Sortări Avansate

Shellsort -> Div. et Impera pt. Insertion

-> împarte tabloul a în h subtablouri
pentru i de la 1 la h
sort. subtabloul a_i

sort. întregul tablou

h prea mic -> subtabl. prea mari
h prea mare -> prea multe subtabl.

---

27 15 83 5 40 13 78 99 18 20 16
↓

$\begin{pmatrix} 27 \\ 13 \\ 16 \end{pmatrix}$ 15 83 5 40
78 99 18 20

(5)

13 15 83 5 20
16 78 99 18 40
27

-> 13 15 83 5 20 - ——— - - --

13 15 83 5 20 16 78 99 18 40 27

5 20 16
78 99 18
40 27

(3)

H → 5 3 1

subtabele

5 15 16
13 20 18
40 27 83
78 99

---

5 15 16 13 20 18 40 27 83 78 99

5 13 15 16 18 20 27 40 78 83 99

(1)

H → descrescător (ultimul obligatoriu 1)

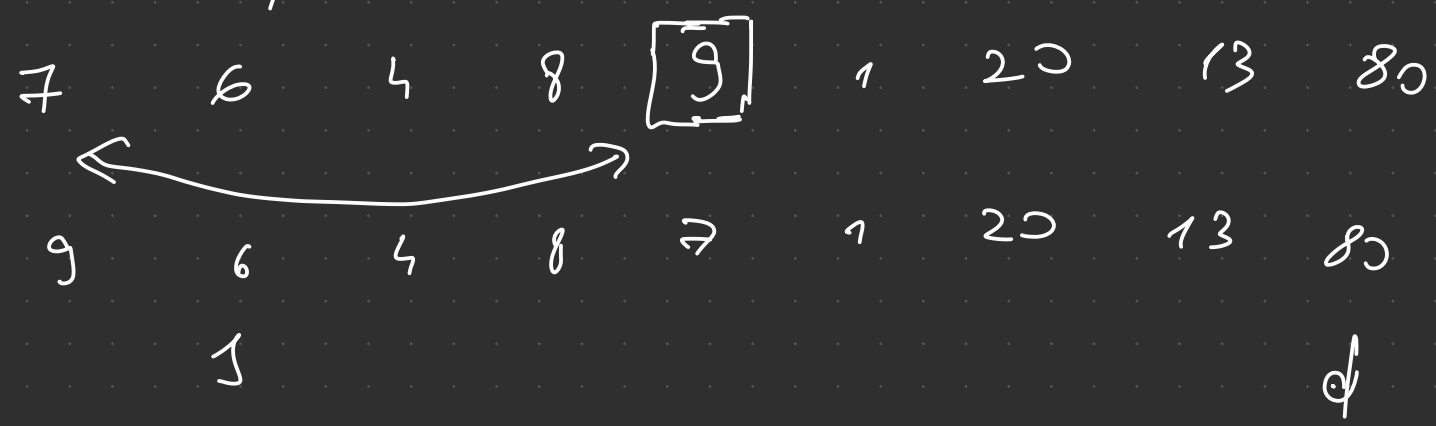
→ $O(n^{\frac{5}{3}}) < O(n^2)$

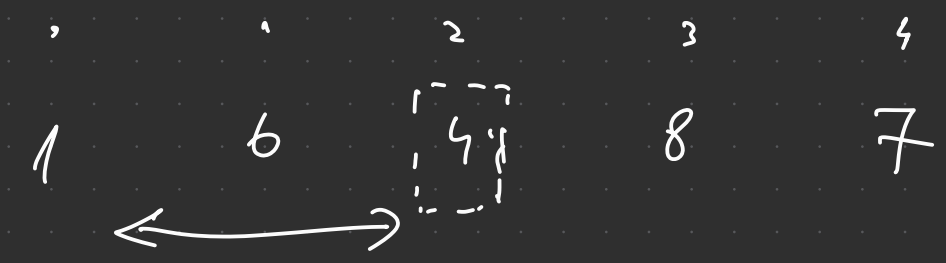
Quicksort

$O(n^2)$ când deg pivotul cel mai mic / mare

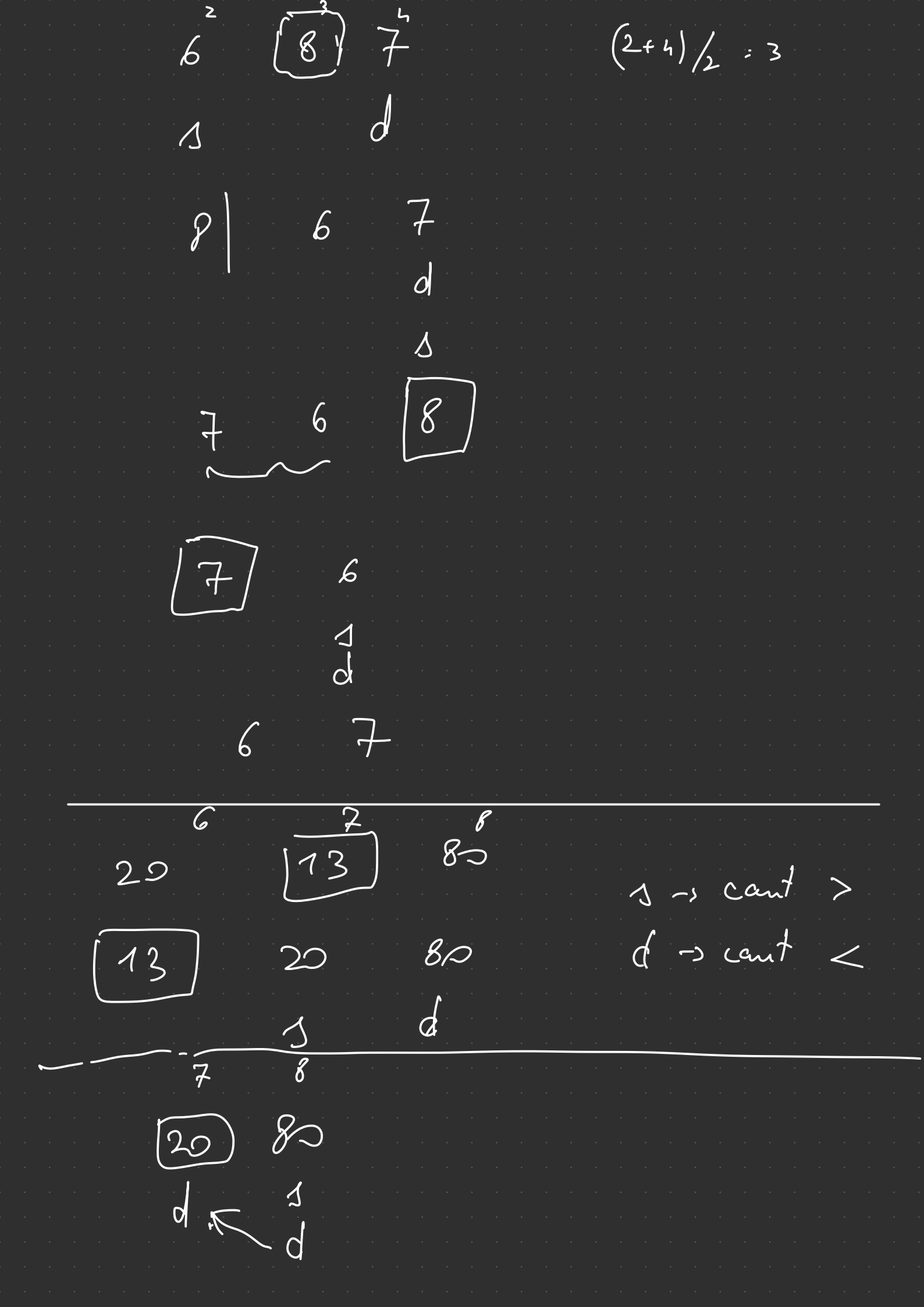
$O(n \log n)$ în general

I pivot median





1 4 6 8 7


(2+4)/2 : 3
s -> cant >
d -> cant <

- aleg pivot $(s+d)/2$ (devine primul element din subtablou)

