



Lista Complementar – Vetores, Ponteiros, Matrizes, Strings e Recursão

1. Geração do Vetor

Escreva um programa que crie um vetor para armazenar 500 números inteiros. O programa deve chamar a função `preencheVetor`, que preenche o vetor com valores aleatórios no intervalo de 1 a 60.

2. Soma Iterativa com Índices

Escreva uma função iterativa, utilizando índices, que receba um vetor e seu tamanho, calcule e retorne a soma de seus elementos. Imprima o resultado no programa principal.

3. Soma Iterativa com Ponteiros

Escreva uma função iterativa, utilizando ponteiros, que receba um vetor e seu tamanho, calcule e retorne a soma de seus elementos. Imprima o resultado no programa principal.

4. Soma Recursiva com Índices

Escreva uma função recursiva, utilizando índices, que receba um vetor e seu tamanho, calcule e retorne a soma de seus elementos. Imprima o resultado no programa principal.

5. Soma Recursiva com Ponteiros

Escreva uma função recursiva, utilizando ponteiros, que receba um vetor e seu tamanho, calcule e retorne a soma de seus elementos. Imprima o resultado no programa principal.

6. Buscar Maior Valor e Sua Posição

Crie um programa que encontre o maior valor em um vetor de inteiros e sua respectiva posição. Para isso, resolva as seguintes variações:

- (a) Escreva uma função iterativa, utilizando índices, que receba o vetor e seu tamanho, e retorne o maior valor e sua posição.
- (b) Escreva uma função iterativa, utilizando ponteiros, que receba o vetor e seu tamanho, e retorne o maior valor e sua posição.
- (c) Escreva uma função recursiva, utilizando índices, que receba o vetor e seu tamanho, e retorne o maior valor e sua posição.
- (d) Escreva uma função recursiva, utilizando ponteiros, que receba o vetor e seu tamanho, e retorne o maior valor e sua posição.

7. Verificar Palíndromo (com String)

Escreva uma função que recebe uma string e verifica se ela é um palíndromo (ex: “arara”, “ovo”, “ana”). Ignore espaços e diferenças entre maiúsculas/minúsculas. Use índice ou ponteiro para percorrer a string.

8. Contar Ocorrências de um Valor (Recursivo com Índices)

Crie uma função recursiva para contar quantas vezes um valor aparece em um vetor de inteiros. Use o protótipo:

```
int contarOcorrencias(int vetor[], int tamanho, int x, int i);
```

9. Inverter Vetor no Próprio Espaço (Com Ponteiros)

Crie uma função que inverta os elementos de um vetor no próprio espaço de memória, usando ponteiros. Não utilize vetor auxiliar. Use o protótipo:

```
void inverter(int *vetor, int n);
```

10. Acessar Elementos de uma Matriz com Ponteiros

Dada uma matriz 3x3 de números inteiros, escreva uma função que imprime os elementos da matriz utilizando exclusivamente aritmética de ponteiros (sem usar `[]`). Use o protótipo:

```
void imprimirMatriz(int *matriz, int linhas, int colunas);
```

Dica: para acessar o elemento da linha `i` e coluna `j`, use: `*(matriz + i * colunas + j)`

11. Contar Vogais em uma String (com Ponteiros e Recursão)

Implemente uma função recursiva que recebe uma string e retorna o número de vogais presentes nela. Utilize ponteiros na implementação da função. Use o protótipo:

```
int contarVogais(char *str);
```

12. Contar Ocorrências de uma Substring (com Ponteiros e Recursão)

Escreva uma função recursiva que conte quantas vezes uma substring ocorre dentro de uma string maior. A função deve ignorar diferenças entre maiúsculas e minúsculas e permitir sobreposição. Use o protótipo:

```
int contarSubstring(char *texto, char *sub);
```

Exemplo: texto: “ana e AnA foram passear”, substring: “ana” → resultado: 3.