

Structuri de date

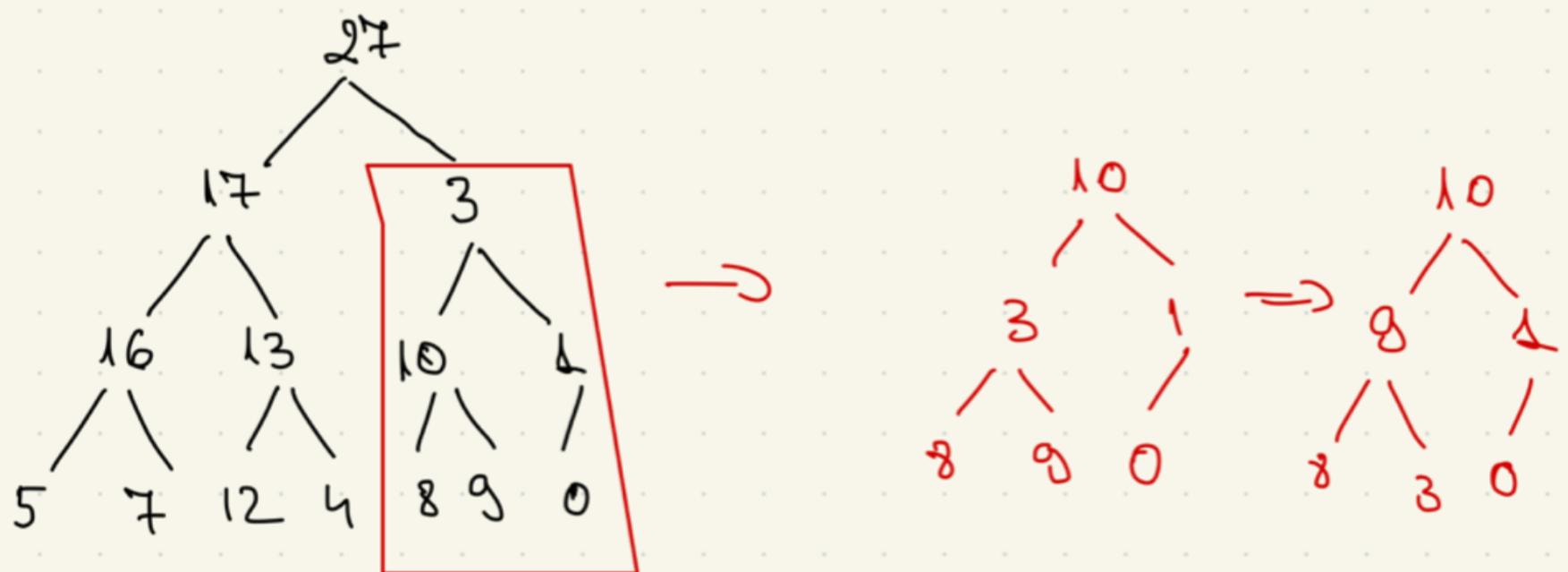
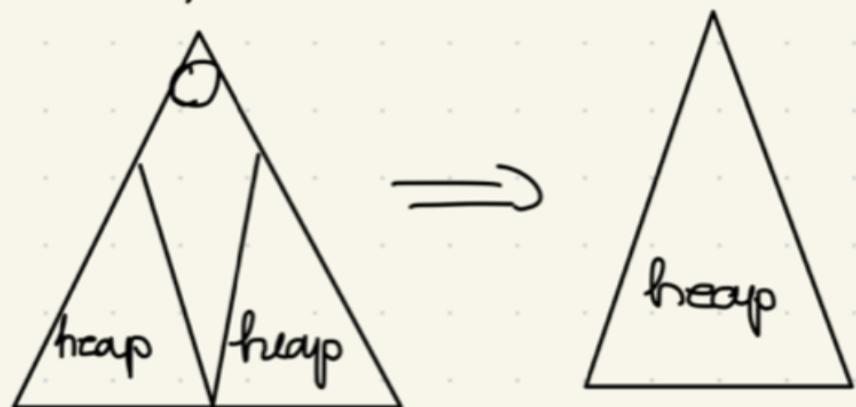


