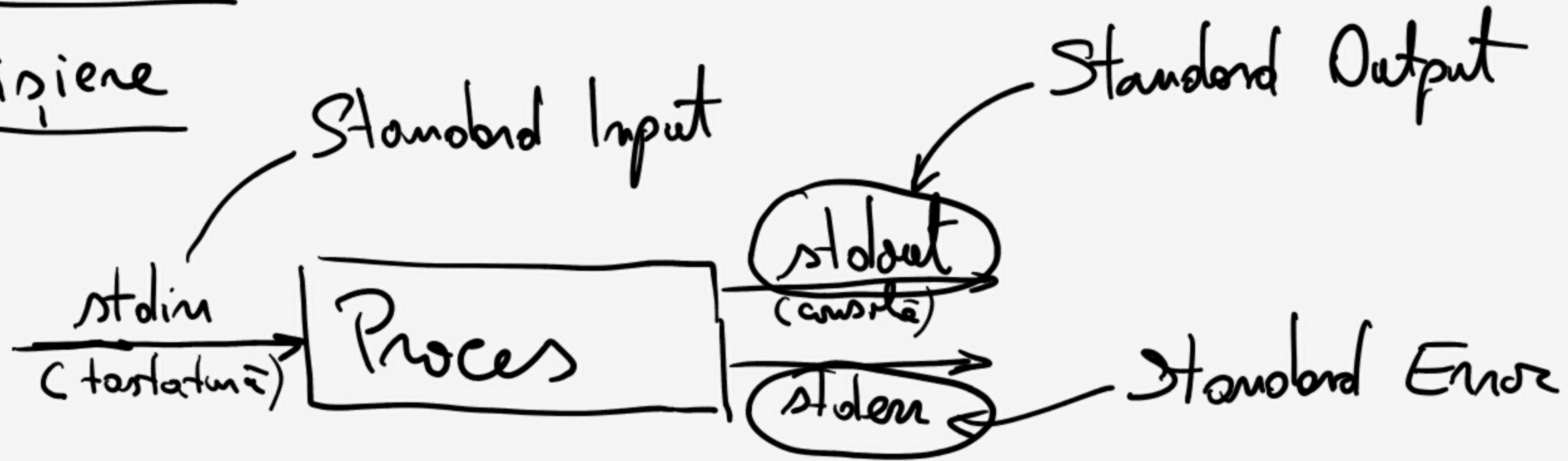
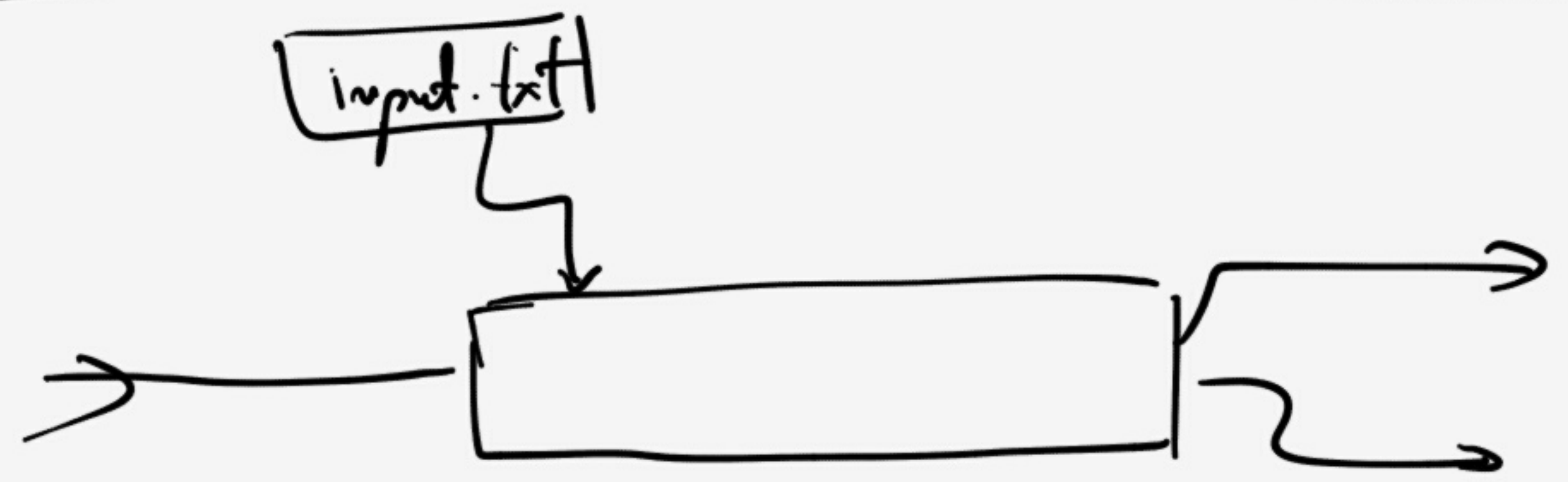
