

# Lista de Exercícios

## ICC 1 – Ponteiros

Professor Cláudio Fabiano de Motta Toledo

- Todos os exercícios devem ser resolvidos utilizando **ponteiros, alocação dinâmica e aritmética de ponteiros**. Não utilize variáveis globais e não desperdice espaço em vetores na função `main()`. O uso de vetores temporários é permitido.
- Libere a memória dos ponteiros criados em todos os exercícios no final do programa.
- Funções que criam vetores devem retorná-los, e não receber o vetor como parâmetro.
- Os programas devem ser modularizados: **cada tarefa dos exercícios deve ser feita em uma função separada da `main()`**.

1. (Fácil) Faça um programa que leia um número  $n$  e:

- Crie e leia um vetor de inteiros de  $n$  posições;
- Leia um número  $x$  e conte quantas vezes ele aparece num vetor de inteiros.

Na sua função `main()`, mostre o resultado encontrado.

2. (Fácil) Faça um programa que leia um número  $n$  e:

- Crie e leia um vetor de inteiro de  $n$  posições;
- Conte os múltiplos de um número inteiro  $x$  num vetor e mostre-os na tela.

Na sua função `main()`, mostre quantos múltiplos foram encontrados.

3. (Fácil) Faça um programa que leia um número  $n$  e:

- Crie e leia um vetor de inteiros de  $n$  posições;
- Conte a quantidade de números pares num vetor de inteiros e mostre-os na tela;
- Conte a quantidade de números ímpares num vetor de inteiros e mostre-os na tela.

Na sua função `main()`, mostre a quantidade de números pares e ímpares encontrados.

4. (Fácil) Faça um programa que leia um número  $n$  e:

- Crie e leia um vetor de inteiros de  $n$  posições;
- Inverta um vetor. Crie uma função de troca de variáveis para isso.

Na sua função `main()`, mostre o vetor após a operação.

5. (Fácil) Faça um programa que leia uma palavra (máximo de 50 letras) e:

- Some 1 no valor ASCII de cada caractere da palavra.

Na sua função `main()`, imprima a string resultante.

6. (Fácil) Faça um programa que leia dois números  $n$  e  $m$  e:

- Crie e leia uma matriz de inteiros  $n \times m$ . Use ponteiro simples;
- Localize os três maiores números de uma matriz e mostre a linha e coluna onde estão. **Pode** alterar a matriz original.

7. (Fácil) Faça um programa que leia um número  $n$  e:

- Crie e leia um vetor de inteiros de  $n$  posições;
- Na mesma função (diferente da `main()`), conte a quantidade de números pares, ímpares e negativos e retorne via parâmetros por referência.

Na sua função `main()`, imprima o vetor e os três resultados obtidos.

8. (Fácil) Faça um programa que leia um número  $n$  e:

- Crie e leia um vetor de inteiros de  $n$  posições;
- Encontre a moda do vetor (o número que mais aparece) e quantas vezes ele aparece. Em caso de empate, retorne o primeiro encontrado.

Na sua função `main()`, imprima o número encontrado e quantas vezes ele aparece.

9. (Médio) Faça um programa que leia um número  $n$  e:

- Crie e leia um vetor de inteiros de  $n$  posições;
- Crie um vetor com os números pares do outro vetor;
- Crie um vetor com os números ímpares do outro vetor.

Na sua função `main()`, imprima os três vetores criados.

10. (Médio) Faça um programa para saber quantos alunos cursam duas disciplinas simultaneamente. Ele deve ler o número de inscritos em cada disciplina e:

- Criar e ler um vetor de inteiros que sejam os números de matrículas dos alunos inscritos numa disciplina. Não há números repetidos;
- Contar quantos números de matrículas aparece simultaneamente em duas disciplinas e mostrar na tela.

**Dica:** não faça sua função criar e ler os dois vetores ao mesmo tempo. Crie apenas um e use-a duas vezes.

11. (Médio) Faça um programa que leia dois números  $n$  e  $m$  e:

- Crie e leia um vetor `num` de inteiros de  $n$  posições;
- Crie e leia um vetor `div` de inteiros de  $m$  posições;
- Crie e um vetor de inteiros e preencha-o com a quantidade de divisores de cada número em `num` que estão em `div`.

