INTRO

```
for ( i=1; i ≤ N; i++)
    for( j=i; j≤N; j+=i)
        f[j]++;
```

$$N + \frac{N}{2} + \frac{N}{3} + \ldots + \frac{N}{N} = N\left(1 + \frac{1}{2} + \ldots + \frac{1}{N}\right)$$

$$\simeq N \ln N \qquad (n \log n)$$

$$\boxed{\log_x y = \log_a y \cdot \log_x a}$$

Heap

S = multiset

$S = \emptyset$

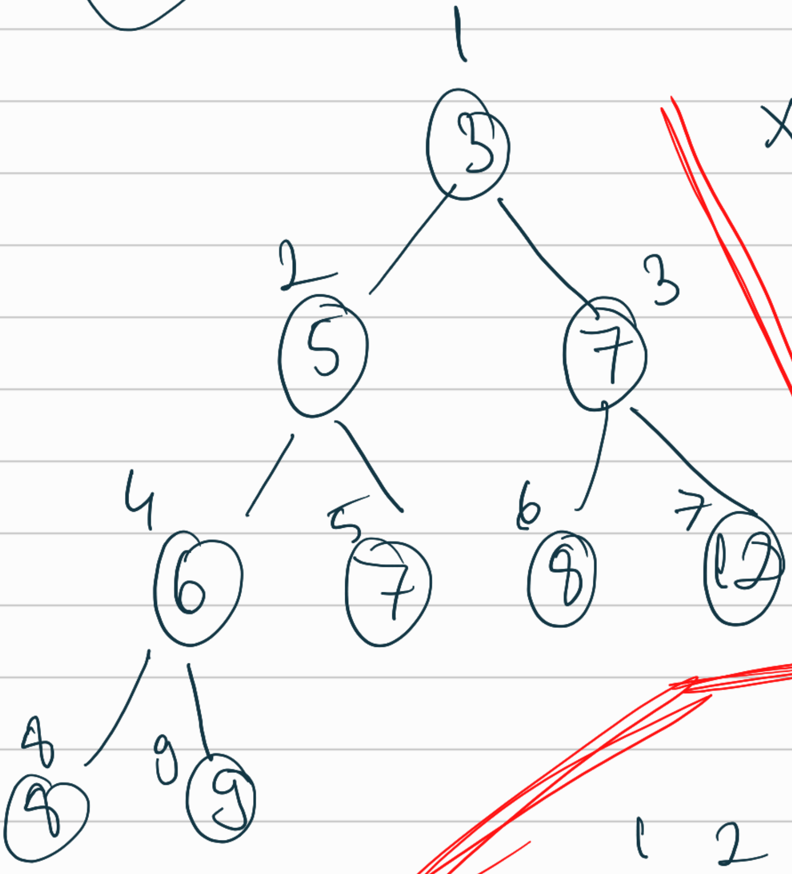
$\boxed{Q}$ ← operații (denumire dubioasă)

ADD $x \longrightarrow S \mathrel{+}= [x] \longrightarrow O(1)$ | $O(\log(|S|))$

GET_MIN → min(S) = ? $O(|S|)$ $O(1)$

POP_MIN → S -= [min(S)] $O(|S|)$ $O(\log(|S|))$

$O(Q^2)$ Pentru Vector
$O(Q \log Q)$ Pentru Heap
} Sortare



$x \to 2*x$

$\quad \to 2*x+1$

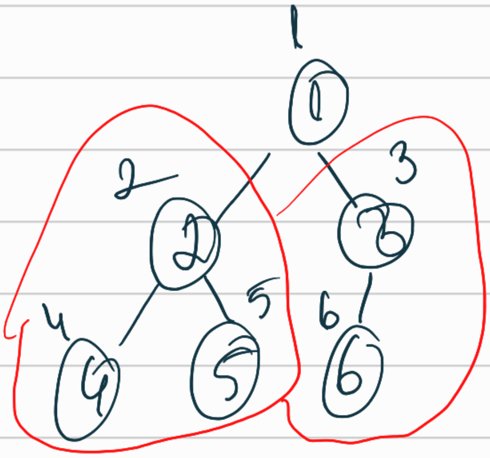
$x \to \left[\frac{x}{2}\right]$

1	2	3	4	5	6	7	8	9
3	5	7	6	7	8	12	8	9

$H[x] \geq H\left[\frac{x}{2}\right]$ $\boxed{\text{Condiție de Heap}}$

Ex: Permutările cu 6 elemente.