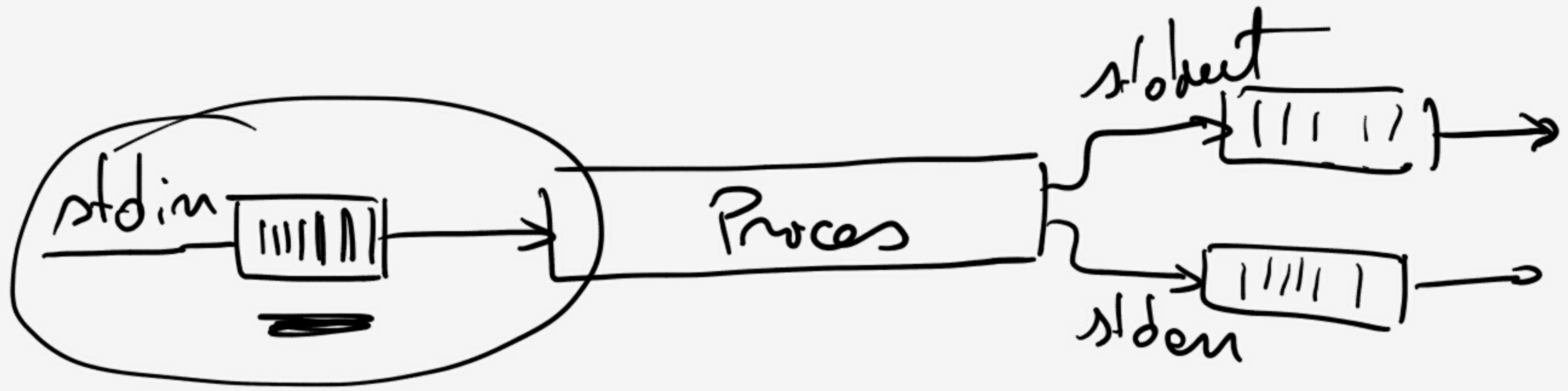


