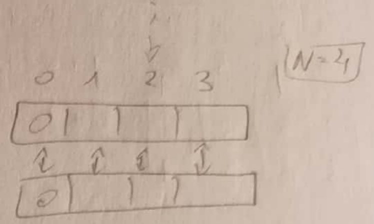
Teste 1 – 2022

(1)

```
int difInd (int a[], int b[], int N) {
    // pre: N > 0
    int i=0;
    i=0;
    while (i<N  && a[i] == b[i])
        i = i+1;

    // pos: (0<=i<N && forall_{0<=k<i} a[k] == b[k] && a[i] != b[i]) ||
    //      (i==N && forall_{0<=k<N} a[k] == b[k])
    return i;
}
```



(2)

```
int Sacos (int p[], int N, int c) {
    // pre: N>0 && c>0 && forall_{0<=k<N} 0 < p[k] <= c
    int r=1, t = p[0], i=1, j=0;
    // inv: 1 <= r <= i && r*C >= Sum_{0<=k<i} p[k] && i<=N && t<=C
                                        v: N-i+1
    while (i<N) {
        if (t + p[i] > c) {
            j=i; t=0; r=r+1;
        }
        t = t + p[i];
        i = i+1;
    }
    return r;
    // pos: 0 < r <= N && r*c >= Sum_{0<=k<N} p[k]
}
```

exemplo N = 6
C = 12

6-1+1 = 6
6-2+1 = 5
6-3+1 = 4
6-4+1 = 3
6-5+1 = 2
6-6+1 = 1

> 0 ✓

decresce ✓

exp. int ✓

Ver exemplo enunciado

$C = 12$

0	1	2	3	4	5
3	6	2	1	5	7

$N = 6$

r	i	j	t	t+p[i]
1	1	0	3	3+6 < 12 ✓
1	2	0	9	9+2 < 12 ✓
1	3	0	11	11+1 < 12 ✗
2	4	3	~~0~~ 1	1+5 < 12 ✓
2	5	3	6	6+7 < 12 ✗
3	6	5	~~0~~ 7	

I: $1 \leq r \leq i$ && $r * C >= Sum\_\{0 <= k < i\}\ p[k]$

→ Testar $\boxed{\text{Init}}$ ✓

{N>0 && C>0 && forall_{0<=k<N} 0 < p[k] <= C} r=1; t=p[0]; i=1; j=0; {I}

N>0 && C>0 && forall_{0≤k<N} 0<p[k]<=C ⟹ 1 ≤ 1 ≤ 1 && 1 * C >= Sum_{0<=k<1} $\underbrace{p[k]}_{p[0]}$ ✓

→ Testar $\boxed{\text{Util}}$ I ∧ ¬c ⟹ Q ✓

{1 ≤ r ≤ i && r * C >= Sum_{0≤k<i} p[k] ∧ i>=N} ⟹ 0 < r <= N && r * C >= Sum_{0<=k<N} p[k]

falta acrescentar ao Inv: i <= N

→ Testar $\boxed{\text{Pres}}$

{I ∧ c} if (t+p[i] > C) {j=i; t=0; r=r+1;} {R} — if

R ⟹ R'
{R} t=t+p[i] {R'} — Atr

R' ⟹ 1 ≤ r ≤ i+1 && r * C >= Sum_{0<=k<i+1} p[k]
{R'} i=i+1; {I} — Atr

Seq

{I ∧ c} if (t+p[i] > C) {j=i; t=0; r=r+1;} t=t+p[i]; i=i+1; {I}

$1 \le r \le i$ && $r*C$ >= Sum_{0<=k<i} p[k] $\wedge\ i<N \wedge t+p[i] > C \Rightarrow 1 \le r+1 \le i+1$ && $\underbrace{(r+1)*C}_{r*C+C}$ >= Sum_{0<=k<i+1} p[k]

$1 \le r \le i < N$
$r*C$ >= Sum_{0<=k<i} p[k]
$\wedge$
$t+p[i] <= C \Rightarrow$ $1 \le r \le i+1$ && $r*C$ >= Sum_{0<=k<i+1} p[k]

$i \ge N$

Seq

$$\frac{\{I \wedge c \wedge t+p[i] > C\}\ j=i;\ t=0;\ r=r+1;\ \{R\} \qquad \{I \wedge \neg c \wedge t+p[i] <= C\} \Rightarrow R}{\{I \wedge c\}\ \text{if } (t+p[i] > C)\ j=i;\ t=0;\ r=r+1;\ \{R\}}\ \text{if}$$

$r*C$ >= Sum_{0<=k<i} p[k]

$C + r*C$ >= $\underbrace{\text{Sum\_\{0<=k<i+1\} p[k]}}_{\text{Sum\_\{0<=k<i\} p[k] + p[i]}}$?!?

e Sabemos que $C < t+p[i]$

logo se $t \le C$ tem-se que $C >= p[i]$

logo falta acrescentar ao I: $\underline{t \le C}$

③ $MC_{Sacos} \rightarrow t + p[i] <= C$ nº op. aritméticas

$PC_{Sacos} \rightarrow t + p[i] > C$

$$T^{PC}_{Sacos}(N) = \sum_{i=0}^{N-1} 4 = \underline{\underline{4N}} = \underline{\Theta(N)}$$

$$T^{MC}_{Sacos}(N) = \sum_{i=0}^{N-1} 3 = 3 \cdot \sum_{i=0}^{N-1} 1 = 3 * N = \underline{\underline{3N}} = \underline{\Theta(N)}$$