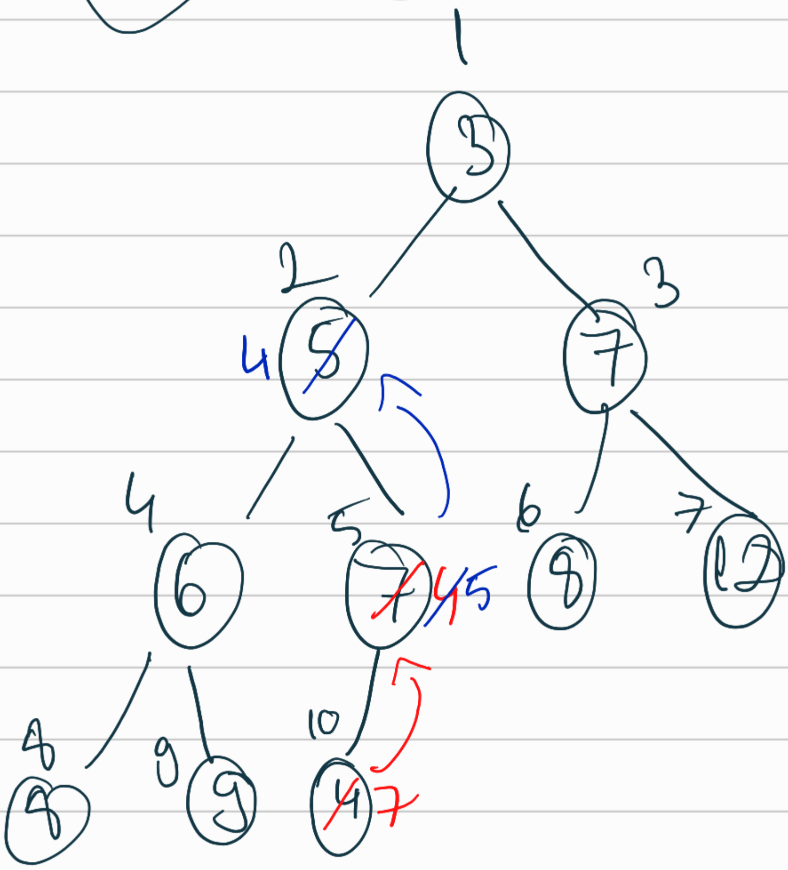
Câte permutări au aspect de Heap



$N = \boxed{2} \rightarrow 1 \text{ Heap}$

$V = 3 \rightarrow 2 \text{ Heap}$

