

Декартово дерево (cartesian tree)

tree (tree + heap)
гура (дерево + гура)
герамида (дерево + пирамида)
курево (кура + дерево)

В n вершине хранятся
числа k и приоритет p .

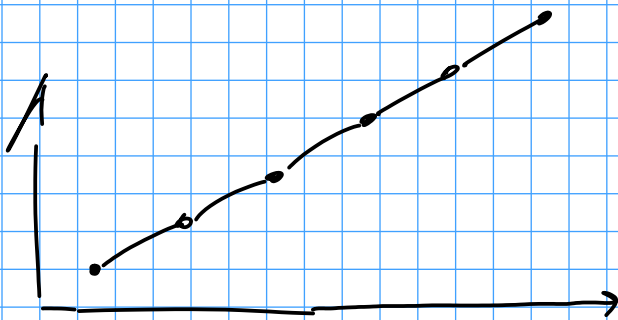
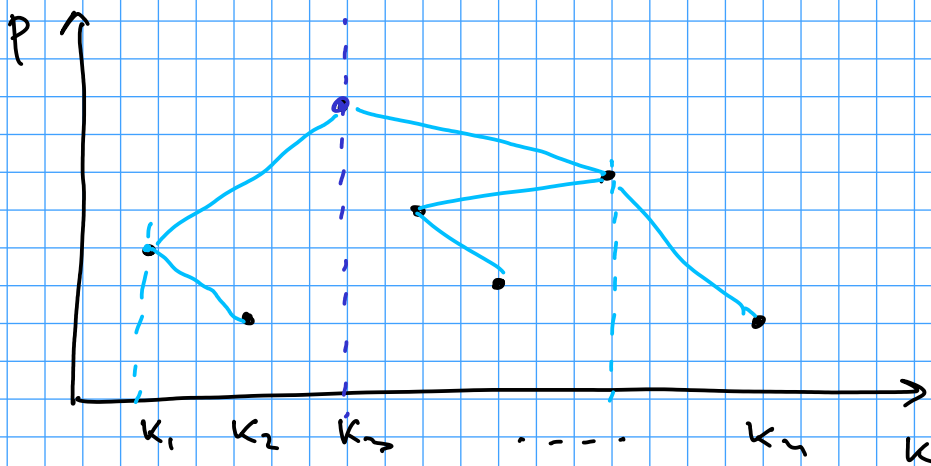
= Декартово дерево

а) дерево по k

б) гура по p .

Существование

Вход: $(k_1, p_1), (k_2, p_2), \dots, (k_n, p_n)$ $k_i \leq k_{i+1}$



Худший случай $O(n^2)$

Сбалансированная гура

Вход: k_1, k_2, \dots, k_n

Случайные: $p_1, p_2, p_3, \dots, p_n$

Уто:

В среднем высота гуры будет $O(\log n)$

Реш.во: Сам. анализ гайдлайн Quick Sort.

Нужно Φ_i - гана управоро цепи на масе i

$$\tilde{C}_i = C_i + \Delta \Phi_i \quad \Phi_0 = 0$$

$$\sum C_i = \sum \tilde{C}_i - (\Phi_n - \Phi_0) \approx \sum \tilde{C}_i$$

≥ 0 0

$$\tilde{C}_i = \underbrace{C_i}_{\text{кон-во матов}} + \underbrace{\Phi_i - \Phi_{i-1}}_{\text{кон-во матов} - 1} = \cancel{k} + (\cancel{\Phi_{i-1}} - \cancel{k+1}) - \cancel{\Phi_{i-1}} = 1$$

$$\sum C_i = O(n)$$

Операция:

Find - перебор

Split ($k, v = \text{root}$):

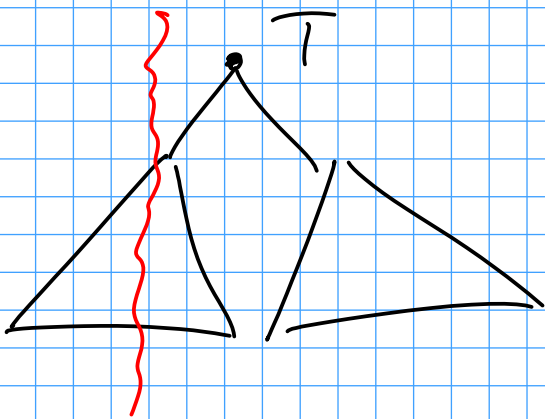
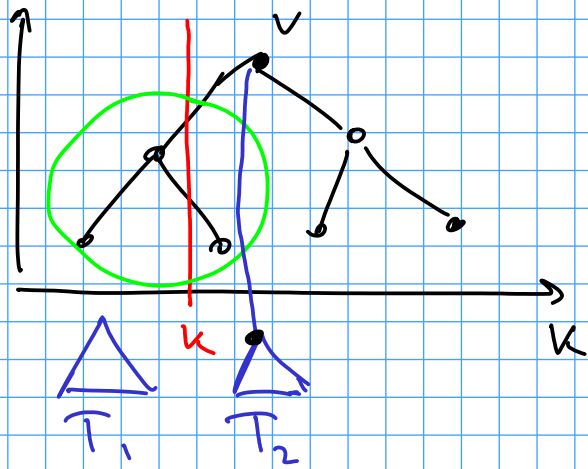
if $k(v) > k$:

$T_1, T_2 = \text{Split}(k, v.\text{left})$

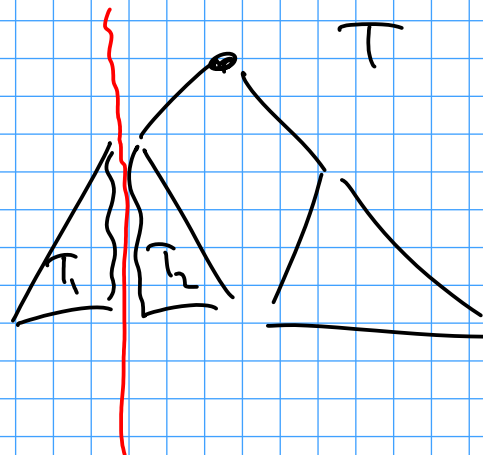
$v.\text{left} = T_2$

return (T_1, T)

else...



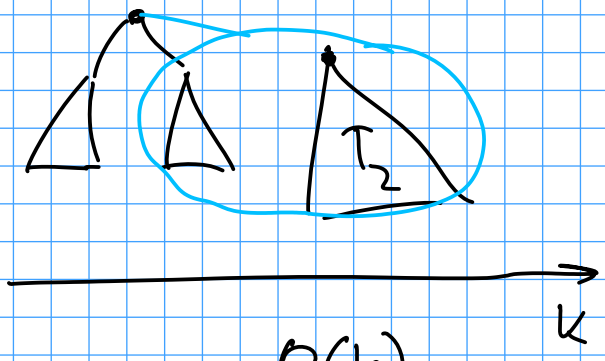
→



$O(n)$

Merge (T_1, T_2): // Все known $T_1 <$
 if $p(\text{root}(T_1)) > p(\text{root}(T_2))$:
 then known T_2

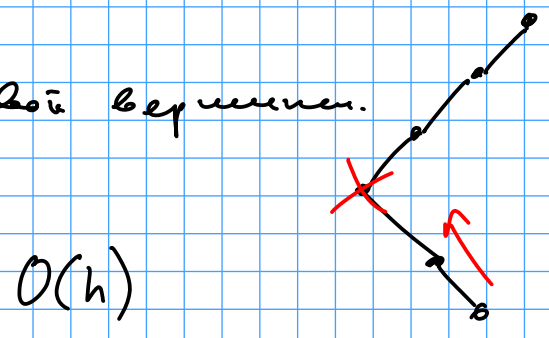
$T_1.\text{right} =$
 Merge ($T_1.\text{right}, T_2$)
 else ...



Insert ($T, (k, p = \text{rand}())$) $O(h)$

$T_1, T_2 = \text{Split}(k, T)$ $O(h)$
 return Merge (Merge ($T_1, (k, p)$), T_2)

Erase (T, k)
 $T_1, T_2 = \text{Split}(k, T)$
 $T_2' = T_2$ say canoni netoi bezumen.
 return Merge (T_1, T_2')

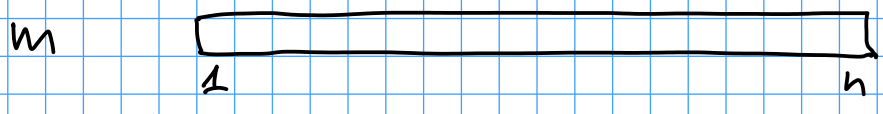


$O(h)$

known: k_1, k_2, \dots, k_n

Пример: 1 2 ... n

RandomShuffle ($[1, 2, \dots, n]$)



$n^n \neq n!$

for $i = 1$ to n :
 $k = \text{rand}(i, n)$
 swap ($m[k], m[i]$)

\neq непер. с вер-ю $\frac{1}{n!}$