Sortări Simple

- Inserție
- Selecție
- Interschimbare

- operații → comparare
  → interschimbare

```
void swap( int *a, int *b)
{   int tmp = *a;
    *a = *b;
    *b = tmp;
}

void cool_swap (...)
{ (*a) = (*a) ^ (*b);
  (*b) = ....
  (*a) = ....
}
```

$a \wedge a = 0$

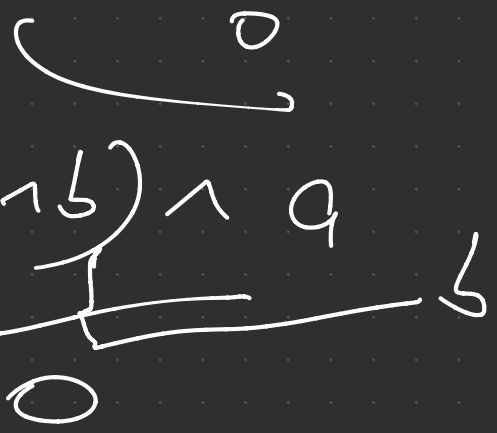
$a \wedge 0 = a$

$a \wedge b = a \oplus b =$

$a = \quad a \wedge b$

$b = \quad (a \wedge b) \to a \wedge b \wedge b = a$

$a = (a \wedge b) = (a \wedge b) \wedge a$



Sortare Stabilă →

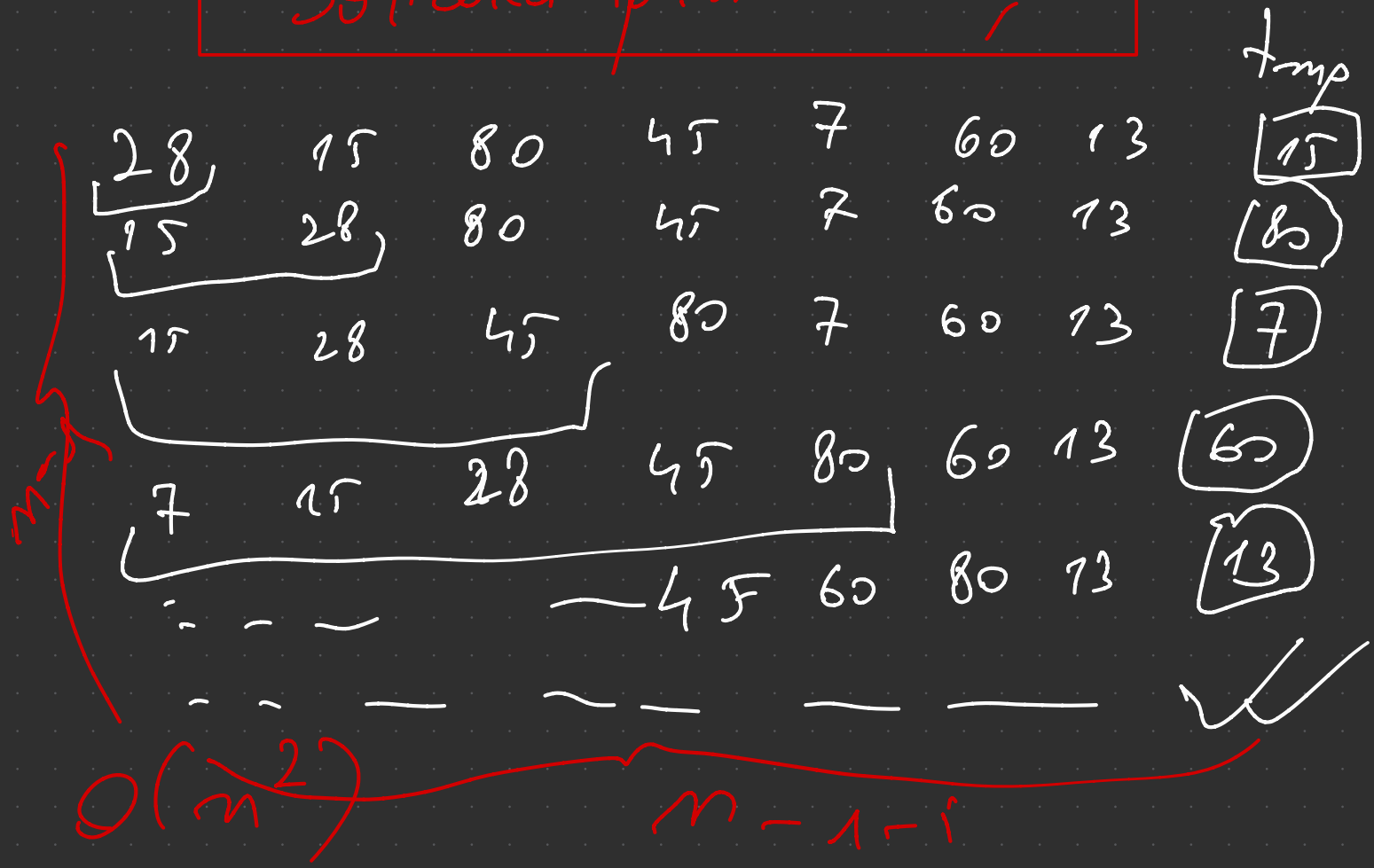
→ el. cu aceeași cheie își mențin poziția inițială

