

Vaje 8

Thursday, November 29, 2018 8:40 AM

Rang

Imamo n števil. Želimo učinkovito implementirati:

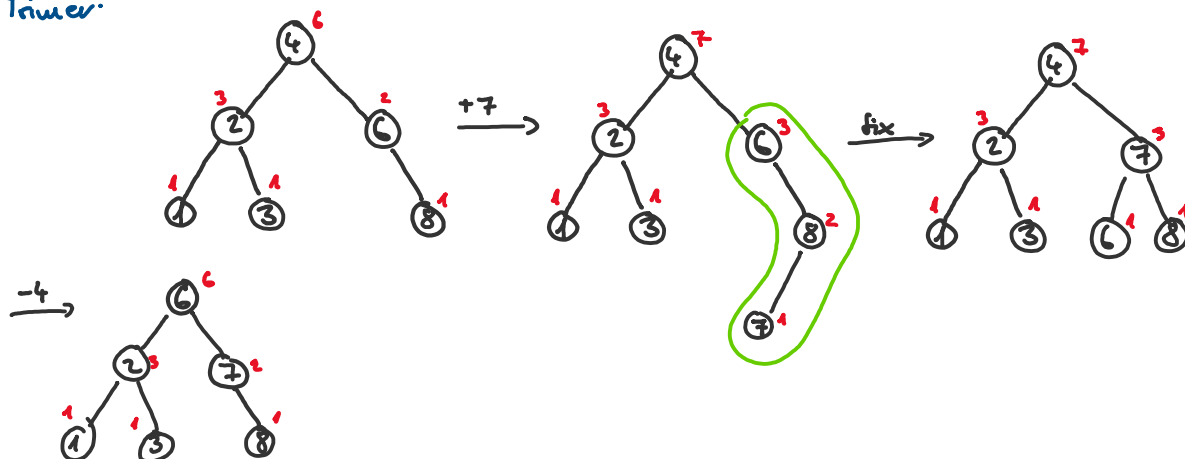
$\text{Rank}(S, x) \rightarrow i$ vrne kateri po vrsti je element x v S (če bi imeli po velikosti)

$\text{Select}(S, i) \rightarrow x$ vrne i -ti element po vrsti v S (če bi S imeli po velikosti)

Implementacija

z AVL drevesom: vsakemu vozlišču dodamo informacijo o številu naslednikov $+1$ (vključno s samim seboj).

Primer:



Naloge:

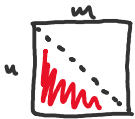
1. Dan je element x v uravnoteženem binarnem iskalnem drevesu z n vozlišči in dodatno informacijo o rangju (kombinirana struktura). Kako lahko v času $O(\log n)$ poiščemo j -tega naslednika elementa x ?

$k\text{-Rang}(S, x)$

$\text{Select}(S, k+j)$

2. Naj bo A polje z n različnimi elementi. Če je $i < j$ in $A[i] > A[j]$, potem par (i, j) rečemo transpozicija. Opisi algoritem, ki z uporabo Ranga prešteje število vseh transp. v A v časi $O(n \log n)$.

	0	1	2	3	4	5	6	7
A	3	8	7	2	10	12	13	-1



$$\frac{u^2 - u}{2} = O(u^2)$$

A

0	1	2	3	4	5	6	7
3	8	7	2	10	12	5	-1

↑
j

$$Trans(j) = \{G_i; i < j \wedge A[i] > A[j]\}$$

$$|Trans(j)| = j - Rank(A[1..j], A[j])$$

$$\#transpozicij = \sum_{j=2}^n Trans(j)$$

Naši gradimo AVL drevo skupaj z informacijo o rangi:

• vstavimo $A[1]$ v prazno AVL drevo T

• za vsak index $j = 2, \dots, n \rightarrow O(\log n)$

• vstavimo $A[j]$ v drevo T , izračunamo rang elementa

$A[j]$ glede na T (glede na $A[1..j]$)

$$O(\log n) \quad \#trans = \#trans + (j - Rank(A[1..j], A[j]))$$

$$\Rightarrow Z(\log n) = O(\log n)$$

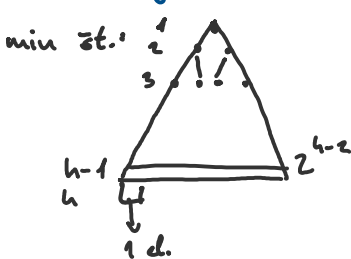
Vrste & prednostjo

Podatkovna struktura, ki omogoča dodajanje el., iskanje in brisanje najmanjšega elementa.