Struct de date - seminar 1 - 6.03.2024

6.2-1

(MAX)-HEAPIFY (A, i)^{poz}

$$A = [27, 17, 3, 16, 13, 10, 1, 5, 7, 12, 4, 8, 9, 0]$$

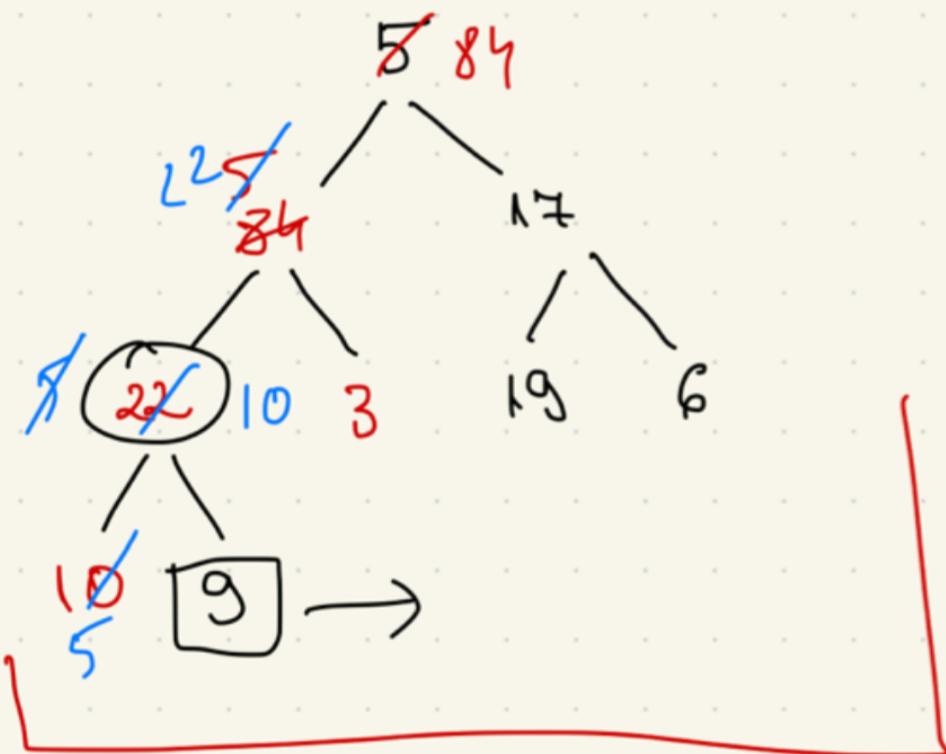
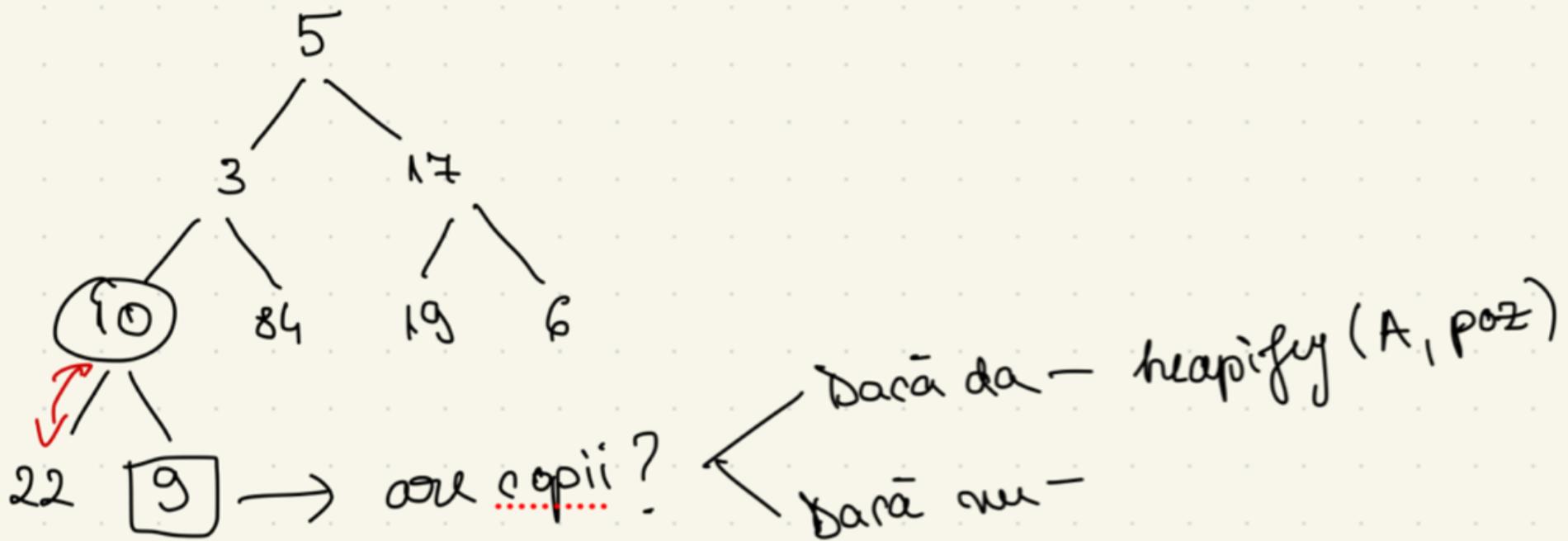


Vectorul după heapify: 27 17 10 16 13 9 1 5 7 12 4 8 30

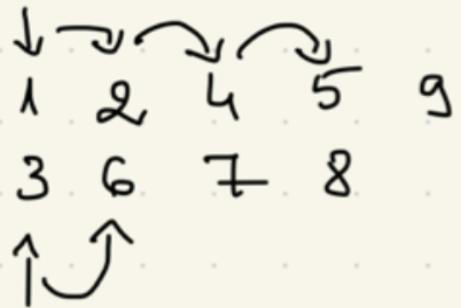
$$\begin{array}{c} i \\ \swarrow \quad \searrow \\ 2i \quad 2i+1 \end{array}$$

Build max

$$A = [5, 3, 17, 10, 84, 19, 6, 22, 9]$$



6.5-8 MERGE SORT $\Theta(n \log n)$

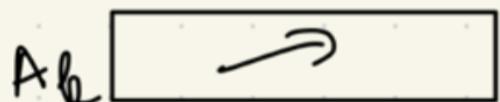


1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---

MERGE (A_1, A_2)