Curs 3

Fișiere



`printf("Hello!");` \equiv `fprintf(stdout, "Hello!");`
? \Rightarrow `fprintf(stderr, "Error!");`



```
#include <stdio.h>
```

```
int main() {
```

```
FILE *file_descriptor;
```

```
file_descriptor = fopen("input.txt", "w");
```

```
if (file_descriptor == NULL) {
```

```
printf(stderr, "N-am putat deschide input.txt");
```

```
return 0;
```

```
}
```

```
printf(file_descriptor, "Ana are %d mere", 10);
```

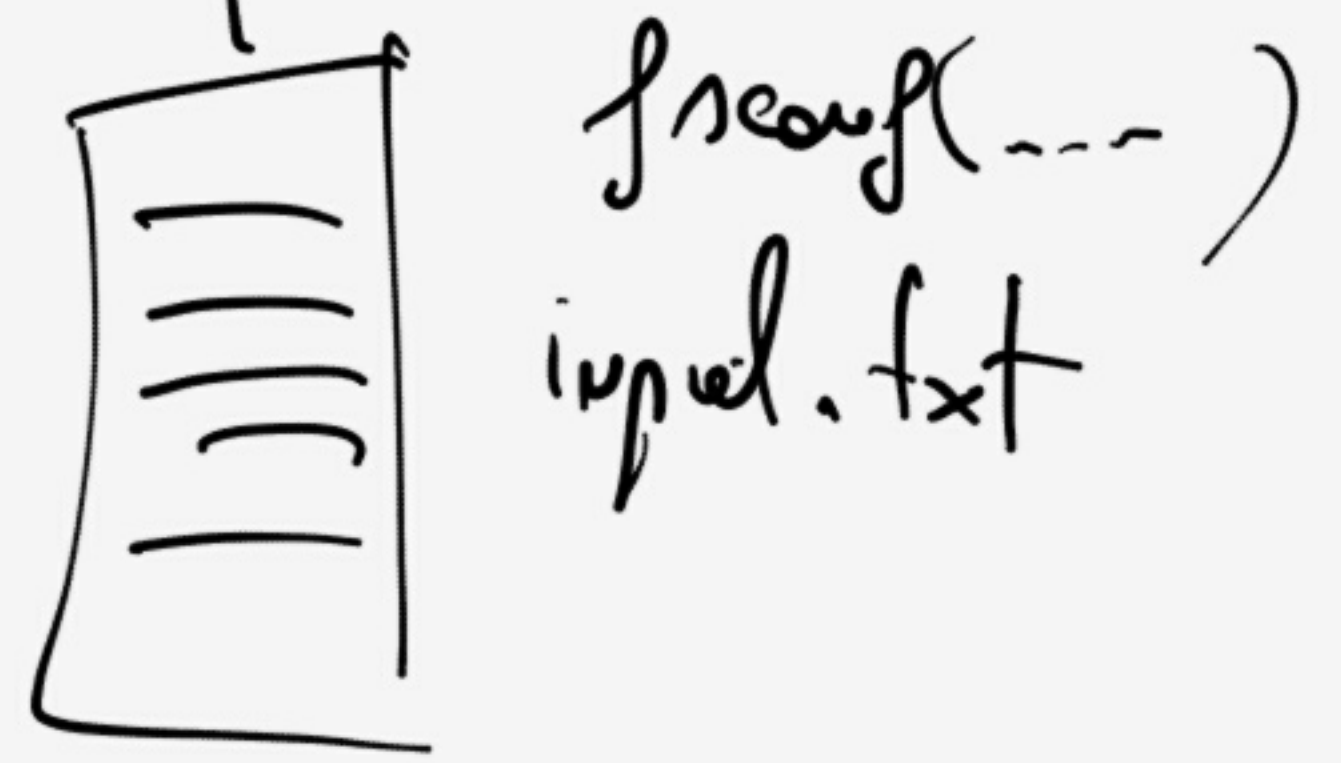
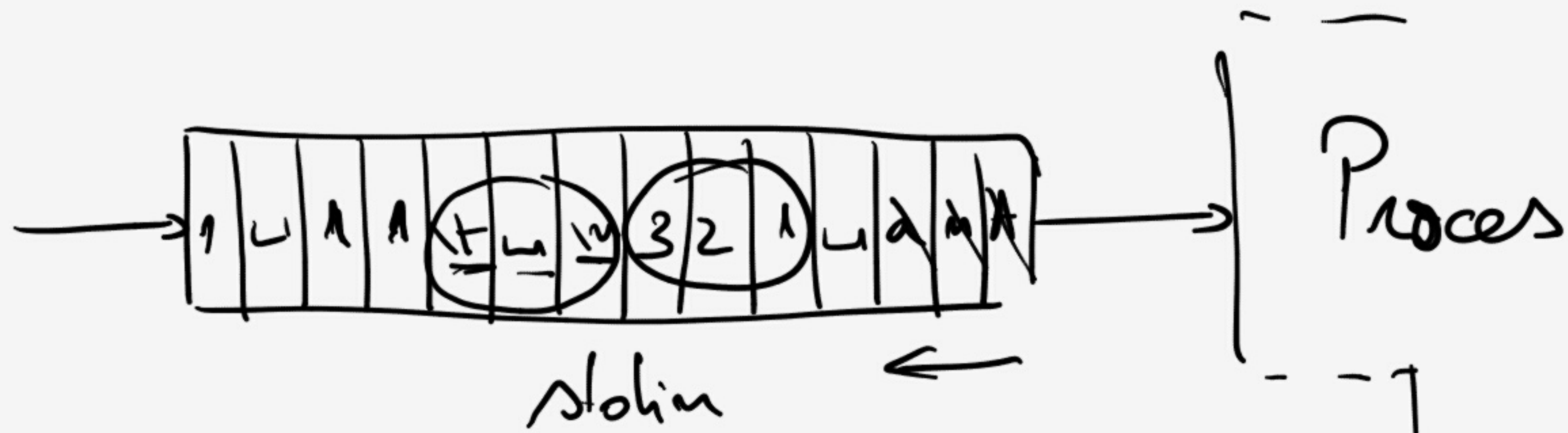
```
fclose(file_descriptor);
```

```
return 0;
```

```
}
```

w - write (dacă nu există, se creează);
r - read
a - append

C:\Users\lan\Desktop\output.txt



fscanf(...)
input.txt

```
int n;  
scanf("%d", &n);  
// n = 123  
scanf("%d", &n);  
// n = 11
```

```
#...
void f(int a) {
    a = 20;
}
int main() {
```

```
    int a = 30;
    f(a);
    → printf("%d\n", a);
    return 0;
}
```

```
void f(int*data_type *p) {
    *p = VALUE1;
}
```

```
int main() {
    data_type a = VALUE0;
    f(&a);
    printf("%d\n", a);
    return 0;
}
```