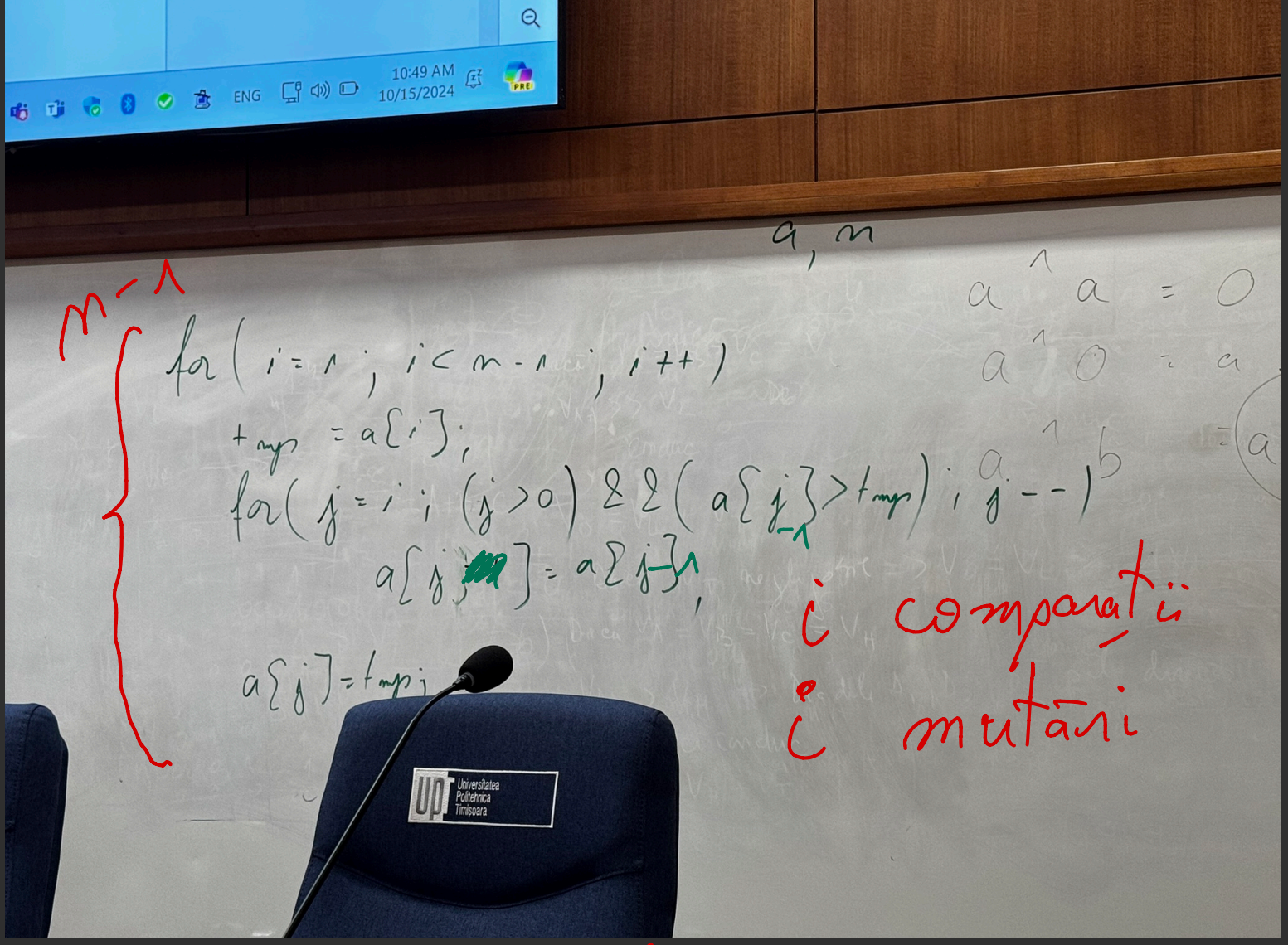
Tablouri -> mem. centrală
Sortările -> mem. RAM