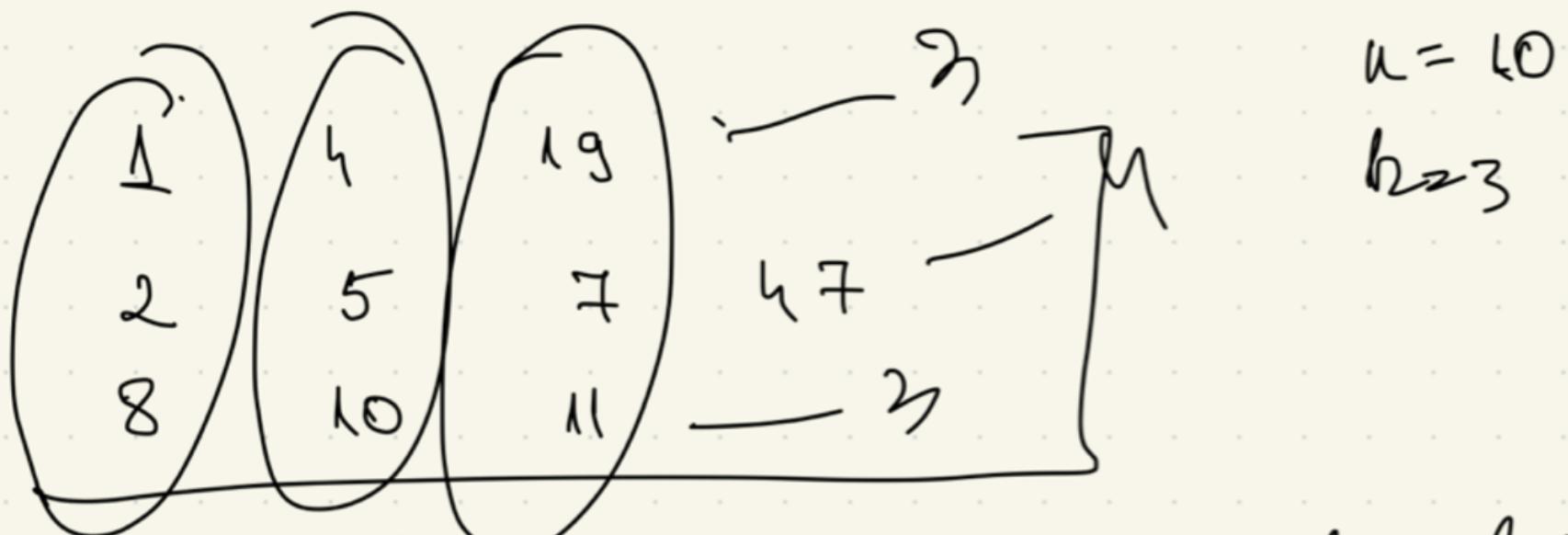


:
:
{'
|

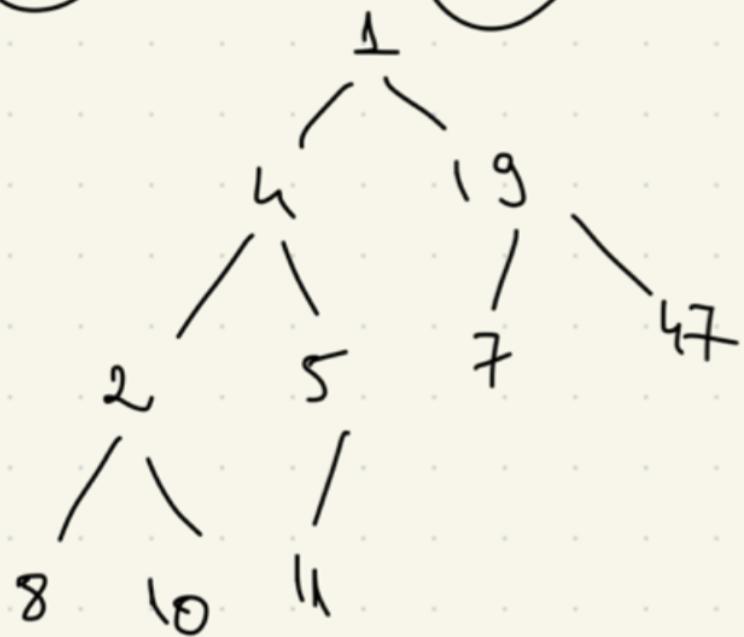


k vectori sortati

să facem un vector sortat astfel încât
să rezulte în $\Theta(n \log k)$



parcursul de la dreapta



$$n = h_1 + h_2 + \dots + h_k$$

SOLUȚIE

V_1 : 2, 3, 8

făcem minihape

V_2 : 4, 5, 9

încăpem cu primele elemente

V_3 : 1, 6, 7, 11



\Rightarrow punem în vect pe \perp și depă în heap adăugăm cum element din vectorul său corespunzător



PB 6-3

TABELA YOUNG

1	2	3	10
4	5	11	12
6	7	13	∞
8	9	∞	∞



Tab young (4, 4, 3, 2)

în căte moduri putem pune nr de la 1 la 13 într-o astfel de tabelă

13 |

$\prod_c f(c)$

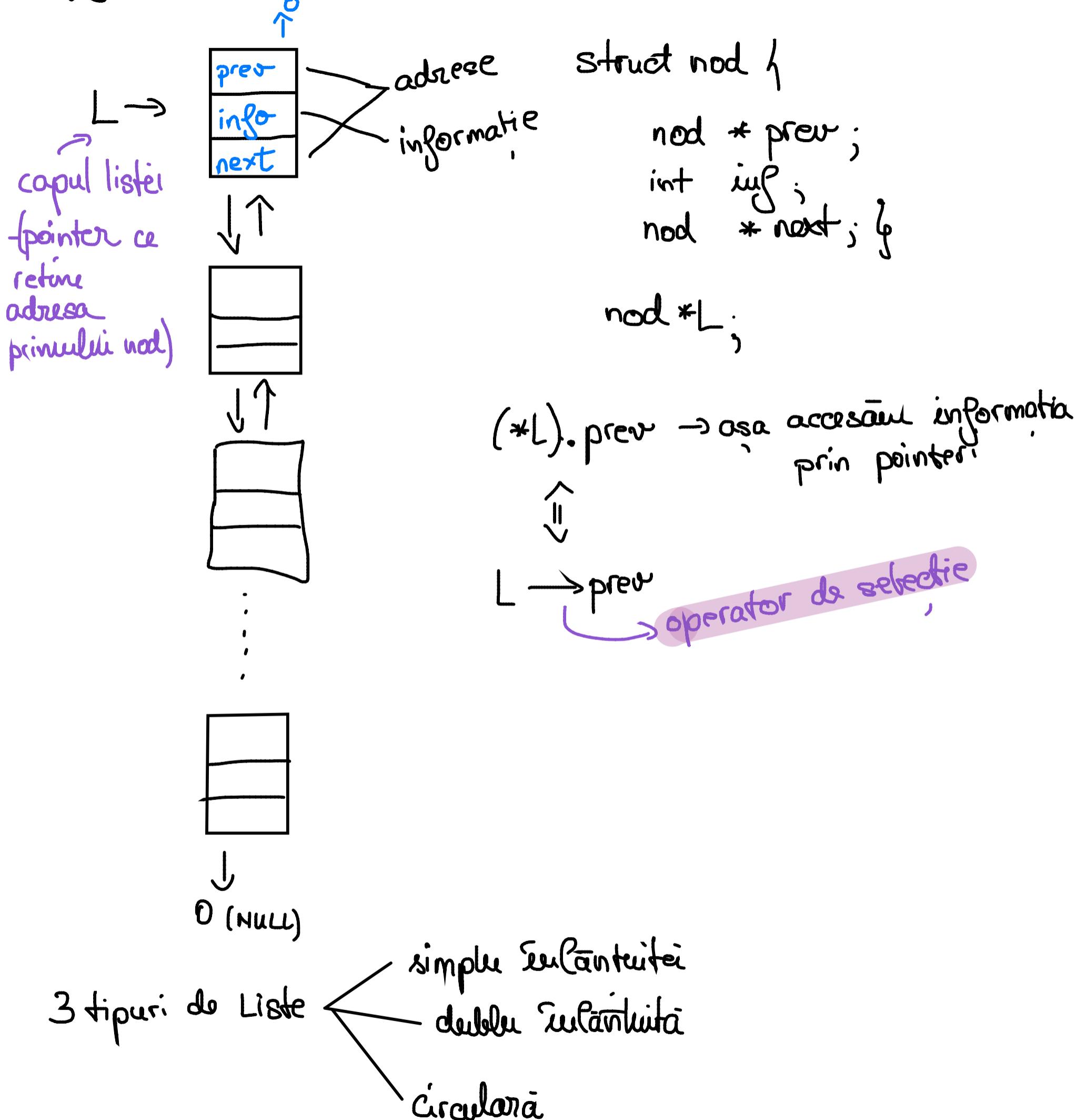
celulă $\nearrow \nwarrow$ funcție care depinde de c