$\to s \quad d$

$\longrightarrow \quad \longleftarrow$

$C_{min} = M_{min}$

$O(n) + 2T\left(\frac{n}{2}\right) + 4T\left(\frac{n}{4}\right) + \ldots$

$\downarrow$

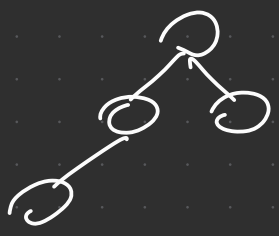
parcurgerea

$= O(n \log n)$

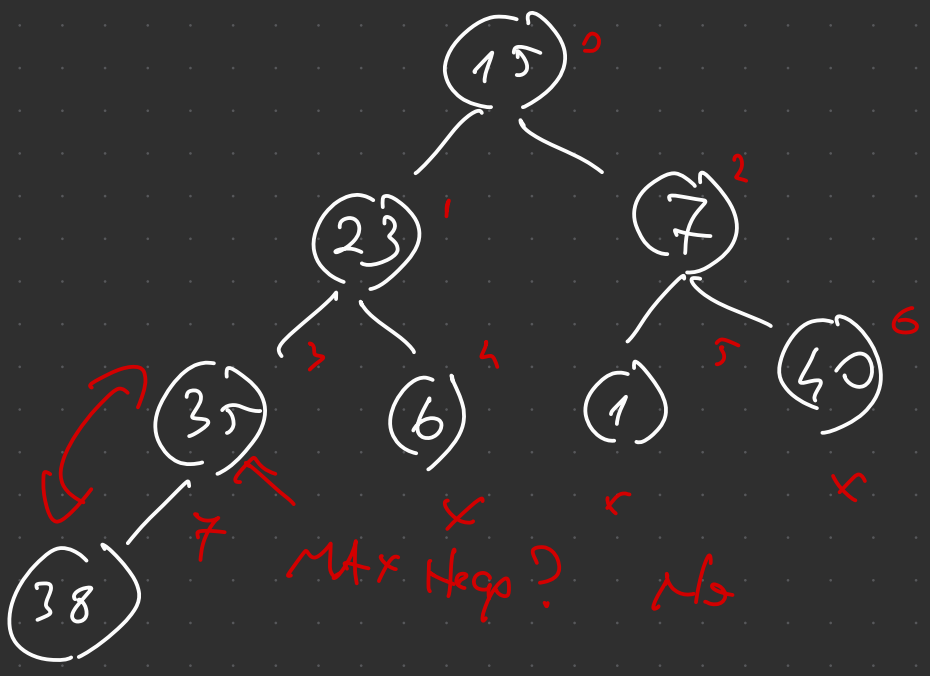
$C_{max} = M_{max} = O(n^2)$ pivot val $\begin{cases} max \\ min \end{cases}$

Heap Sort

-> nu folosește recursivitatea
-> se bazează pe selection sort
-> bazată pe un arbore binar