Complexitate

① a_0, a_1, \dots, a_{n-1}

Fie v , care este i, a.i. $a_i = v$

$O(1)$ \rightarrow best case
 $\rightarrow O(n)$ - worst case
 $O(\frac{n}{2}) \equiv O(n)$

② Printarea toate un. de la 0 la $n-1$. $\rightarrow O(n)$

③ Fie $v \in [n]$, printarea primului element. $\rightarrow O(1)$

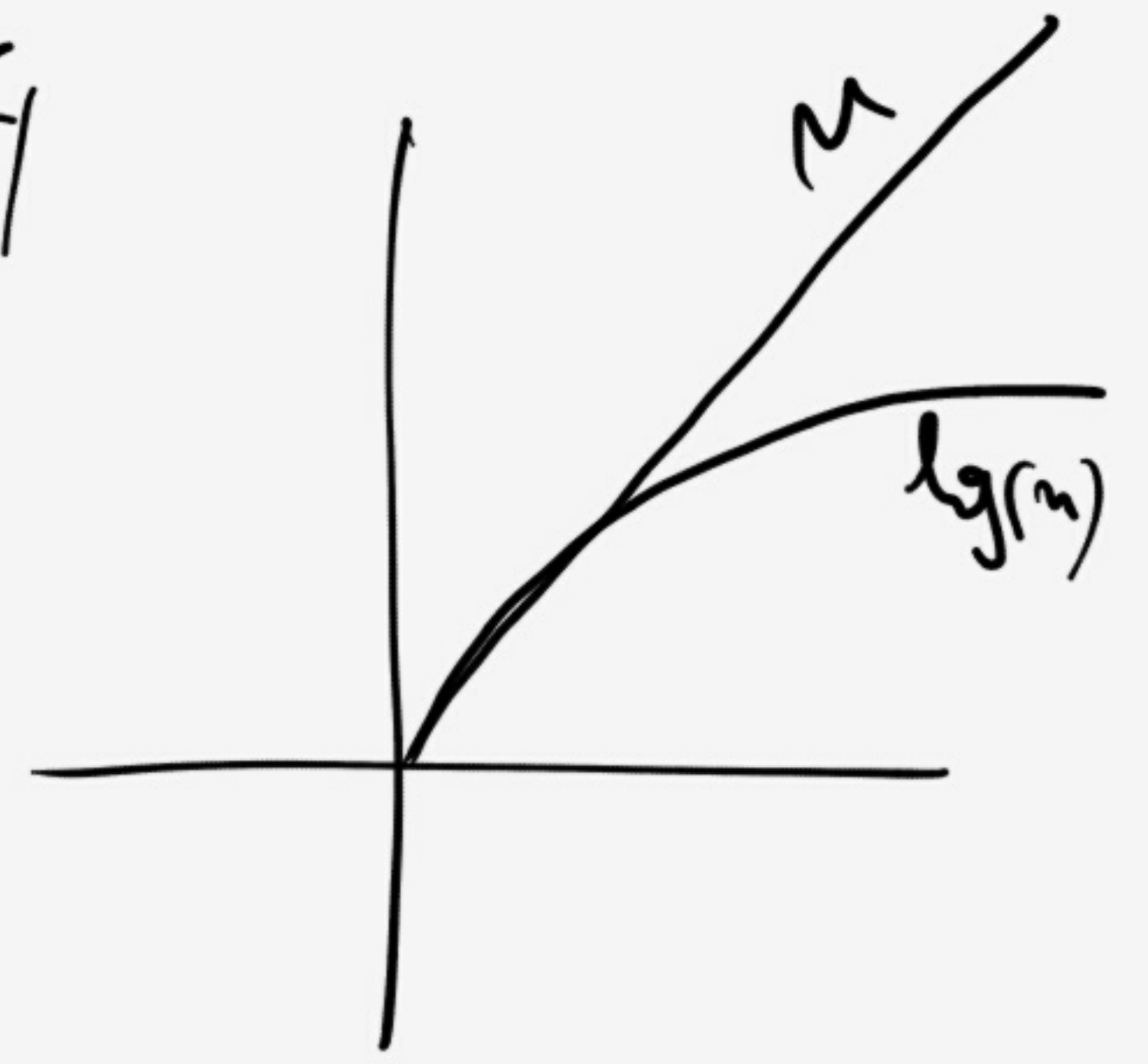
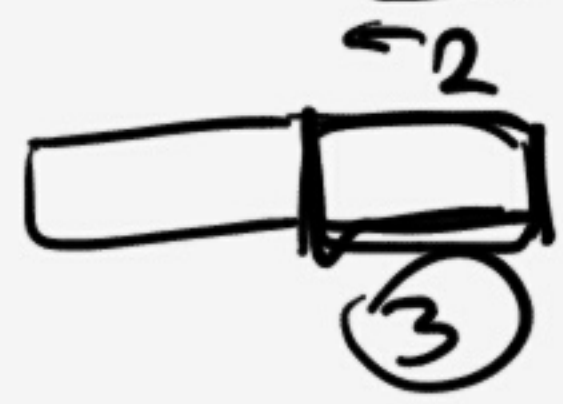
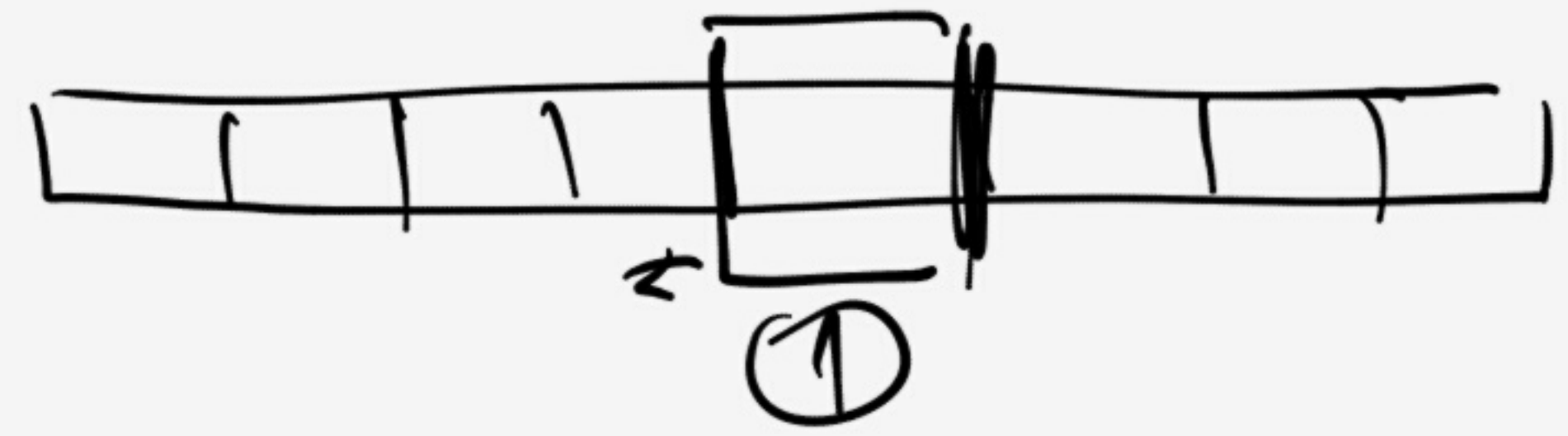
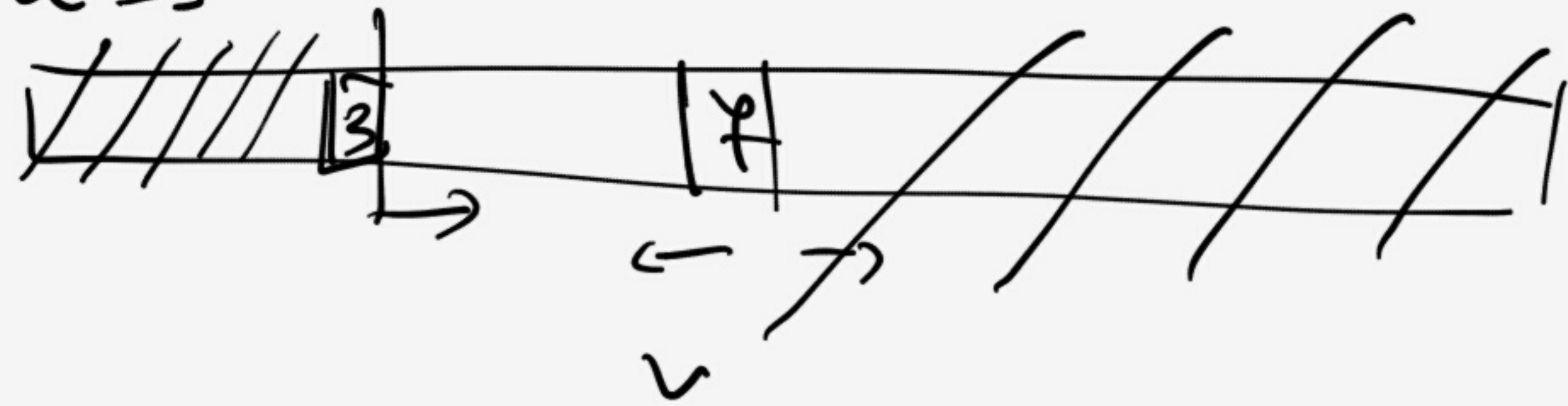
④ Fie $v \in [n]$, printarea toate perechile $(v_i, v_k), v_i, v_k \in v \rightarrow O(n^2)$