Sortarea „în situ"
-> nu se folosesc tablouri auxiliare

Metode directe de sortare
-> $O(n^2)$

Sortarea prin inserție





$$C_{max} = \sum_{i=1}^{n-1} i = \frac{n(n-1)}{2} \Rightarrow O(n^2)$$

$$C_{min} = (n-1) \cdot 1 \Rightarrow O(n)$$

(deja sortat)

$$C_{mediu} = O(n^2)$$

Mutări:

$$M_{max} = \sum_{i=1}^{n-1} (i+2) \Rightarrow O(n^2)$$

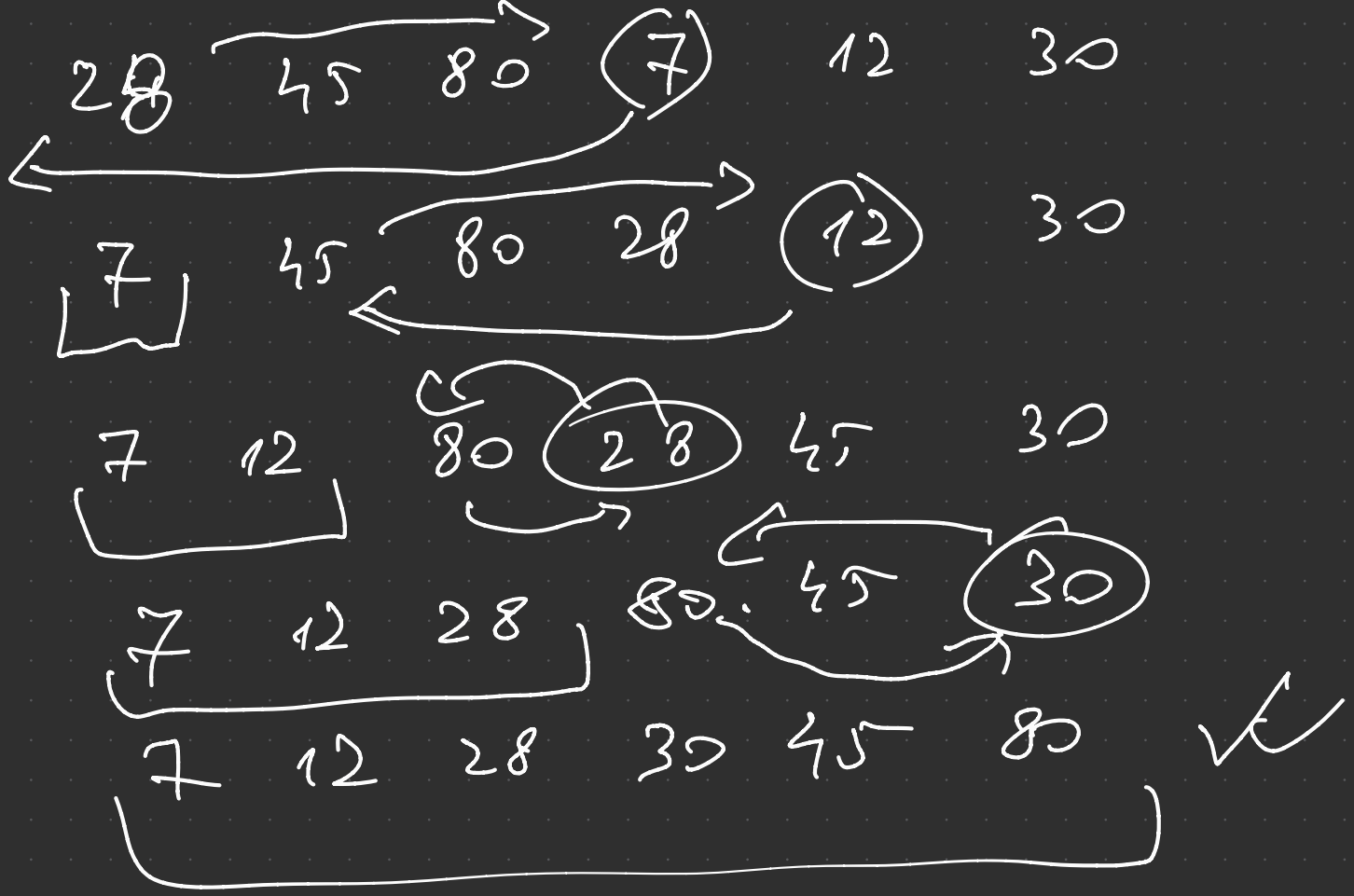
Avantaje