- Desenati o tabelă Young cu nr 9, 16, 3, 2, 14, 8, 18, 14, 12

2	3	4	5
8	9	12	14
16	18	∞	∞
∞	∞	∞	∞

Extract min

Laborator - 15.03.2024



Operatii liste:

- 1) Crearea listei
- 2) Afisare
- 3) Inserare
- 4) Cautare
- 5) Stergere

pt lista

Seminarul - 20.03.2024
- pe cînd editia a 3-a -

STIVE SI COZI

STIVE

$\boxed{\text{push}(x)}$
 $\text{pop}(x)$

COZI

$\boxed{\text{ENQUEUE}(x)}$
 $\text{DEQUEUE}(x)$

10.1 - 6

O coadă cu 2 stive + complexitatea operațiilor

Readare

1 stivă pt a adăuga elemente în coadă
1 stivă pt a inversa elementele

Când adaugăm un elem în coadă

il adăugăm în stiva de adăugare

Când vrem să elrimuim din coadă e goală \Rightarrow mutăm verificăm dacă stiva 2 e goală elem din stiva $1 \rightarrow 2$

un egodă \rightarrow eliminarea direct a cel element adăugat

Enqueue $\rightarrow \Theta(1)$

Dequeue $\rightarrow \Theta(n)$

Pe ideea astfel

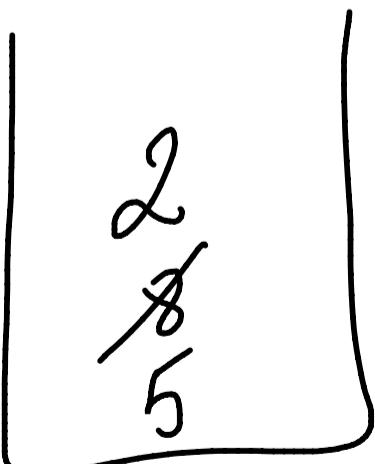
```

class queue()
{
public:
    void enqueue();
    int dequeue();
private:
    stack<int> s1;
    stack<int> s2;
}

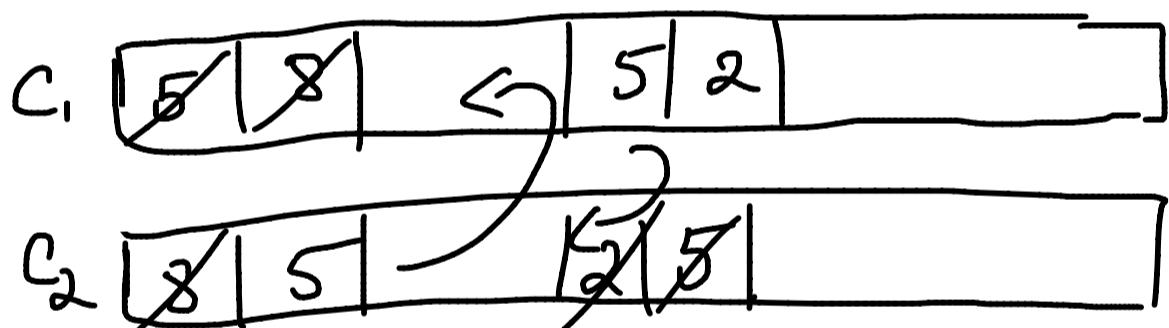
```

Pb 10.1 - 7 : Să implementăm o stivă cu 2 cozi
+ Complexitate

push(5)
push(8)
pop() → 8
push(2)
pop() →
pop() →



Coda last-in - first out



push(2) - enqueue(2)
după pop()

- dequeue din prima
- enqueue în a două
- pop dequeue în a două

și după dequeue și enqueue
ca să le menținem
în prima
coadă

PROBLEMA CELEBRITĂȚII

$\Theta(n)$

→ găsește celebritate
→ sau spune micăluță

$\text{KNOWS}(x, y)$ → DA
 → NU

x celebritate dacă
 $\forall y \neq x, \text{knows}(y, x) = \text{DA}$
 $\forall y \neq x, \text{KNOWS}(x, y) = \text{NU}$

HINT 2 persoane

$\text{KNOWS}(x, y)$

- $K(x, y) = K(y, x) = \text{DA}$
- $K(x, y) = \text{DA}, K(y, x) = \text{NU}$
- $K(x, y) = \text{NU} \rightarrow K(y, x) = \text{DA}$
- $K(x, y) = K(y, x) = \text{NU}$

$\} \Rightarrow$ oici e celebritate

coada

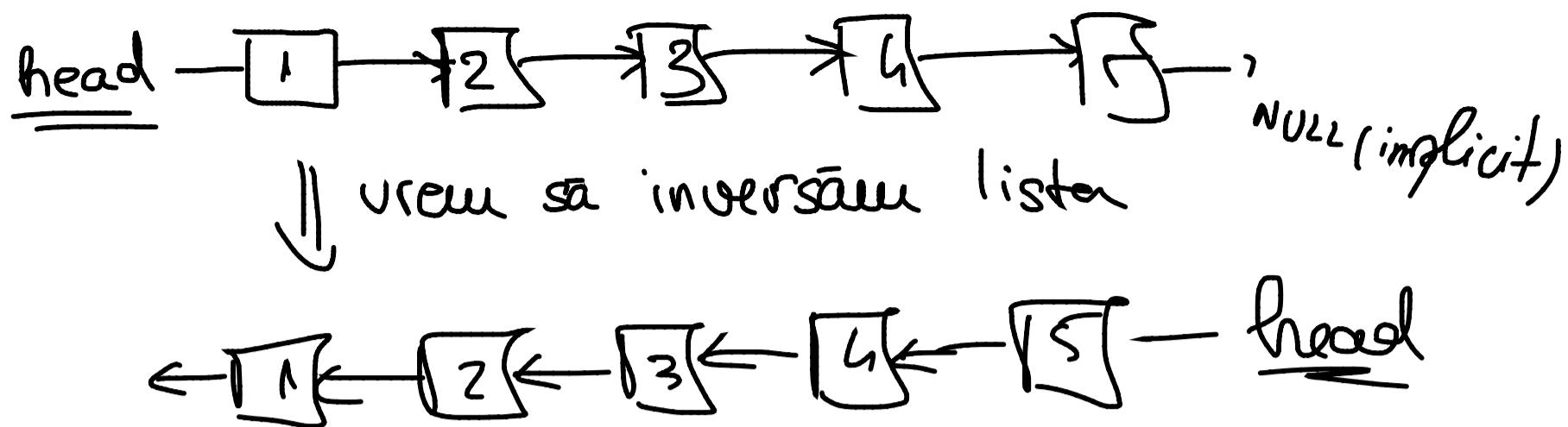
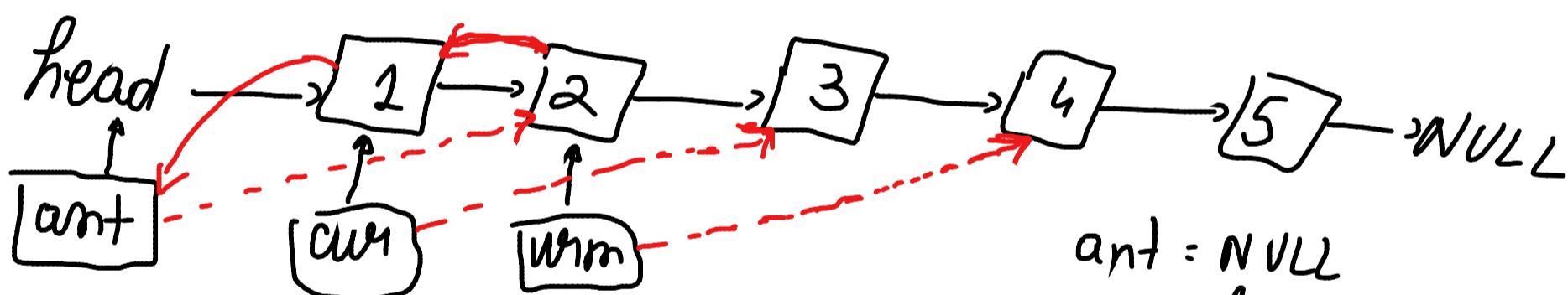