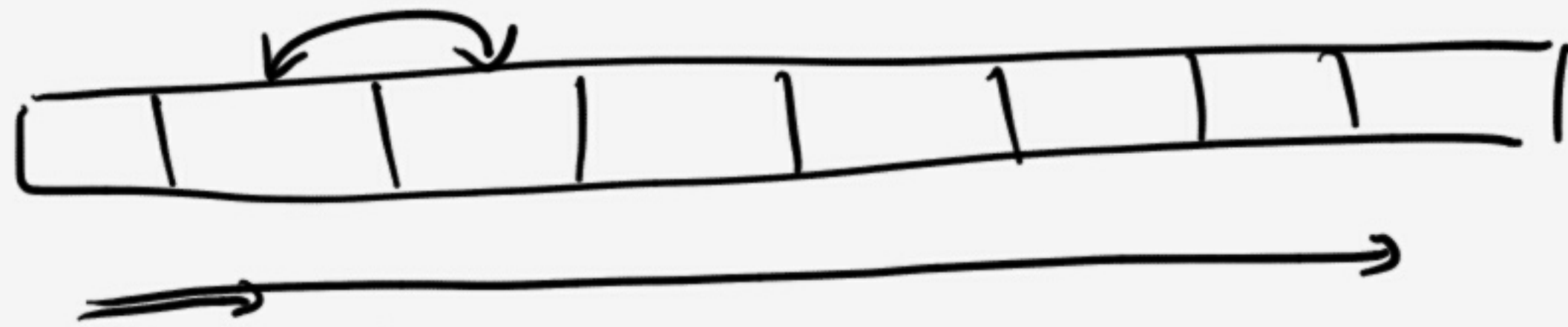
⑤ Căutare binară

$O(\log_2 n)$

v_0, v_1, \dots, v_n , $v_i \geq v_j \Leftrightarrow i > j$

$a = 5$





read n
 read $v[n]$

ob

done \leftarrow true

for $i = 0 \dots n-2$

if $v[i] > v[i+1]$

tmp $\leftarrow v[i]$

$v[i] \leftarrow v[i+1]$

$v[i+1] \leftarrow$ tmp

done \leftarrow false

while done == false

best case $\rightarrow O(n+1) \equiv O(n)$

worst case $\rightarrow O(n^2)$

average case $\rightarrow O\left(\frac{n^2+n}{2}\right)$

$$\equiv O\left(\frac{n^2}{2} + \frac{n}{2}\right)$$

$$\equiv \underline{O\left(\frac{n^2}{2}\right)} \equiv \underline{\underline{O(n^2)}}$$

complexitate $\left\{ \begin{array}{l} \text{timp} \rightarrow \text{cum crește timpul de execuție} \\ \text{spațiu} \rightarrow \text{cum crește memoria consumată} \end{array} \right.$
en cantitatea de date de intrare
en cantitatea de date de intrare

Combining set

$$v = [\dots]$$

$v_i \rightarrow$ de tip int $\Rightarrow 32$ biti $= 2^{32} \rightarrow 4$ Giga octeți

$v_i \rightarrow$ de tip char $\Rightarrow 8$ biti $= 2^8 = 256$

$$v_0 \stackrel{256}{=} \rightarrow \parallel \Rightarrow O(n + 2^k) \Leftrightarrow O(n \lg n)$$