$C^{3}_{5} \cdot 2 \cdot 1 = 20$



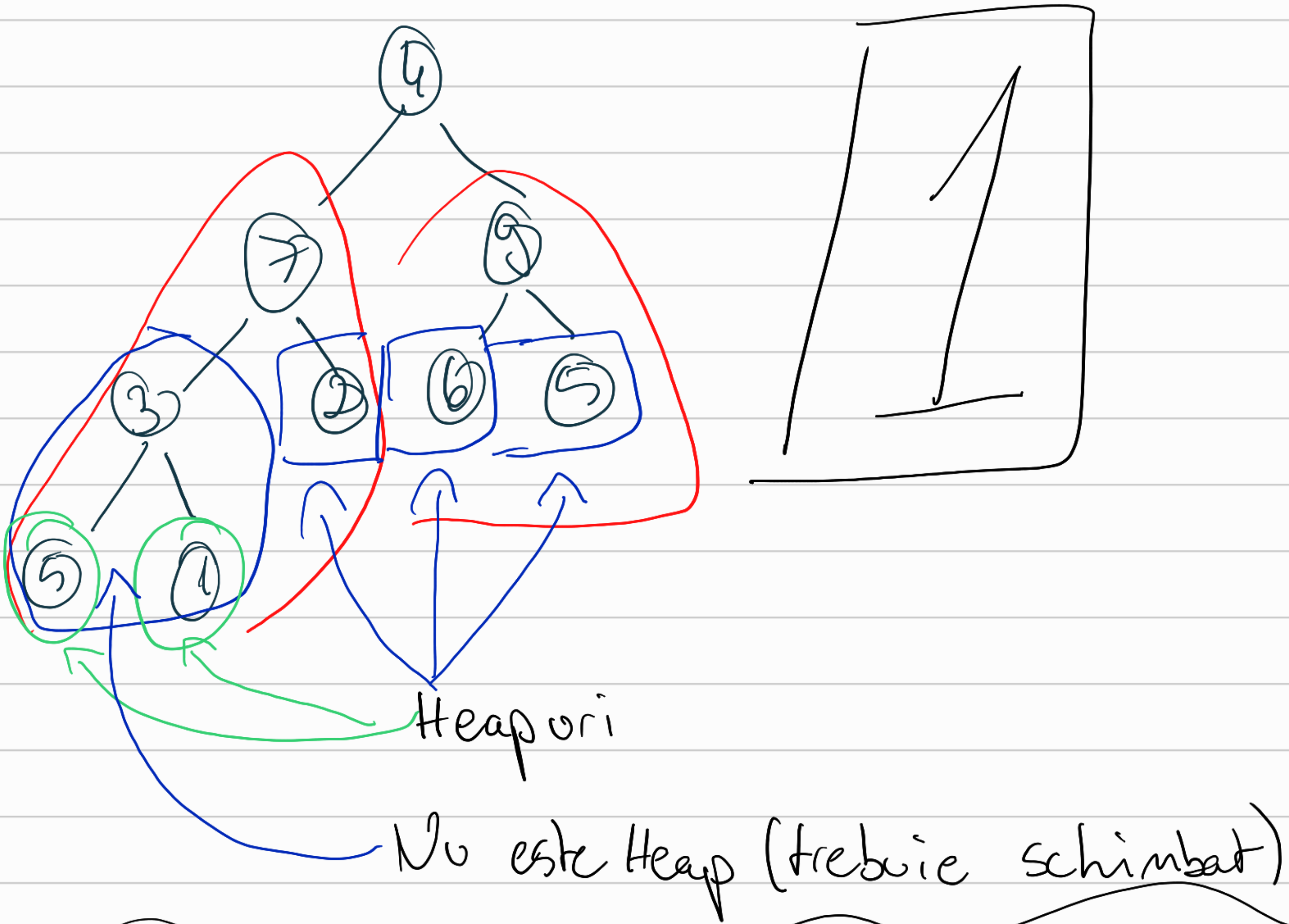
ADD 4

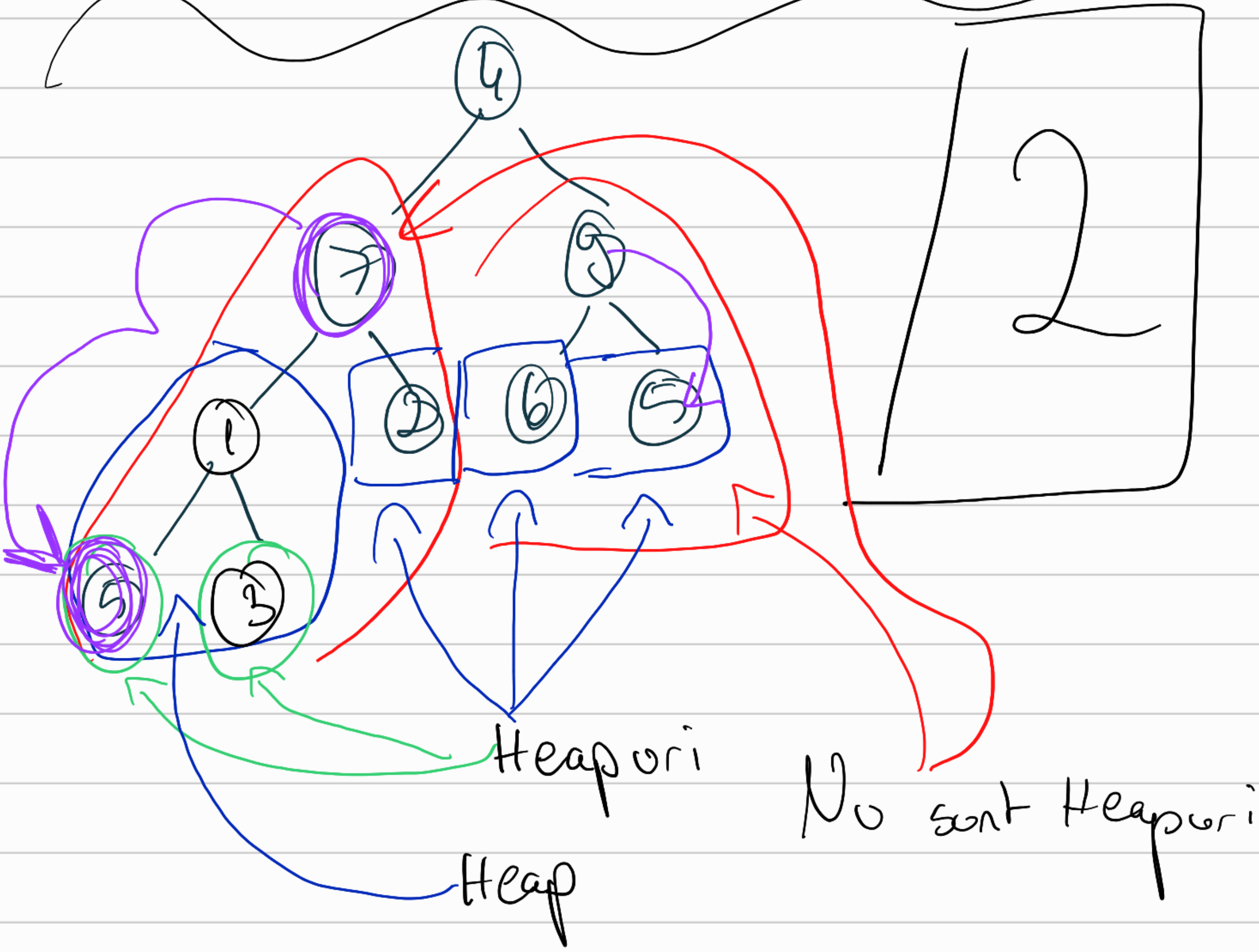