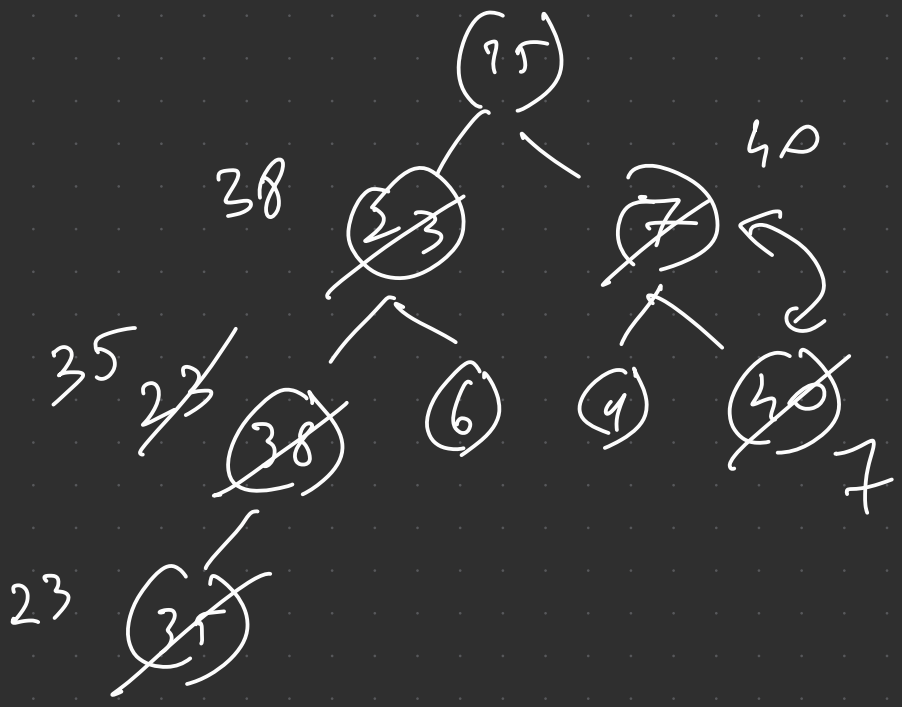
0	1	2	3	4	5	6	7
15	23	7	35	6	1	40	38



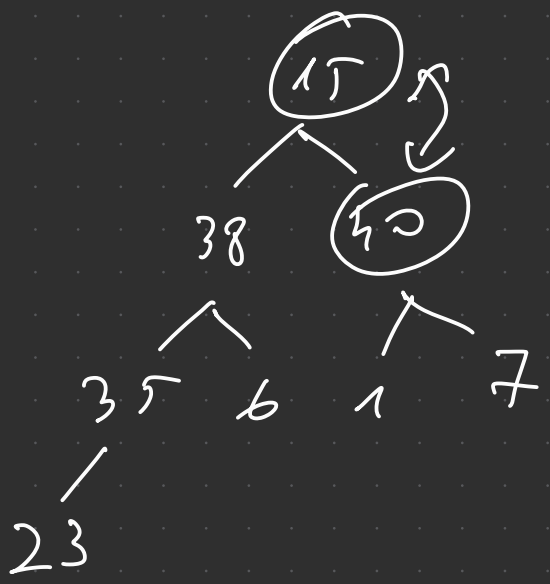
Heap - MAX Heap

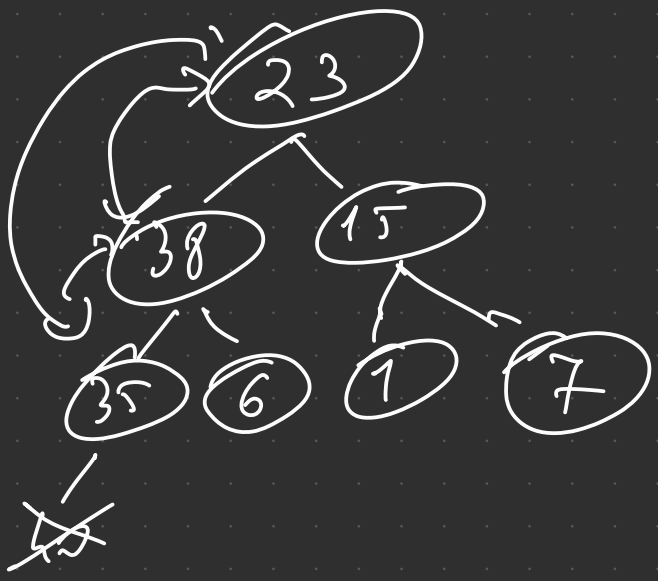
$$a[i] \geq a[2i+1]$$

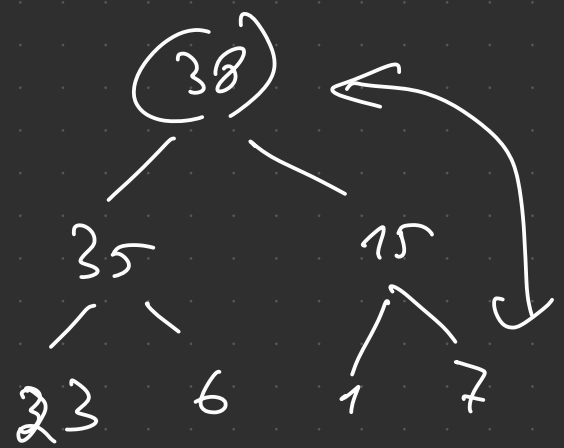