④

```c
int ensaca (int p[], int N, int C){
    int x, t;
    if (N==0) return 0;
    x = ensaca(p+1, N-1, C);
    if(p[0] <= C){
        t = 1 + ensaca(p+1, N-1, C-p[0]);
        if (t>x) x=t;
    }
    return x;
}
```

$\textcircled{+}\ 2 + 2T(N-1)$

nº acessos ao array

MC → chamar a função ensaca N vezes e $p[0] > C$

PC → " " " e $p[0] <= C$

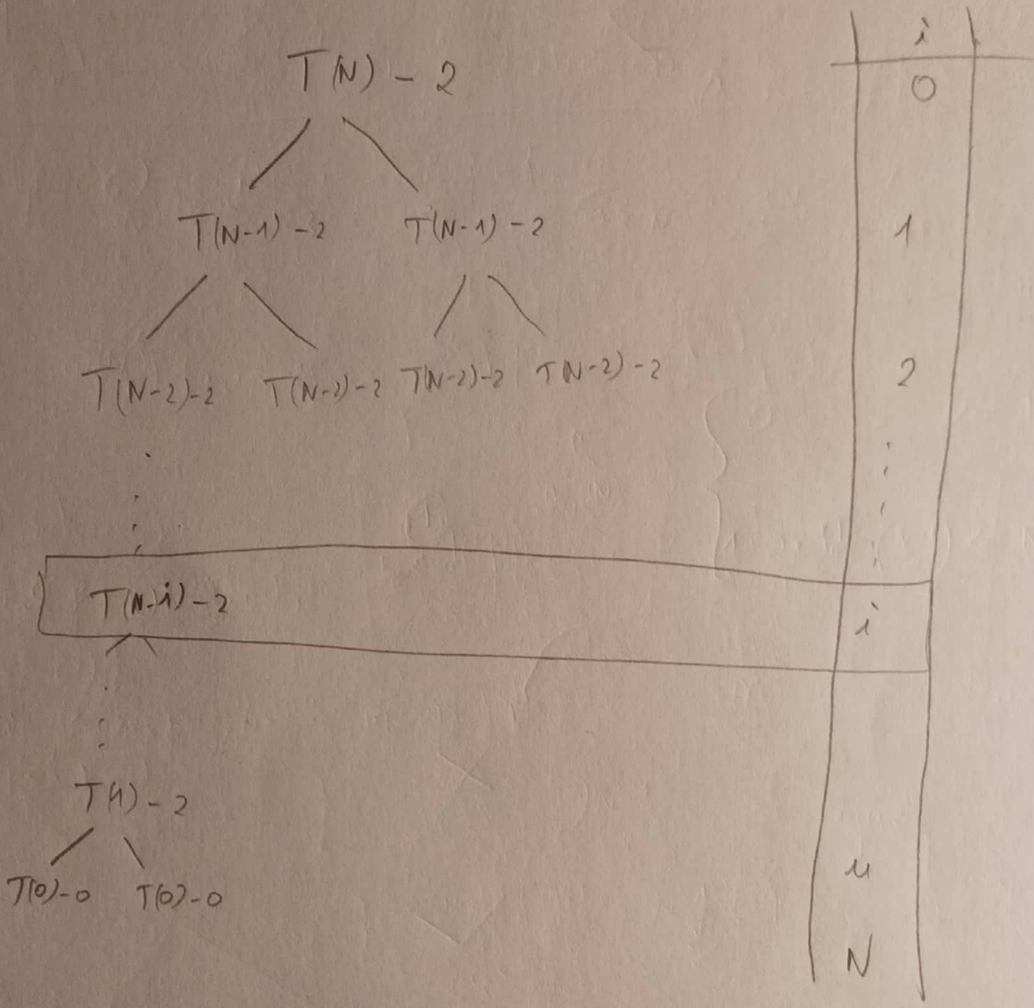
$$T^{MC}_{ensaca}(N) = \begin{cases} 0 & \Leftarrow N == 0 \\ 1 + T_{ensaca}(N-1) & \Leftarrow N > 0 \end{cases}$$

$$T^{MC}_{ensaca}(N) = \sum_{i=0}^{N-1} 1 + 0 = N - 1 - 0 + 1 = \underline{\underline{N}} = \underline{\Theta(N)}$$

$1 = N - u \Leftrightarrow u = N - 1$

	i
$T(N) - 1$	0
$T(N-1) - 1$	1
⋮	⋮
$T(N-i) - 1$	i
$T(1) - 1$	u
$T(0) - 0$	N

$$T^{PC}_{TemSaca}(N) = \begin{cases} 0 & \Leftarrow N==0 \\ \underbrace{2 + 2\,T_{TemSaca}(N-1)}_{\oplus} & \Leftarrow N>0 \end{cases}$$



$1 = N-u$

$\Leftrightarrow u = N-1$

Soma dos 2 a cada i-nível

i	Soma
0	2
1	4
2	8
⋮	
i	2^{i+1}

$$T^{PC}_{TemSaca}(N) = \sum_{i=0}^{N-1} 2^{i+1} + 0$$

$$= 2 \cdot \sum_{i=0}^{N-1} 2^i$$

$$= 2 \cdot \frac{1-2^{N-1+1}}{1-2} = 2 \cdot \frac{1-2^N}{-1} = 2.(2^N-1) = 2^{N+1}-1$$

$$= \Theta(2^N)$$