



## PARTE 1

- (1) Escreva um programa que recebe um vetor de inteiros, calcula a soma de todos os elementos do vetor e imprime a soma calculada.
- (2) Escreva um programa que recebe um vetor de inteiros positivos e substitui seus elementos de valor ímpar por -1 e os pares por +1. Em seguida, imprime o vetor alterado.
- (3) Escreva um programa que solicita 50 valores inteiros ao usuário, armazena esses valores em um arranjo chamado *vet1*. Depois, copia o conteúdo desse arranjo para um segundo arranjo chamado *vet2*. O programa deve imprimir os dois arranjos na tela.
- (4) Modifique o exercício anterior de forma que o conteúdo de *vet1* seja copiado para *vet2* de forma invertida. Por exemplo:  
$$vet1 = 1\ 5\ 0\ 4\ 9 \qquad vet2 = 9\ 4\ 0\ 5\ 1$$

Por fim, o programa deve imprimir *vet1* e *vet2*.
- (5) Faça um programa para criar um vetor *A* com 100 elementos inteiros e em seguida chamar um programa que preencha um vetor *B* de mesmo tipo e tamanho. Cada elemento do vetor *B* deverá ser o elemento correspondente de *A* multiplicado por 2. Por exemplo,  $B[0] = A[0] * 2$ ;  $B[1] = A[1] * 2$ ; ...;  $B[99] = A[99] * 2$   
  
Por fim, o programa deve imprimir *A* e *B*.
- (6) Faça um programa para criar um vetor *A* com 15 elementos inteiros e em seguida preencher um vetor *B* de mesmo tipo e tamanho. Cada elemento do vetor *B* deverá ser o quadrado do elemento correspondente de *A*.  
  
Por fim, o programa deve imprimir *A* e *B*.
- (7) Faça um programa para criar um vetor *A* com 15 elementos inteiros e em seguida preencher um vetor *B* de mesmo tipo e tamanho. Cada elemento do vetor *B* deverá ser o elemento correspondente de *A* multiplicado por seu índice.  
  
Por fim, o programa deve imprimir *A* e *B*.
- (8) Escreva um programa que preenche dois vetores, *A* e *B*, com inteiros positivos lidos do teclado e preenche um terceiro vetor *C* em que cada elemento é igual à soma dos elementos correspondentes dos vetores recebidos. Por fim o programa deve imprimir *C*.
- (9) Escreva um programa que recebe 20 valores inteiros positivos e os armazena em um vetor. Em seguida verifica se todos os elementos do vetor são pares. Se pelo menos um elemento for ímpar, o programa deve imprimir **FALSO**. Caso contrário, deve retornar **VERDADEIRO**.
- (10) Escreva um programa que recebe 20 valores inteiros positivos e armazena 10 desses valores no vetor *A* e 10 no vetor *B*. Em seguida, o programa deve preencher um terceiro vetor *C* de acordo com os seguintes critérios:  
$$C_i \text{ deverá receber } 1 \text{ quando } A_i \text{ for maior que } B_i;$$
$$C_i \text{ deverá receber } 0 \text{ quando } A_i \text{ for igual a } B_i;$$
$$C_i \text{ deverá receber } -1 \text{ quando } A_i \text{ for menor que } B_i.$$

Por fim, o programa deve imprimir *A*, *B* e *C*.

- (11) Faça um programa para criar um vetor  $A$  com 15 elementos inteiros lidos do teclado e em seguida preencher um vetor  $B$  de mesmo tamanho de acordo com os seguintes critérios:
- $B_i$  deverá receber 'a' quando  $A_i$  for menor que 7;
  - $B_i$  deverá receber 'b' quando  $A_i$  for igual a 7;
  - $B_i$  deverá receber 'c' quando  $A_i$  for maior que 7 e menor que 10;
  - $B_i$  deverá receber 'd' quando  $A_i$  for igual a 10; e
  - $B_i$  deverá receber 'e' quando  $A_i$  for maior que 10.
- O programa deverá então imprimir  $A$  e  $B$ .
- (12) Escreva um programa que recebe 20 valores inteiros positivos do teclado e armazena 10 desses valores no vetor  $A$  e 10 no vetor  $B$ . O programa deve preencher um terceiro vetor,  $C$ , que é a junção de  $A$  e  $B$ . Os primeiros elementos de  $C$  serão os elementos de  $A$  e os últimos elementos de  $C$  serão os elementos de  $B$ .
- Por fim, o programa deve imprimir  $A$ ,  $B$  e  $C$ .
- (13) Escreva um programa que recebe 20 valores inteiros positivos do teclado e armazena 10 desses valores no vetor  $A$  e 10 no vetor  $B$ . O programa deve preencher um terceiro vetor  $C$  que é a intersecção de  $A$  e  $B$ .
- Por fim, o programa deve imprimir  $A$ ,  $B$  e  $C$ .
- (14) Escreva um programa que leia 20 valores inteiros e os armazene em um arranjo  $A$ . Em seguida preencha outros dois arranjos  $B$  e  $C$  com os valores pares e ímpares, respectivamente, de  $A$ .
- Por fim, o programa deve imprimir  $A$ ,  $B$  e  $C$ .
- (15) Escreva um programa que leia 20 valores inteiros e os armazene no vetor  $V$ . Em seguida, o programa deve determinar o índice do menor elemento em  $V$  e imprimir este índice.

