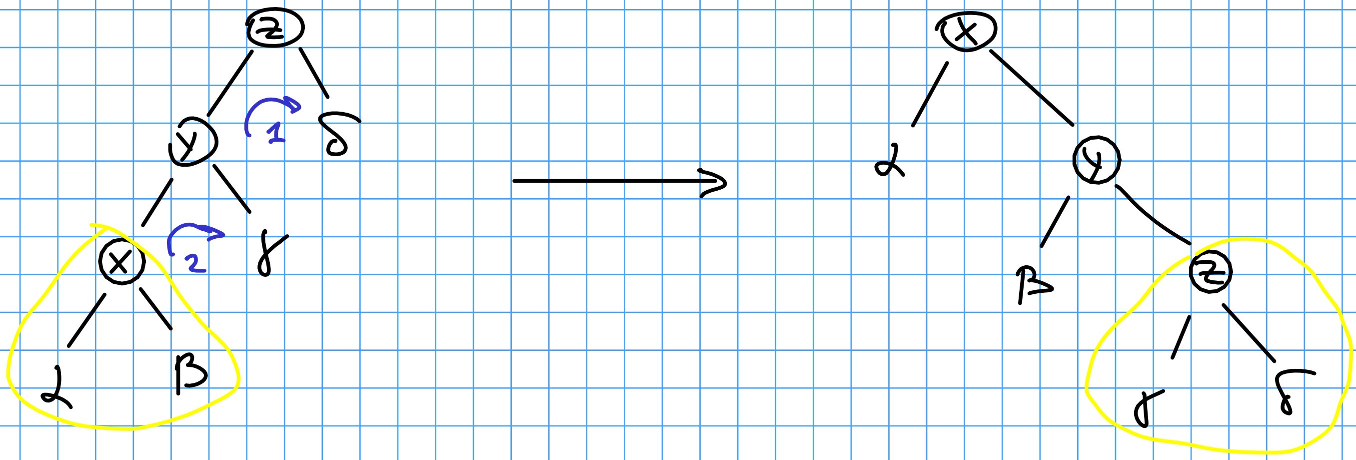
Утв: Все операции над Splay-деревом "окончены" вызовами Splay() $\Rightarrow$ можно оценивать только стоимость Splay()

Устройство Splay:

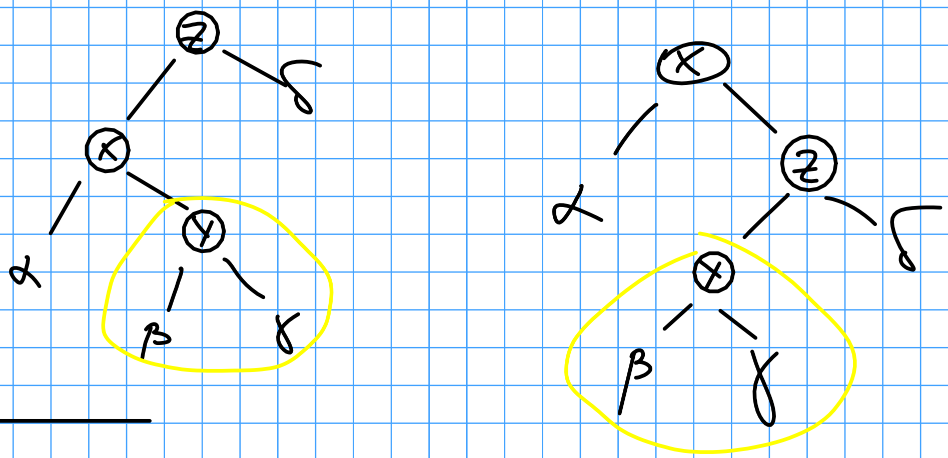
a) zig



б) zig zig

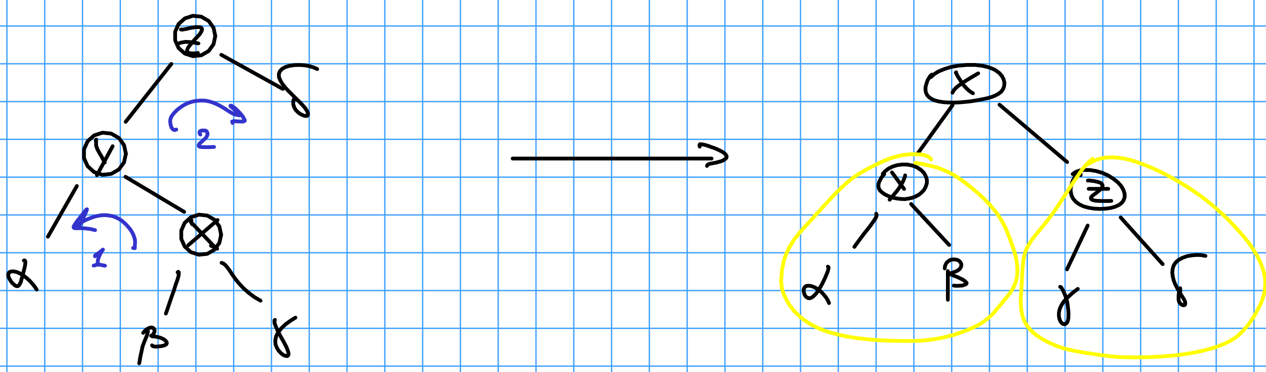


Замечание: сначала верхний поворот, а потом — нижний.



так не делать! неправильно!

в) zig zag



Следствие: Лемма 1 доказана.

Доказательство Леммы 2:

] $w(x)$ — это # вершин в поддереве x

$\equiv \quad \Phi(x) = \lfloor \log_2 w(x) \rfloor$

$$\Phi(T) = \sum_{x \in T} \Phi(x) \Rightarrow \Phi(T) = O(n \log n)$$

(пункт а) леммы 2)

Утв: а) Стоимость zig шага $\leq 3(\Phi'(x) - \Phi(x)) + 1$,

б) стоимость zigzig и zigzag шагов $\leq 3(\Phi'(x) - \Phi(x))$

Φ − потенциал до шага, а Φ' − после

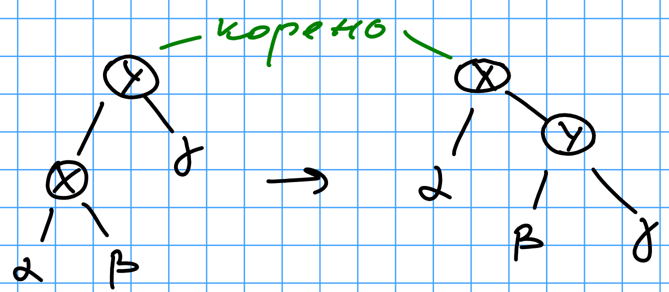
Следствие:

Стоимость операции $Splay(x) \leq$

$$\leq 3(\underset{\text{корень}}{\Phi'(x)} - \Phi(x)) + 1 \overset{?}{=} O(\log n)$$

т.к. $\Phi(x) = O(\log n)$

Док-во а):



$$\tilde{C} = C + \Delta\Phi = 1 + \Phi'(y) + \underbrace{\cancel{\Phi'(x)}}_{=\Phi(y)} - \Phi(x) - \cancel{\Phi(y)} =$$

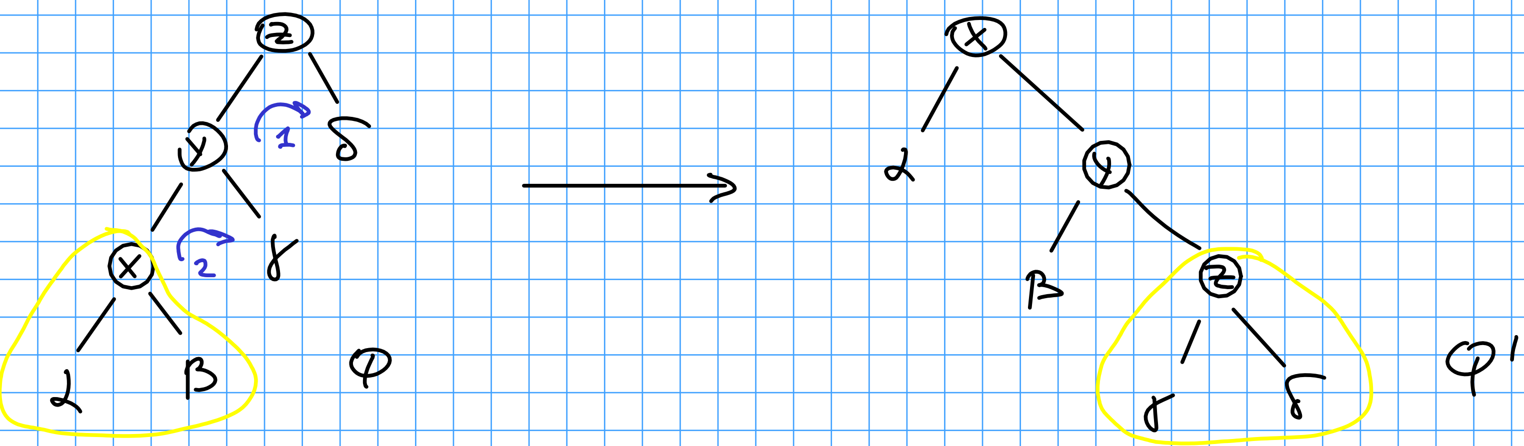
$$= 1 + \underbrace{\Phi'(y)}_{\leq \Phi'(x)} - \Phi(x) \leq$$