LISP

```
(defun fibo(n)
  (cond
    ((=n 0) 1)
    ((=n 1) 1)
    (T
      (+ (fibo (-n 1))
          (fibo (-n 2))))
```

Aveam un fisier cu prog în LISP

Vrem să facem un editor de texte care caute greseli în program

ne recursiv $O(1)$ - memorie suplimentarăprev
next

while (current != NULL)

urm = cur → next

cur → next = ant

ant = cur

cur = urm

urm = urm → next

head = ant

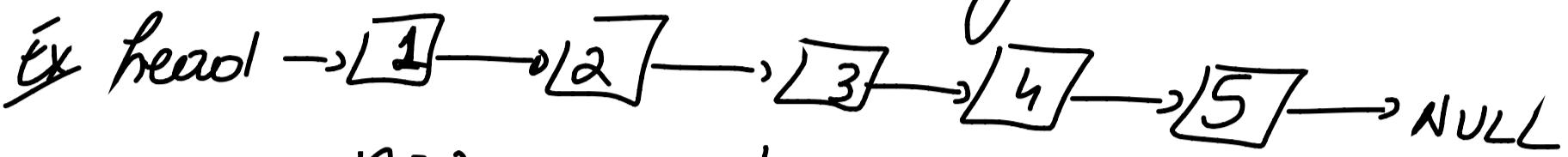
* la final ant = 5, iar
cur = NULL→ 3 pointeri *ant, *cur, *urm să fie elem
succ al unei liste

Problema 2

- Lista simplu înțăntuită

- $n \geq n$

VREAU să printez valoarea de la n -lea nod (de la cap la coadă). recursivă. NV parcurg lista de la n -lea nod (de la cap la coadă)