$\rightarrow O(\log(|S|))$

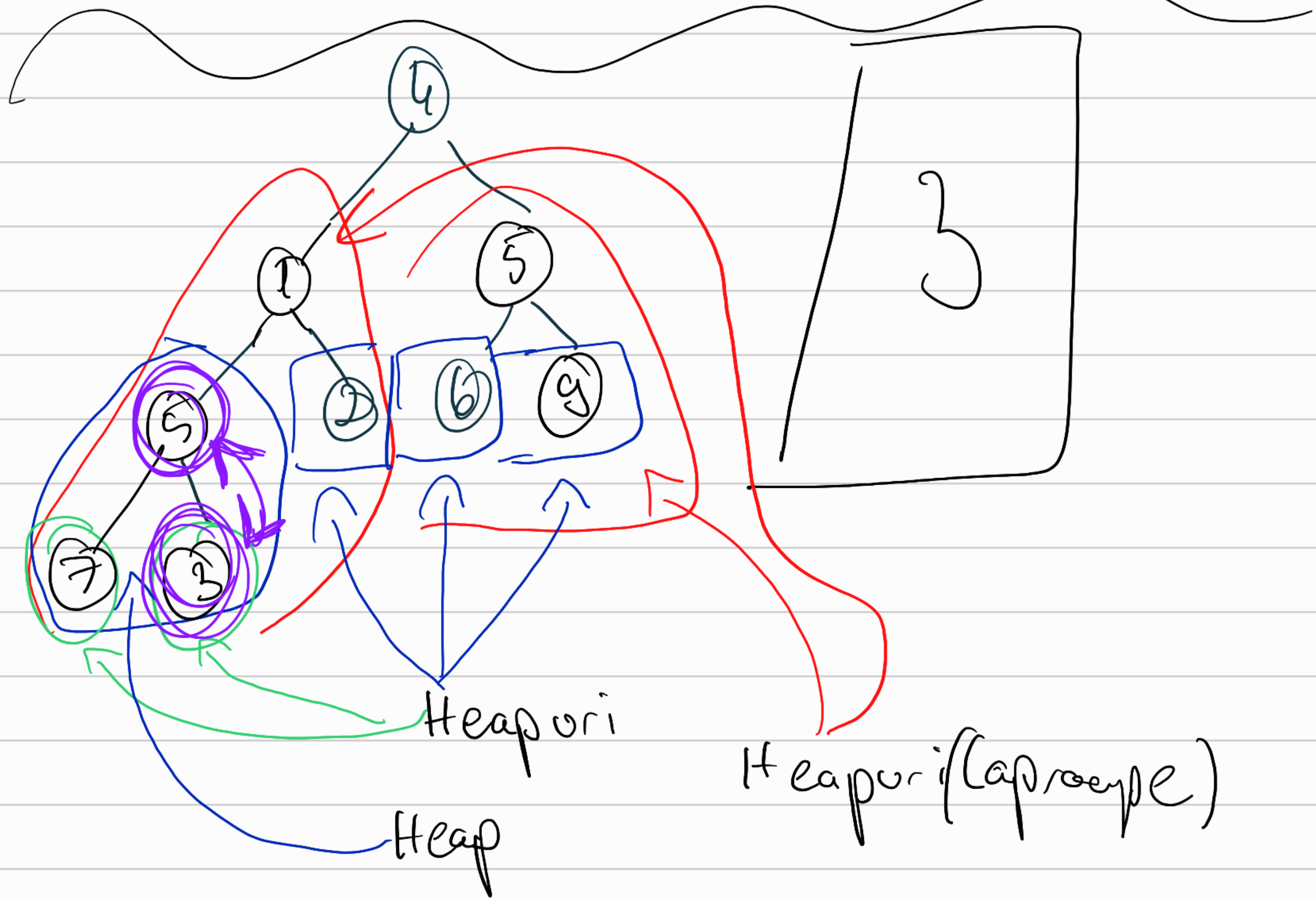
1	2	3	4	5	6	7	8	9	10
3	~~5~~ 4	7	6	~~7~~ ~~4~~ 5	8	12	8	9	~~4~~ 7

