

GABARITO SEMANA 1

Monitora: Adriana Jacoto Unger

Questão 1 de 2

0.5 Pontos

Dado um número x e um vetor $A[1... n]$, escreva em pseudocódigo um algoritmo recursivo de busca binária para encontrar j tal que $A[j-1] < x \leq A[j]$. A função deve ter os seguintes parâmetros:

- Vetor A com n elementos
- Inteiro e que representa a posição do lado esquerdo do vetor
- Inteiro d que representa a posição do lado direito do vetor
- Valor x com o elemento que se deseja encontrar

Solução 1:

buscaBinaria (A, e, d, x)

```
se (e <= d) // há pelo menos um elemento no subvector
    m ← (e+d)/2
    se (x = A[m]) devolva m
    se (x > A[m])
        devolva buscaBinaria (A, m+1, d, x)
    senão devolva buscaBinaria (A, e, m-1, x)
senão devolva e // ou d+1
```

Assim, a primeira chamada ao método deve ser buscaBinaria (A, 1, n, x).

Se o vetor A não tem elementos, a chamada deve ser buscaBinaria (A, 1, 0, x).

Solução 2:

buscaBinaria (A, e, d, x) // <https://www.ime.usp.br/~pf/algoritmos/aulas/bubi.html>

```
se (e == d-1) devolva d; // não há elementos no subvector
senão {
    m = (e + d)/2;
    se (A[m] < x)
        devolva buscaBinaria (A, m, d, x)

    senão
        devolva buscaBinaria (A, e, m, x)
```

Em que e representa uma posição antes do lado esquerdo do vetor e d uma posição depois. Assim, a primeira chamada ao método deve ser buscaBinaria (A, 0, n+1, x).

Se o vetor A não tem elementos, a chamada deve ser buscaBinaria (A, 0, 1, x).

0.5 Pontos

Questão 2 de 2

Determine se cada sentença é verdadeira(V) ou falsa(F):

() Na pilha de execução do algoritmo abaixo chamado com $A=[3,5,4,-1]$ e $n=4$, serão criados apenas espaços de trabalho para $SomaPositivos(A,4)$, $SomaPositivos(A,3)$, $SomaPositivos(A,2)$ e $SomaPositivos(A,1)$.

```
SomaPositivos (A, n)
se n = 0
    devolva 0
senão s ← SomaPositivos (A, n - 1)
    se A[n] > 0
        devolva s + A[n]
    senão devolva s
```

Solução: A sentença é falsa (F) pois será criada também a chamada $SomaPositivos (A, 0)$ que corresponde ao caso base.

() Algoritmos iterativos em geral possuem maior custo de memória que sua versão recursiva.

Solução: A sentença é falsa (F) pois algoritmos recursivos em geral gastam mais memória do que sua versão iterativa, devido à alocação do espaço de trabalho para cada chamada recursiva na pilha de execução.