

UNIVERSIDADE FEDERAL DO CEARÁ

CK0215 - 2019.1 - T01

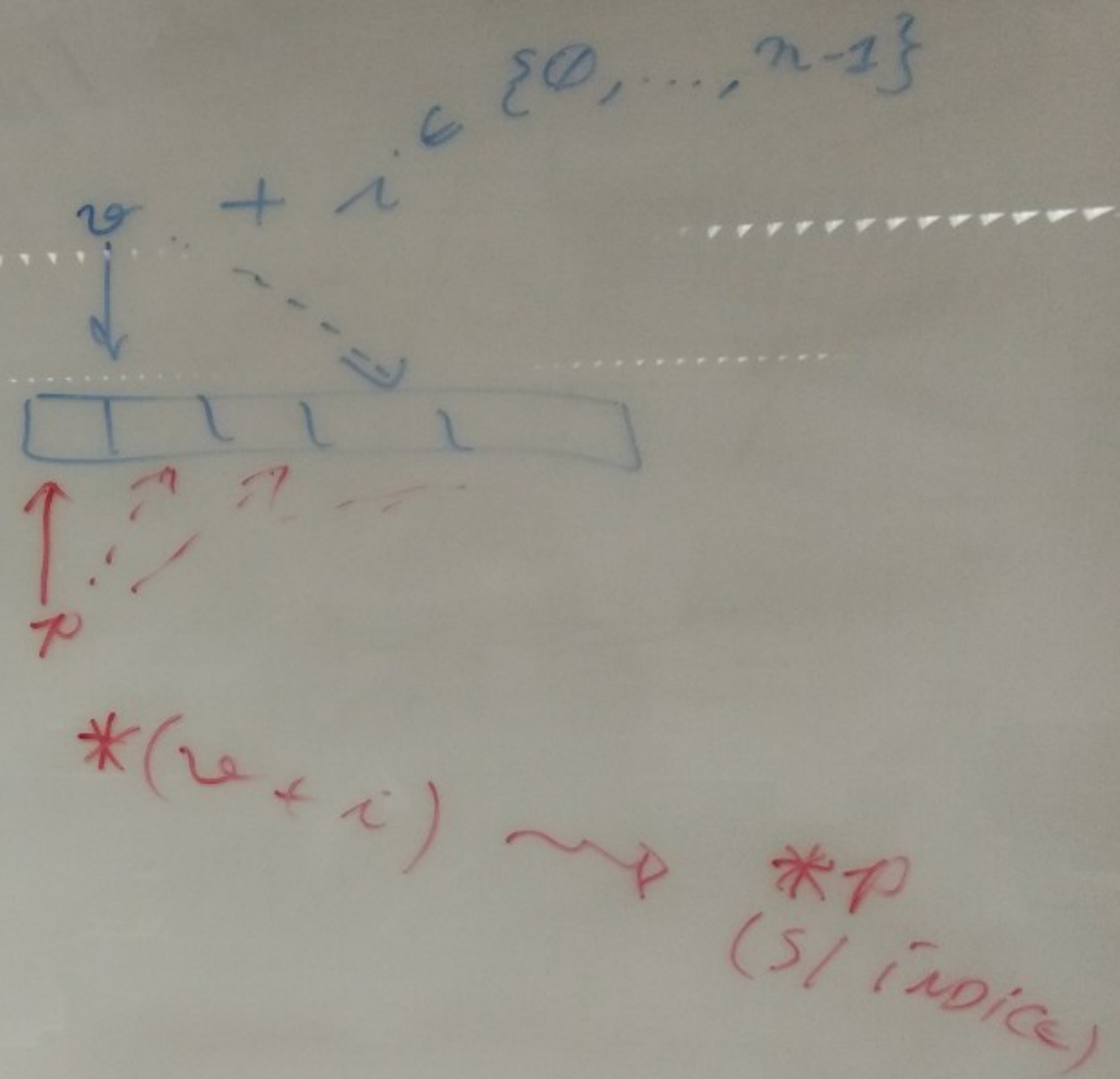
LABORATÓRIO DE PROGRAMAÇÃO

AULA 5 - 11/03/2019

0. OBSERVAÇÃO:

```
for (int i = 0; i < n; ++i)
{ *(&v + i) = ...; }
```

$v[i]$



1. QUICKSORT:

0	1	2	3	4	5	6	7	8	9	10
8	-9	3	0	-2	5	7	1	-4	4	-1



0	1	2	3	4	5	6	7	8	9	10
-1	-4	4	1	-2	0	3	-9	5	7	8

0	1	2	3	4	5	6	7	8	9	10
-9	-1	4	-4	1	-2	3	0	5	7	8

0	1	2	3	4	5	6	7	8	9	10
-9	-1	4	-4	1	-2	3	0	5	7	8

$[0, 10] \in$

$[0, 7] \in$

$[1, 7] \in$

$[1, 4]$

0	1	2	3	4	5	6	7	8	9	10
-9	-1	-4	-2	0	1	4	3	5	7	8

0	1	2	3	4	5	6	7	8	9	10
-9	-1	-4	-2	0	1	4	3	5	7	8

0	1	2	3	4	5	6	7	8	9	10
-9	-1	-4	-2	0	1	4	3	5	7	8

2. SUBTRAÇÃO DE PONTEIROS: SE p E q

APONTAM RESPECTIVAMENTE PARA OS ELEMENTOS $v[i]$ E $v[j]$ DE UM VETOR " v " DE " n " ELEMENTOS (INCLUINDO POSSIVELMENTE O "ELEMENTO" $v[n]$), ENTÃO:

$$p - q \stackrel{\text{def}}{=} i - j.$$

$++p$

3. CHAMADA DE CAUDA: SE O ÚLTIMO

COMANDO EXECUTADO POR UMA FUNÇÃO

" f " É CHAMAR UMA FUNÇÃO " g ", ENTÃO

ESSA CHAMADA É UMA CHAMADA DE

CAUDA EM f . SE f E g FOREM A

MESMA FUNÇÃO, ENTÃO TRATA-SE DE UMA

CHAMADA RECURSIVA DE CAUDA.

4. Exemplo:

```

void imprimir-1 (int *v, int n)
{
    for(;;) cout << *v;
    if (n > 1) { imprimir-1(v+1, n-1); }
}
v = v+1; n = n-1;
    
```

12	
13	7
14	
15	
16	
17	-2
18	
19	
20	
21	0
22	
23	
24	

$v: [7, -2, 0]$

3ª cham.: v n
 20 1

2ª chamada: v n
 16 2

1ª chamada: v n
 12 3

5. Exemplo:

```
void imprimir_2 (int *v, int n)
{
    for(i; i)
    {
        cout << *v;
        if(n > 1) { ++v; --n; }
        ELSE BREAK;
    }
}
```