$[5/2]$

$[10/2]$


1
Heapori
Nu este Heap (trebuie schimbat)


2
Heapori
Nu sunt Heapuri
Heap



Şi aşa mai departe!!!

Heapsort

Se dă un vector V

MAX_HEAP din V $\Theta(n)$

POP_MAX de N ori → V devine sortat

$O(n \log n)$

GATA HEAPSORT!!!

Problemă: S - mulțime

$S = \emptyset$

Q - operații $\le 10^6$ $\quad x \le 10^6$

ADD x → $S = S \cup \{x\}$

GET_MIN

DEL x → $S = S \setminus \{x\}$

pos[x] → poz. lui x în Heap

Problemă:

$Q, K \le 10^5$

$$Q = \begin{cases} \text{ADD } x, & V \mathrel{+}= [x] \\ & \text{SORT}(V) \\ & \text{Afișăm } V[K] \end{cases}$$

Un vector V[K]

$O(Q \cdot (Q+K) \cdot \log(Q+K))$

Facem Max HEAP

Dacă $x < H[1]$ îl adăugăm, altfel, irelevant.

$$\Theta(K + Q \log K)$$

Problemă: Se dă un vector $v[N]$

Se dă un K

Care este subsecvența de sumă maximă

$lg \leq K$

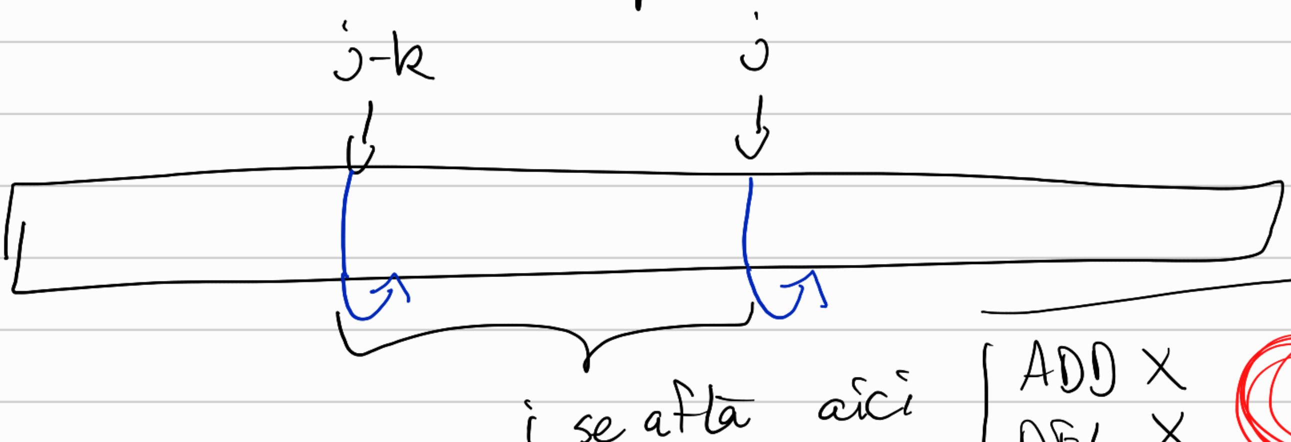
$$\Theta(n \log K)$$

$$S[i] = v[1] + v[2] + \dots + v[i]$$

$i < j$

$$s[j] - s[i] = v[i] + v[i+1] + \dots + v[j-1] + v[j]$$

$1 \leq j - i \leq K$ max posibil



$s[j] - s[i]$ ← minim

ADD X
DEL X
GET_MIN