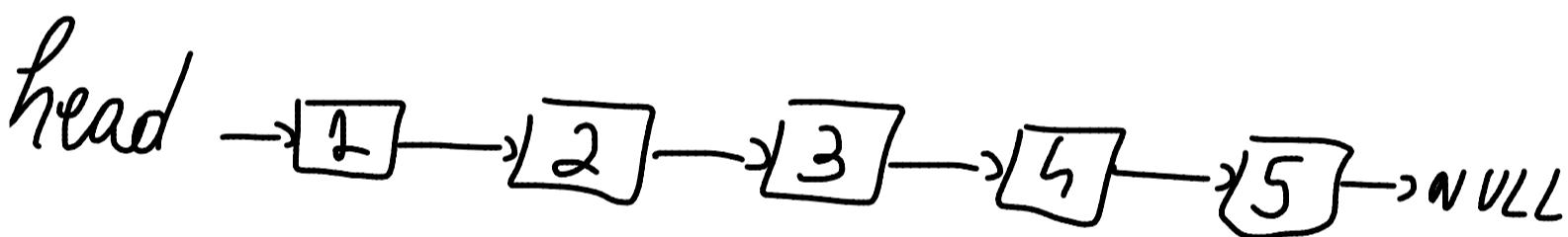


$n=2 \rightarrow$ printăm 4

$n=7 \rightarrow NV$ EXISTĂ

$n=3 \rightarrow$ printăm 3

* noi stim doar HEAD la inceput



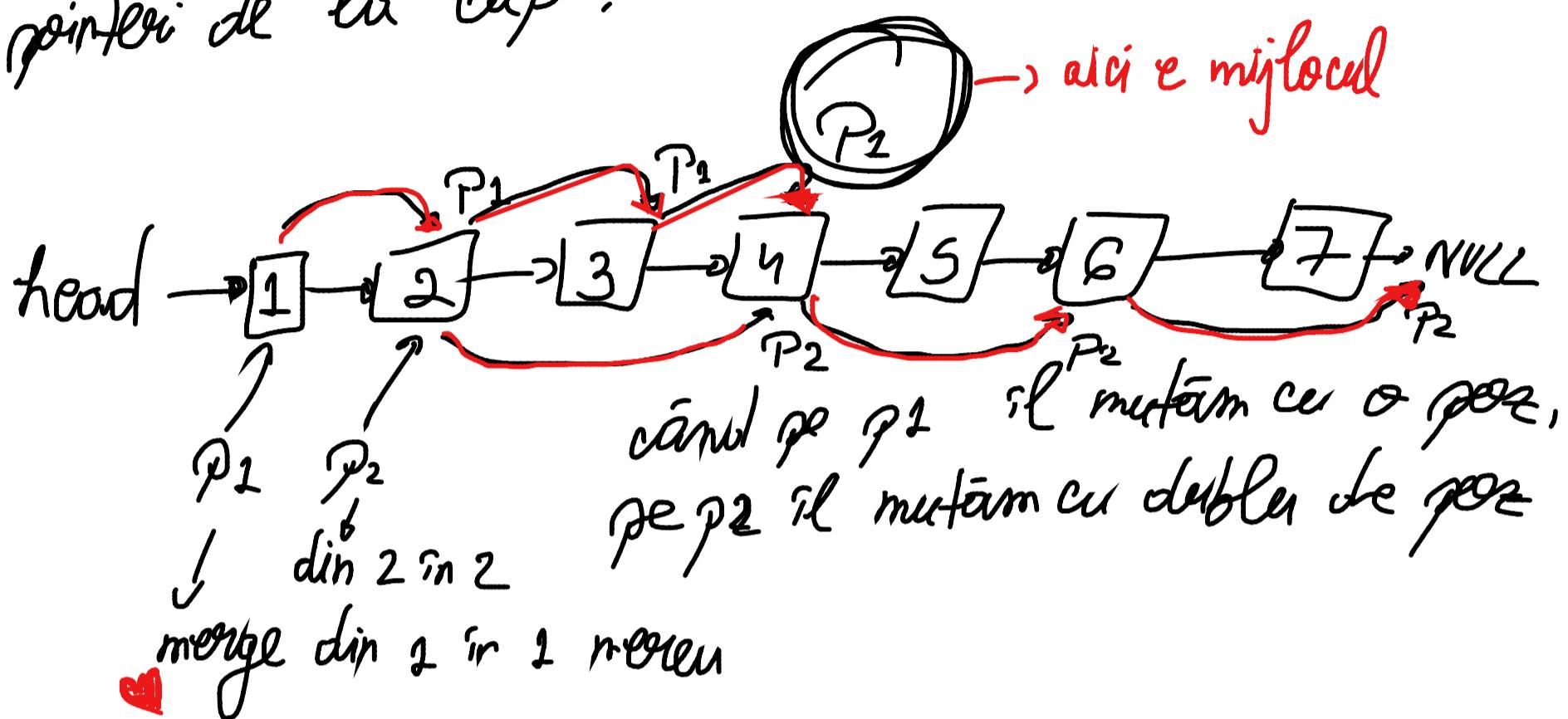
- avem 2 pointere care vor să se întâlnească în ac.
- timp pe listă,

când al doilea pointer ajunge la NULL, primul pointer ajunge la al n -lea de la COADA la CAP ! , deci

dist dintre 2 pointeri este de n
cei

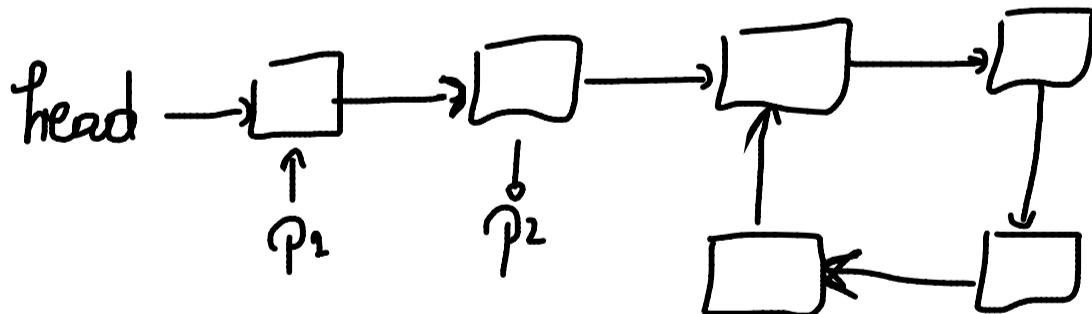
Problema 3

- pornind de la head tb să afleam mijlocul liste;
 nu putem parcurge de 2 ori, nici să plecăm cu 2
 pointeri de la cap și de la coda