$$\leq 1 + (\Phi'(x) - \Phi(x)) \leq$$

$$\leq 1 + 3(\Phi'(x) - \Phi(x))$$

т.к. $\Phi'(x) \geq \Phi(x)$ (≥ 0)

Док-во б):



$$\tilde{C} = C + \Delta\Phi = 1 + \cancel{\Phi'(x)} + \underset{\le \Phi'(x)}{\Phi'(y)} + \underset{\le \Phi'(x)}{\Phi'(z)} - \Phi(x) - \underset{\ge \Phi(x)}{\Phi(y)} - \underset{= \Phi'(x)}{\cancel{\Phi(z)}} \le$$

$$\le 1 + 2\big(\underbrace{\Phi'(x) - \Phi(x)}_{\Delta}\big) \underset{\Delta \ge 1}{\le} 3\big(\Phi'(x) - \Phi(x)\big)$$

Утв: $\Delta \ge 1$.

Докажем от обратного:

$$(*) \begin{cases} \Phi'(x) = \Phi(x) \\ \Phi'(x) = \Phi'(y) = \Phi'(z) \\ \Phi(x) = \Phi(y) \end{cases} \Rightarrow \Phi'(x) = \Phi'(y) = \Phi'(z) = \\ = \Phi(x) = \Phi(y) = \Phi(z)$$

Пусть w', w — весовые функции после и до шага.

Пусть k: $2^{k+1} > w'(x) \ge 2^k$, т.е. $\Phi'(x) = k$ — x, y и z

$$2^{k+1} > w'(x) = w'(\alpha) + w'(\beta) + w'(\gamma) + w'(\delta) + 3 =$$

$$= \underbrace{(w(\alpha) + w(\beta) + 1)}_{w(x)} + \underbrace{(w'(\gamma) + w'(\delta) + 1)}_{w'(z)} \ge$$

$$\ge 2^k + 2^k + 1 = 2^{k+1} + 1$$

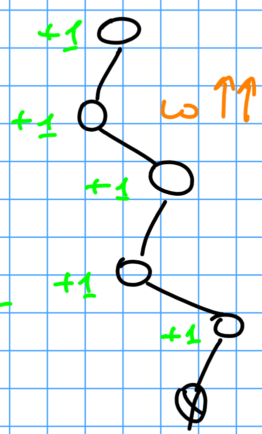
противоречие

$\Rightarrow$ $(*)$ неверно.

см. MTF

Замечание 1:

Insert изменяет $\Phi(T)$ на $O(\log n)$



$$\Delta\Phi \le (\#\text{ степеней } 2 \le n) = \log_2 n$$

$2^k - 1 \to 2^k$

Замечание 2:

Remove изменяет $\Phi(T)$ на $O(\log n)$

$\Delta\Phi = \Delta\Phi(Split) + \Delta\Phi(Merge)$

$\left.\begin{array}{l}\Delta\Phi(Split) = -\lfloor \log_2 n \rfloor \\ \Delta\Phi(Merge) \leq \lfloor \log_2 n \rfloor\end{array}\right| \Rightarrow \Delta\Phi = O(\log n)$