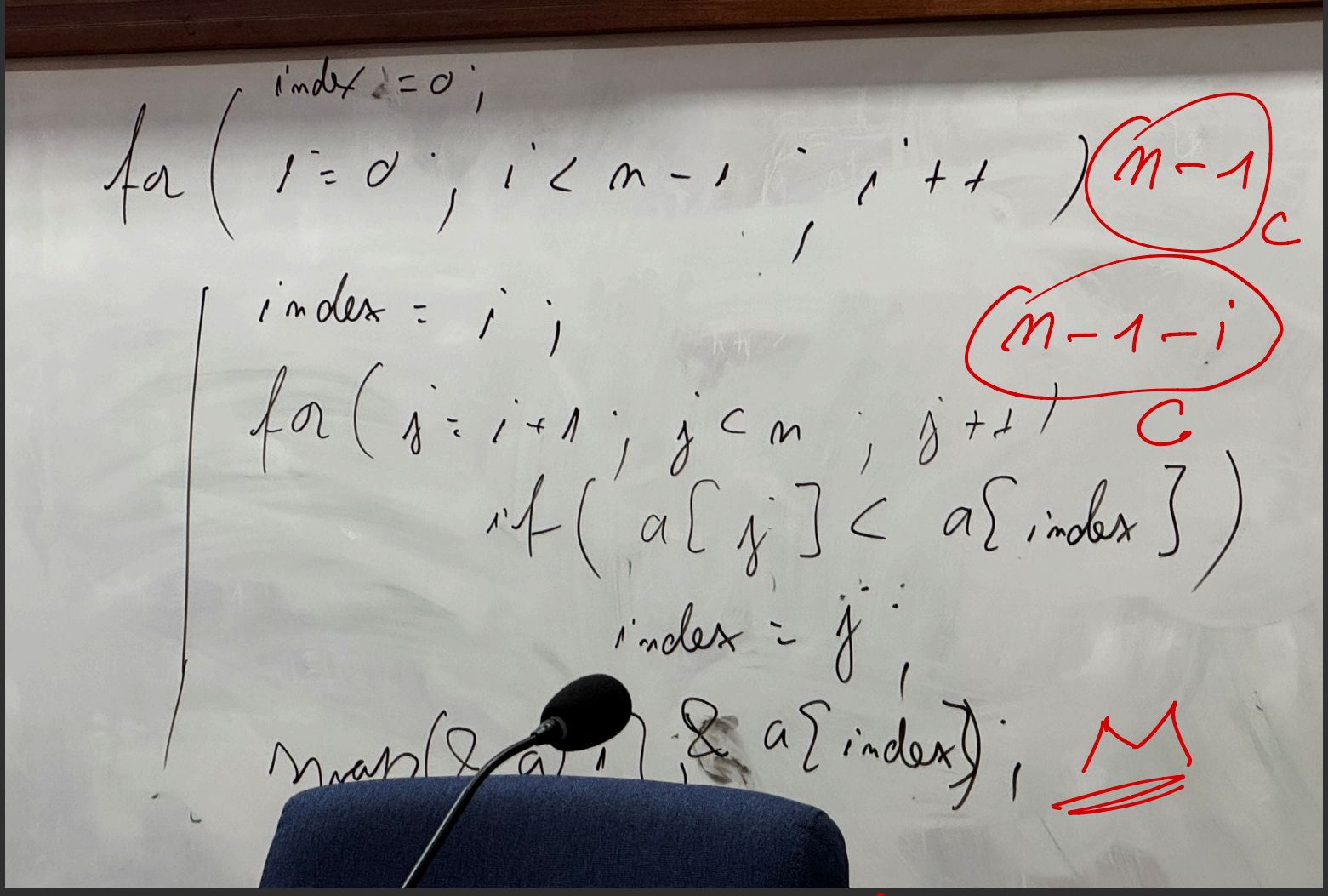
- simplă

Dezavantaje

ineficientă
(caz mediu == caz nefavorabil)

Sortare prin selecție





$$C_{max} = (n-1)(n-1-i) \Rightarrow O(n^2)$$

$$\sum_{i=0}^{n-1} (n-1-i) = (n-1)+(n-2)+\dots+2+1 = \frac{n(n-1)}{2}$$

$$M = 3(n-1) \quad O(n)$$

Sortarea prin interschimbare

28 45 80 7 12 30

swap

28 45 7 80 12 30

swap

28 45 7 12 80 30

swap

28 45 7 12 30 80

swap

28 7 45 12 30 80