Na sua função `main()`, imprima os três vetores criados. Exemplo:

| <code>num</code>  | <code>div</code> | <code>vetorC</code> |
|-------------------|------------------|---------------------|
| 5, 10, 20, 30, 60 | 1, 2, 3, 4       | 1, 2, 3, 3, 4       |

12. (Médio) Faça um programa que leia o número de vendedores de uma loja e:

- Crie e leia o total de vendas feitas por cada vendedor (em reais)
- Crie e leia um vetor de taxas de comissão recebida por venda de cada vendedor (como porcentagem)
- Crie e leia um vetor de comissão recebida por cada vendedor

Na sua função `main()`, mostre os dados no formato: índice, venda, taxa, valor recebido. Exemplo:

| Índice | Vendas | Taxa | Recebido |
|--------|--------|------|----------|
| 0      | 150.0  | 50.0 | 225.0    |
| 1      | 200.0  | 25.0 | 250.0    |
| 2      | 500.0  | 10.0 | 550.0    |

13. (Médio) Faça um programa que leia dois números  $n$  e  $m$  e, **usando ponteiros duplos**:

- Crie e leia uma matriz  $n \times m$  de inteiros;
- Crie e construa uma matriz transposta  $m \times n$  de inteiros.

Na sua função `main()`, mostre as duas matrizes.

**Dica:** lembre-se de que uma matriz é um “vetor de vetores”.

14. (Médio) Faça um programa que leia dois números  $a$  e  $b$  (positivos menores que 10000) e:

- Crie um vetor onde cada posição é um algarismo do número. A primeira posição é o algarismo menos significativo;
- Crie um vetor que seja a soma de  $a$  e  $b$ , mas faça-o usando apenas os vetores construídos anteriormente.

**Dica:** some as posições correspondentes. Se a soma ultrapassar 10, subtraia 10 do resultado e some 1 à próxima posição.

15. (Médio) Faça um programa que:

- Crie e leia uma palavra (máximo de 30 letras);
- Crie e construa outra palavra apenas com as consoantes da primeira.

Na sua função `main()`, imprima ambas as palavras.

16. (Médio) Faça um programa que:

- Crie e leia uma palavra `str` (máximo de 15 letras);
- Inverta esta palavra, **sem modificar a original**;

Ao final do programa, devem-se ter duas palavras: a original (`str`) e a invertida. Na sua `main()`, imprima-as e mostre uma mensagem indicando se ela é palíndromo ou não (ou seja, se a palavra invertida é igual à original).

17. (Médio) Faça um programa que:

- Crie e leia duas palavras  $s1$  e  $s2$  (máximo de 15 letras);
- Concatene essas duas palavras em uma única frase, com um espaço entre elas. **Não use** `strcat(...)`.

Na sua função `main()`, imprima o resultado da concatenação.

18. (Médio) Faça um programa que leia dois números  $n$  e  $m$  e:

- Crie e leia um vetor de inteiros de  $n$  posições;
- Crie e leia um vetor de inteiros de  $m$  posições;
- Crie e construa um vetor de inteiros que seja a interseção entre os dois vetores anteriores, ou seja, que contém apenas os números que estão em ambos os vetores. Não deve conter números repetidos.

Na sua função `main()`, imprima os três vetores criados.

19. (Médio) Faça um programa que leia dois números  $n$  e  $m$  e:

- Crie e leia um vetor de inteiros de  $n$  posições;
- Crie e leia um vetor de inteiros de  $m$  posições;
- Crie e construa um vetor de inteiros que seja a união entre os dois vetores anteriores, ou seja, que contém os elementos dos dois vetores (inclusive repetidos).

Na sua função `main()`, imprima os três vetores criados.

20. (Médio) Faça uma função que:

- Crie e leia duas palavras (máximo de 15 letras);
- Converta as letras de uma palavra para caixa alta (maiúsculas);
- Compare se as duas palavras lidas são iguais, independentemente da caixa.

Na sua função `main()`, imprima as duas palavras convertidas para caixa alta e uma mensagem informando o resultado da comparação. Você pode usar `strcmp(...)`, mas não `stricmp(...)`.

21. (Médio) Faça uma função que

- Crie e leia uma palavra (máximo de 30 letras);
- Leia um caractere e crie uma palavra que é a cópia da anterior, mas excluindo todas as ocorrências desse caractere.

Na sua função `main()`, imprima a palavra original e o resultado da operação.

22. (Difícil) Faça um programa que leia um número de alunos e:

- Crie e leia um vetor de nomes (máximo de 15 letras);
- Crie e leia um vetor de notas (tipo *float*), mostrando quem é o aluno correspondente ao pedir para digitar.

