



## Frequência 2

- (3 valores) Recursividade. Implemente uma função recursiva em linguagem C que mostre no écran o conteúdo de uma string invertido.

**Protótipo:** void mostraString(char \*st);

- Considere a seguinte função:

```
void função(int *v, int tot){
    int i, j;
    for (i=0;i<tot;i++){
        if (i%2==0)
            continue;
        for (j=i+1;j<tot;j++)
            v[j]=v[j]+3;
    }
    for (j=0;j<tot;j++)
        printf("%d",v[j]);
}
```

- (2 valores) Exemplifique dois vectores com pelo menos 4 elementos, correspondentes ao melhor e pior caso em termos do número de passos a executar pela função.
  - (2 valores) Calcule a complexidade computacional da função apresentada, utilizando a notação *Big-O*. Deve apresentar os cálculos necessários para a obtenção da resposta.
- Considere que um número complexo é representado através da seguinte estrutura de dados:

```
typedef struct{
    float real;
    float imaginário;
}Complexo;
```

- (3 valores) Implemente uma função que calcule e devolva um vector correspondente às respectivas normas de um vector de números complexos. Exemplo:  $|1+2i| = (1^2+2^2)^{0.5}$
- Protótipo:** float\* norma(Complexo \*v, int totV);
- (3 valores) Utilize a função da alínea anterior para codificar uma função que ordene decrescentemente um vector de números complexos, utilizando por critério a respectiva norma.
- Protótipo:** void ordenaVector(Complexo \*v, int totV);
- (4 valores) Pesquisa Binária. Implemente um programa em linguagem C que peça ao utilizador um vector de "n" números complexos ("n" definido pelo utilizador), os ordene segundo o respectivo valor da norma e finalmente mostre no écran os números complexos recebidos com norma igual a determinado valor (definido pelo utilizador). Devido a preocupações de custo computacional, deve implementar mecanismos de pesquisa binária, tendo em consideração que vários elementos no vector podem ter normas iguais.
- (3 valores) Suponha que uma função que implementa o algoritmo de ordenação "Quicksort" recebe o seguinte vector de inteiros: [1 4 2 6 7 9 0 3 8 5]. Mostre o estado do vector após 2 iterações do algoritmo, considerando que este escolhe sempre como pivot o primeiro elemento do vector.