Problema 4

- verificati dacă există ciclu între-o LSG i

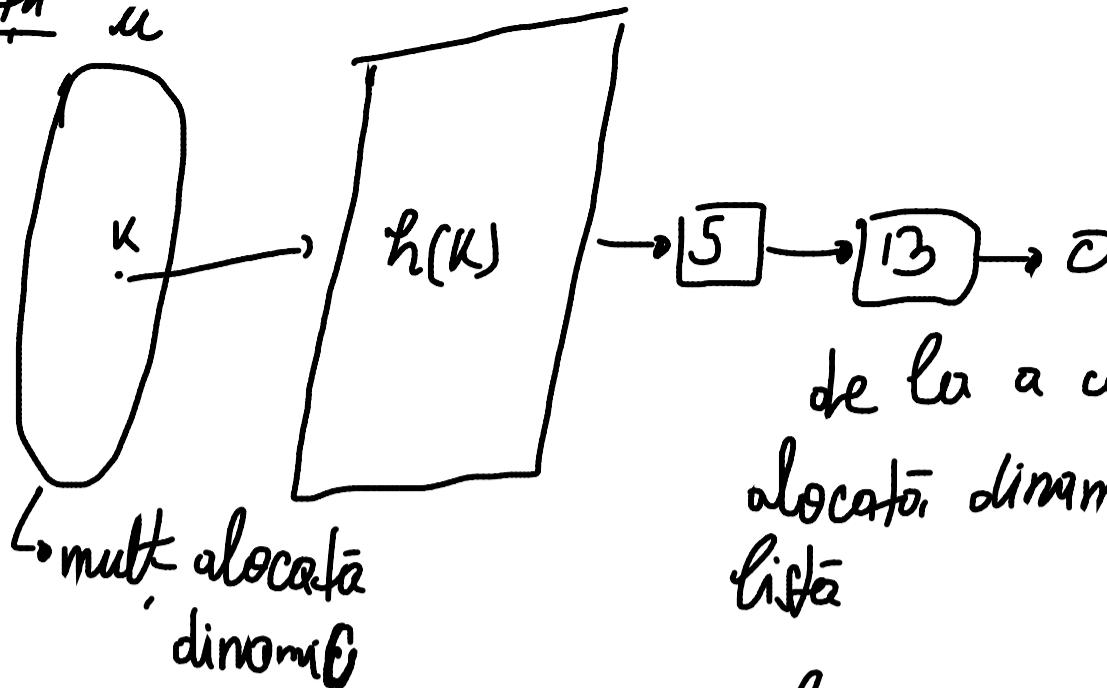


↓
NU SE DĂ ASTA
LA EXAMEN

Caz I: un pointer aj la NULL → nu avem ciclu

Caz II: dacă aj în ac poz → lista are ciclu

Exerciții



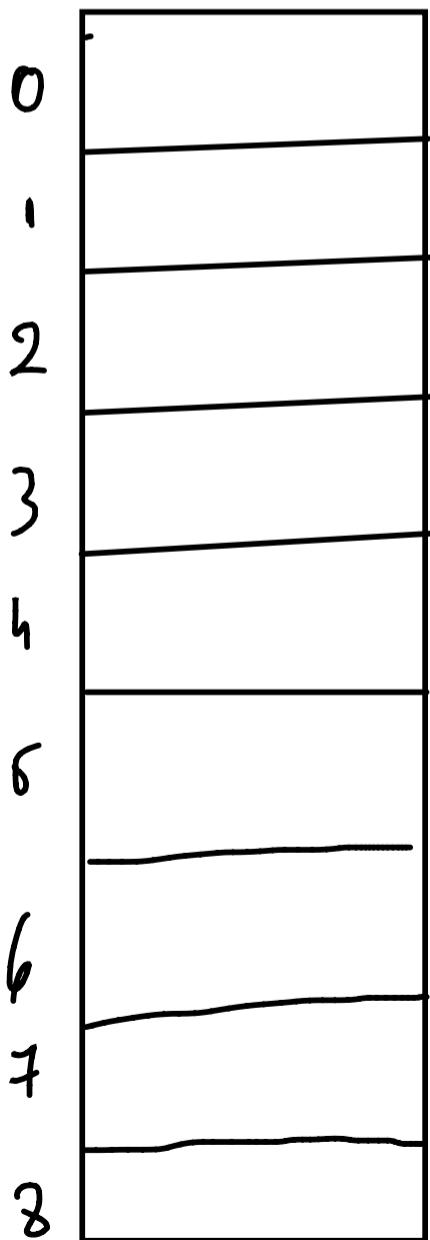
de la a căuta într-o mt
alocată dinamic căutăm într-o
listă

11.2 - 2 - dem să inserăm urm elem 5, 28, 19, 15, 20, 33, 12, 17, 10

într-o tabelă de dispersie cu 9 sloturi
(adică tabela are 9 pozitii)

$$h(K) = K \% 9$$

$\rightarrow 28 \rightarrow 19 \rightarrow 10$



$\rightarrow 15 \rightarrow 33$

$\rightarrow 17$

$\rightarrow 28 \rightarrow 19 \rightarrow 10 \rightarrow 20 \rightarrow 12 \rightarrow 5 \rightarrow 15 \rightarrow 33 \rightarrow 17$

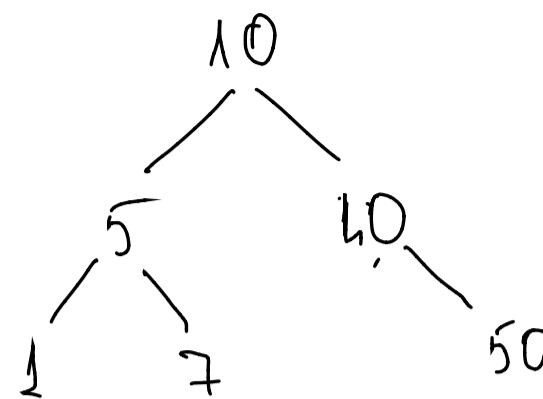
Seminar 5 - 22 mai -

1) lista de întregi $\{10, 5, 1, 7, 40, 50\}$

preord $\text{BST} \quad T$

$T = ?$

preordine VSD
inordine SVD
post ordine SDV



2) 12.2.-1 Cormen

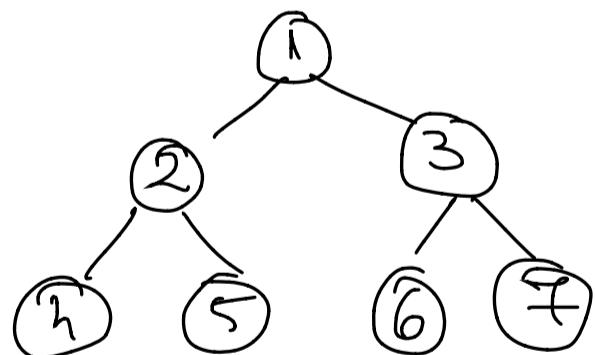
or între 1 și 1000 într-un BST

SEARCH (363)

care poate fi lista

a) 2, 252, 401, 398, 330, 344, 397, 363

3) Se dă un arbore binar



Vreau să printezi
 $\begin{bmatrix} 1 \\ 2 \ 3 \\ 4 \ \leftarrow 6 \ 7 \end{bmatrix}$

Vreau să printezi
 $\begin{pmatrix} 1 \\ 2 \ 3 \\ 4 \ \leftarrow 6 \ 7 \end{pmatrix}$

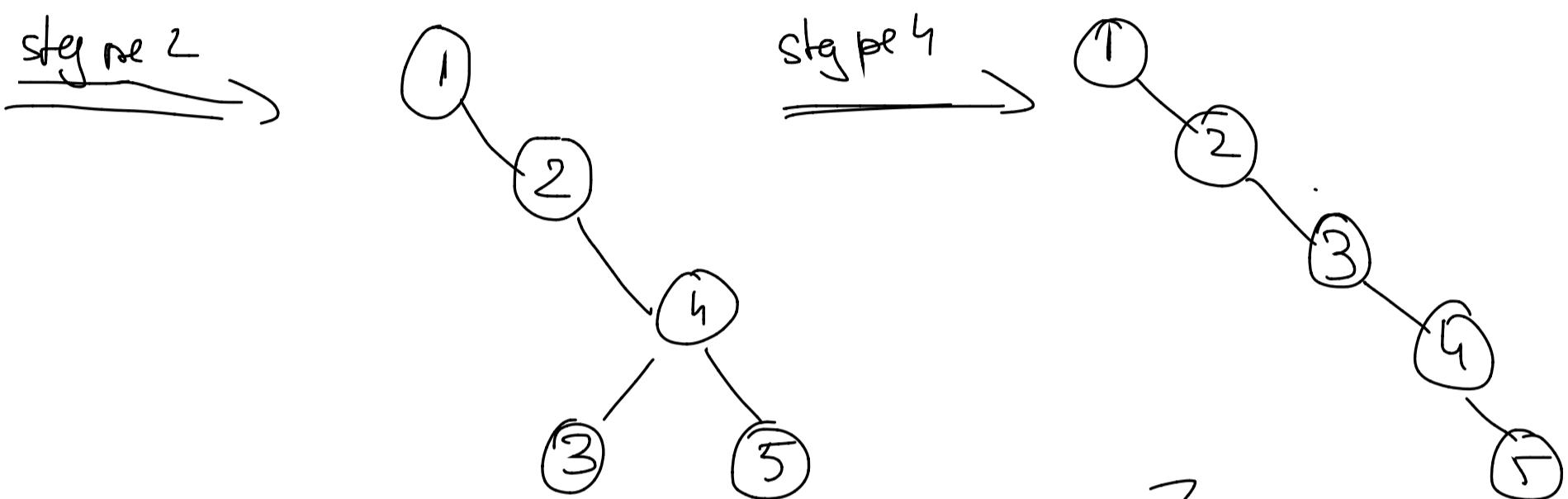
pt prima: [radacina]