Na sua função `main()`, mostre os dois vetores criados com a nota na frente do nome do aluno correspondente. Exemplo:

|         |      |
|---------|------|
| Marcelo | 8.0  |
| Yukio   | 5.0  |
| Bressan | 6.5  |
| Hossomi | 10.0 |

**Dica:** vetor de nomes na verdade é uma matriz. Não use ponteiro duplo, use aritmética de ponteiros. Neste exercício, você pode desperdiçar espaço não utilizado pelos nomes.

23. (Difícil) Faça um programa que leia um número de alunos  $n$  e um número de provas  $p$ , e:

- Crie e leia um vetor tamanho  $n$  com os nomes de todos os alunos (máximo de 15 letras);
- Crie e leia uma matriz de notas  $n \times p$ ;
- Crie um vetor de  $n$  inteiros com a média de cada aluno.

Na sua função `main()`, imprima os dados lidos em forma de tabela (use `"\t"` para formatar), bem como a média de todos os alunos.

**Dica:** vetor de nomes na verdade é uma matriz. Não use ponteiro duplo, use aritmética de ponteiros. Neste exercício, você pode desperdiçar espaço não utilizado pelos nomes.

24. (Difícil) Faça um programa que leia quatro números  $a$ ,  $b$ ,  $c$  e  $d$ , que serão as dimensões de duas matrizes, e:

- Crie e leia uma matriz, dadas as dimensões dela;

- Crie e construa uma matriz que seja o produto de duas matrizes.

Na sua função `main()`, imprima as duas matrizes e o produto entre elas, se existir.

25. (Difícil) Faça um programa que leia palavras (até 10 letras) até que o usuário digite um ponto final, apenas. Seu programa deverá:

- Concatenar todas as palavras numa frase no sentido **inverso**, com um espaço entre elas. **Não use** `strcat(...)`.

Na sua função `main()`, imprima a frase resultante. Exemplo:

```
atirei
o
pau
no
gato
.
gato no pau o atirei
```

**Dica:** não leia todas as palavras de uma vez para só depois concatená-las.

26. (Difícil) Faça um programa que:

- Crie e leia duas palavras (máximo de 10 letras);
- Crie e construa uma nova palavra que seja a intercalação letra a letra da primeira com a segunda palavra.

Na sua função `main()`, imprima as duas palavras e o resultado da intercalação. Exemplos:

```
abcde          marcelo        ai             waeer
ABCDE          hossomi       trei           htv
aAbBcCdDeE    mhaorscseolmoi  atirei        whatever
```

27. (Difícil) Faça um programa que controle o estoque de mercadorias. Leia o número de produtos existentes e:

- Crie e leia um vetor com os códigos de cada produto (diferente de zero);
- Crie e leia um vetor com a quantidade de cada produto em estoque.

A seguir, o usuário deverá digitar uma seqüência de pedidos (código do produto e quantidade), até que ele digite o valor zero para código de produto. Seu programa, nessa parte, deverá:

- Verificar se é possível atender ao pedido integralmente (se há quantidade suficiente em estoque, ou se o código existe) e imprimir uma mensagem indicando;
- Atualizar o estoque caso o pedido for atendido.

Exemplo:

```
2
Produto [0]: 5
Produto [1]: 15
Quantidade [0]: 10
Quantidade [1]: 15
*** Pedido ***
Codigo: 5
Quantidade: 20
Nao e possivel atender esse pedido
*** Pedido ***
Codigo: 15
Quantidade: 10
Pedido atendido. Obrigado.
*** Pedido ***
Codigo: 15
Quantidade: 10
Nao e possivel atender esse pedido
*** Pedido ***
Codigo: 0
```

28. (Difícil) Faça um programa que construa uma lista de vetores de tamanho diferentes. A diferença disso para uma matriz é que cada vetor da matriz tem sempre o mesmo tamanho, compare:

|           |                   |
|-----------|-------------------|
| Matriz:   | Lista de vetores: |
| 1 5 3 8 6 | 5 1 4 6 3 2       |
| 1 2 3 4 5 | 2 1 2             |
| 2 4 5 6 7 | 4 6 7 8 9         |

Use o primeiro elemento de cada vetor da lista para armazenar o número de elementos (além do primeiro). O exemplo acima segue essa regra.

Para construir a lista de vetores, você deve ler o número  $n$  de vetores da lista e:

- Criar um vetor de ponteiros para inteiros de tamanho  $n$ .
- Ler o tamanho  $m_i$  do vetor  $v_i$ , criar e ler o vetor  $v_i$  (que terá tamanho  $(m+1)$  na verdade, pois o primeiro elemento armazenará  $m$ )
- Repetir o processo anterior para  $i = 0$  até  $(n-1)$ .

Na sua função `main()`, imprima todos os vetores da lista, exceto o primeiro elemento deles.