Binarna kopica

Levo poravnano binarno drevo, kjer za poljubno vozlišče velja, da so vsi njegovi nasledniki večji od njega.

(N.) Koliko je min in max št. elementov v kopici višine h ?



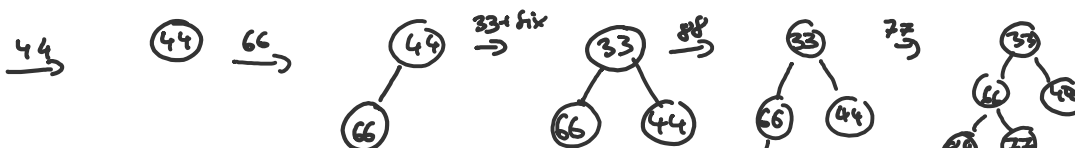
$$\text{min št.: } (2^{h-1} - 1) + 1 = 2^{h-1}$$

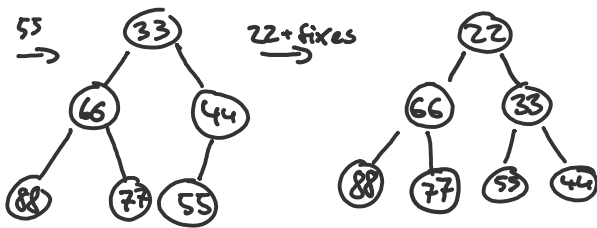
$$\sum_{i=0}^{h-2} 2^i = \frac{2^{h-1} - 1}{(2-1)} = 2^{h-1} - 1$$

$$\text{max št.: } 2^h - 1$$

(N.) Vstavi v binarno kopico naslednje elemente:

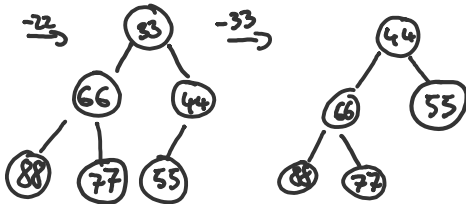
44, 66, 33, 88, 77, 55, 22



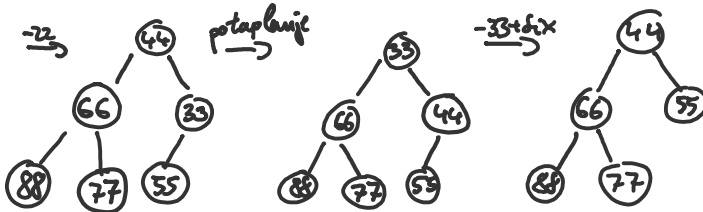


1. Duokrat zapored DeleteMin iz zadnje kopice.

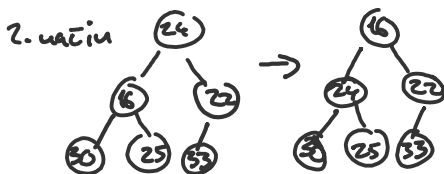
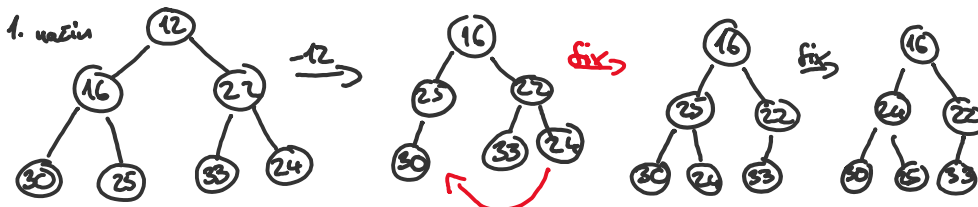
1. način



2. način



2. Izbriši min el. iz naslednje kopice:



3. Ali je vrsta s prednostjo slovar?

Ne, ne omogoča učinkovitega brisanja in iskanja el.!

Implicitno podana kopica:

indexi: 1... in ne 0... L otok

Če $A[i]$, ki predstavlja neko vozlišče v kopici, sta otroka $A[2i]$ in $A[2i+1]$ (D otrok) starša elementa $A[i]$ pa je $A[\lfloor \frac{i}{2} \rfloor]$.

1	2	3	4	5	6	7
12	16	22	30	25	33	24

4. Katere izmed naslednjih tabel predstavljajo kopico?

a)

5	7	10	8	15	12	16
---	---	----	---	----	----	----

b)

5	7	12	8	15	10	19
---	---	----	---	----	----	----

c)

3	2	10	8	15	18	20
---	---	----	---	----	----	----