[fiu 1] [LR RR]

[LLR RLR LRL RRR]

oici e posibile in
inordine și
actualizăm pe
pt coda

13.2-4



\Rightarrow iar pt cel de dreapta, il transforme an si depeste aplicarea rotatii inverse

AVL

Se gasit formula pt $T(h) = \# \text{minim de noduri al lui AVL}$

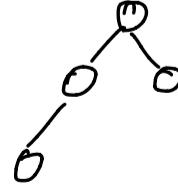
$$T(0) = 1$$



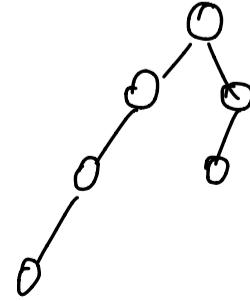
$$T(1) = 2$$



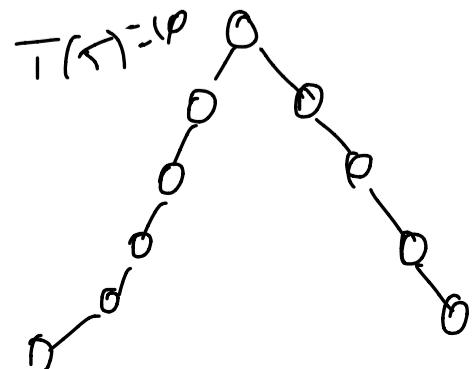
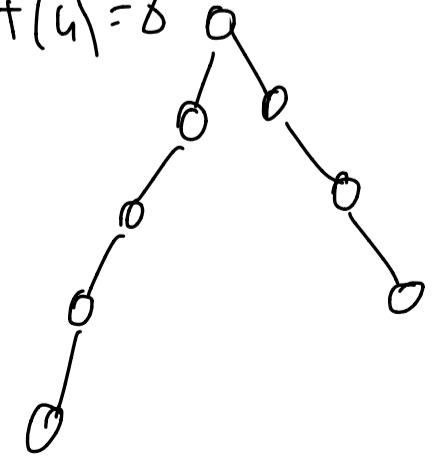
$$T(2) = 4$$



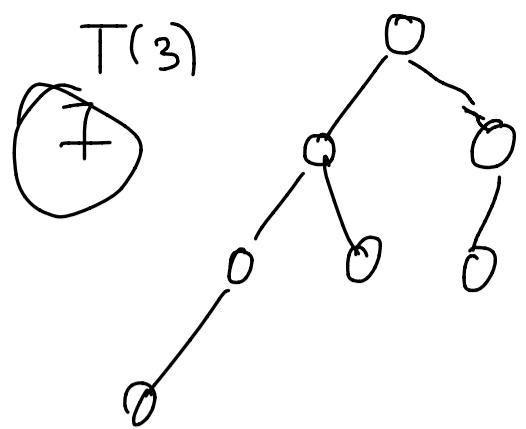
$$T(3) = 6$$



$$T(4) = 8$$

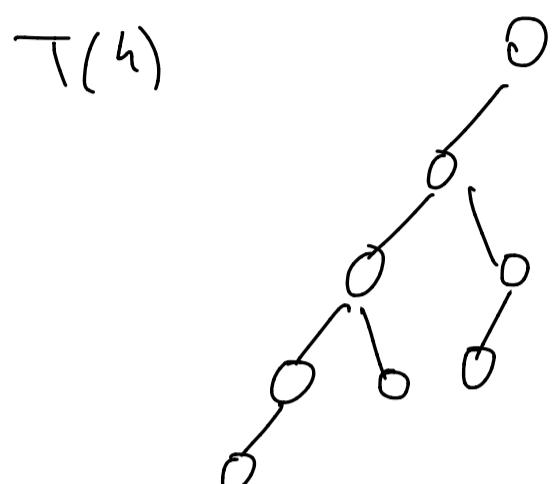


$$T(h) = \# \text{daca } h =$$



$$T(3) = 2^3 - 1 = 7$$

$$T(h) = T(h-1) + T(h-2) + 1$$



$$T(4) = 2^4 - 1 = 15$$

TREEC

$$T(h) = S(h) - a$$

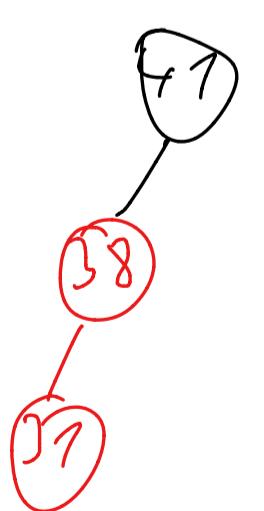
$$S(h) = a = S(h-1) - a + S(h-2) - a + 1$$

$$\xrightarrow{a} S(h) = S(h-1) + S(h-2)$$

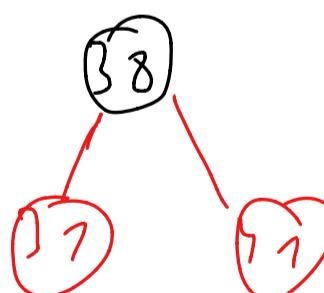
$$L = [47, 38, 37, 72, 79, 8] .$$

De introduc într-un RBT initial gol.

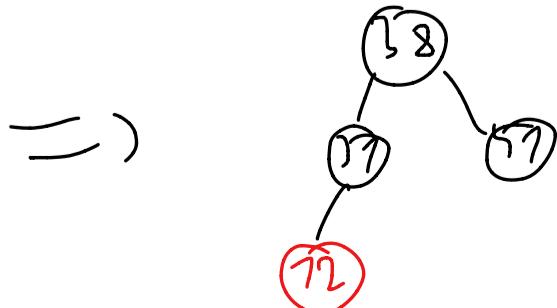
Încercare 47:



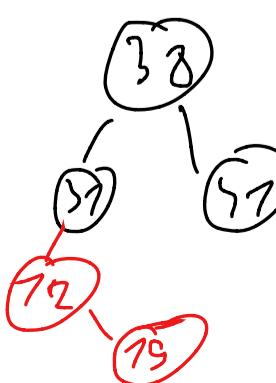
\Rightarrow



$+ 12 \Rightarrow$



$+ 19 \Rightarrow$



$(=)$