$$\text{si } a[i] \geq a[2i+2]$$

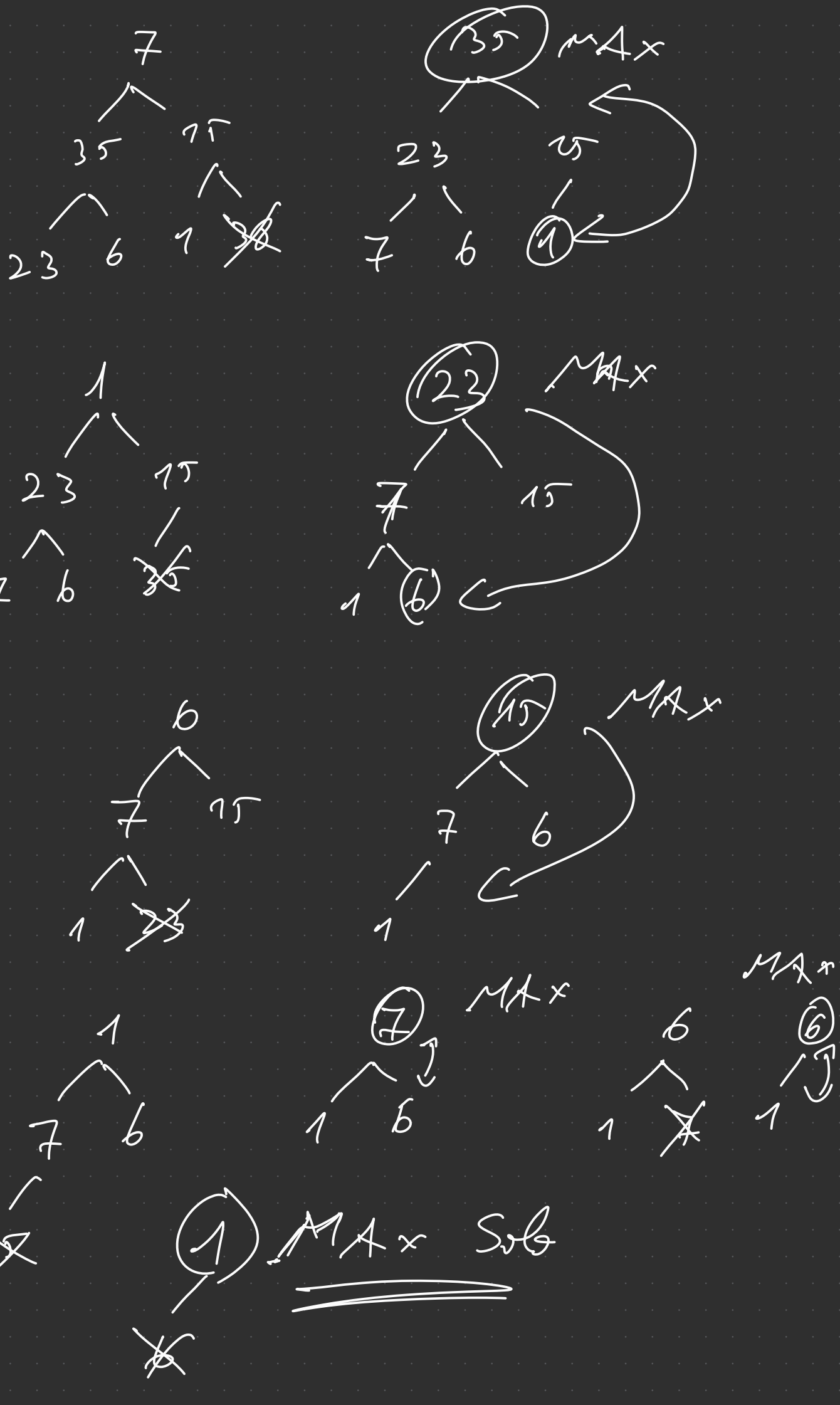


Nodul > fii → swap cu max dintre fii








7
35
15
23
6
1
35
MAx
23
15
7
6
1
1
23
15
7
6
23
MAx
7
15
1
6
6
7
15
1
23
15
MAx
7
6
1
7
6
1
7
MAx
1
6
6
MAx
1
6
1
1
MAx
Sub