## PARTE 2

- (16) Escreva um programa que leia 21 valores inteiros, sendo que os primeiros 20 valores devem ser armazenados no vetor  $V$ . Em seguida, o programa deve determinar se o 21º valor lido está no vetor, isto é, se pelo menos um dos elementos do vetor é igual ao 21º valor lido. Se estiver, o programa deve imprimir a posição (o índice) em que o valor foi encontrado. Se não estiver, deve imprimir a mensagem: "O valor chave não foi encontrado".
- (17) Modifique a questão acima considerando que os 20 primeiros valores lidos são distintos. Antes de fazer a busca no vetor, o programa deve ordenar os dados no vetor e então fazer uma busca binária. Se o valor da chave estiver no vetor, o programa deve imprimir a posição (o índice) em que o valor foi encontrado. Se não estiver, deve imprimir a mensagem: "O valor chave não foi encontrado".

## PARTE 3

- (18) Escreva um programa que armazene de modo automático os seguintes valores em arranjos bidimensionais:

a)	0	1	2	3	4	5	6	7	8	9
	0	1	2	3	4	5	6	7	8	9
	0	1	2	3	4	5	6	7	8	9

b)	0	1	4	9	16	25	36	49	64	81
	1	2	5	10	17	26	37	50	65	82
	2	3	6	11	18	27	38	51	66	83
	3	4	7	12	19	28	39	52	67	84
	4	5	8	13	20	29	40	53	68	85

c)	0	0	0	0	0	0
	1	1	1	1	1	1
	2	2	2	2	2	2
	3	3	3	3	3	3

d)	-1	-1	-1	-1	-1	-1
	0	0	0	0	0	0
	-1	-1	-1	-1	-1	-1
	0	0	0	0	0	0

4 4 4 4 4 4  
5 5 5 5 5 5

-1 -1 -1 -1 -1 -1  
0 0 0 0 0 0  
-1 -1 -1 -1 -1 -1  
0 0 0 0 0 0  
-1 -1 -1 -1 -1 -1

(19) Escreva um programa que lê  $n \times m$  valores e os armazene em uma matriz real  $A_{m \times n}$ , linha a linha. Em seguida:

- i) imprime esta matriz.
- ii) calcula e imprime a soma dos elementos da linha  $i$  da matriz.
- iii) calcula e imprime o produto dos elementos da coluna  $j$  da matriz.

(20) Escreva um programa que lê  $n \times m$  valores e os armazene em uma matriz real  $A_{m \times n}$ , linha a linha. Em seguida:

- (a) imprime a matriz  $A$ .
- (b) cria uma matriz  $B$  igual à matriz  $A$  ampliada da seguinte maneira:
  - i. acrescenta uma coluna  $m$ , cujos elementos  $B[i][m]$  são as somas dos elementos da linha  $i$  da matriz  $A$ , respectivamente.
  - ii. acrescenta uma linha  $n$ , cujos elementos  $B[j][n]$  são as somas dos elementos da coluna  $j$  da matriz  $A$ , respectivamente.
- (c) imprime as matrizes  $A$  e  $B$ .

(21) Escreva um programa que armazene 24 inteiros em um arranjo bidimensional  $6 \times 4$  e imprime três inteiros:  $k$ ,  $Lin$  e  $Col$ . O inteiro  $k$  é o maior elemento de  $A$  e é igual a  $A[Lin][Col]$ .

Obs.: Se o elemento máximo ocorrer mais de uma vez, indique em  $Lin$  e  $Col$  qualquer uma das possíveis posições.

### PARTE 3

(22) Escreva as seguintes funções e um programa que as chame:

- (a) Uma função que lê, linha a linha, uma matriz real  $A_{m \times n}$ .
- (b) Uma função que imprime uma matriz real  $A_{m \times n}$ .
- (c) Uma função que calcula a soma dos elementos da linha  $i$  de uma matriz real  $A_{m \times n}$ .
- (d) Uma função que calcula o produto dos elementos da coluna  $j$  de uma matriz real  $A_{m \times n}$ .
- (e) Uma função que troca o conteúdo de duas variáveis.
- (f) Escreva uma função que recebe dois inteiros,  $i$  e  $j$ , uma matriz real  $A_{m \times n}$  e troca linha  $i$  pela linha  $j$ . Utilize a função do item anterior.

(23) Faça uma função **MAX** que recebe como entrada um inteiro  $n$ , uma matriz inteira  $A_{m \times n}$  e devolve três inteiros:  $k$ ,  $Lin$  e  $Col$ . O inteiro  $k$  é o maior elemento de  $A$  e é igual a  $A[Lin][Col]$ .

Por exemplo, se  $A = [371128534]$  então  $k = 8; Lin = 1; Col = 2$ .

Obs.: Se o elemento máximo ocorrer mais de uma vez, indique em  $Lin$  e  $Col$  qualquer uma das possíveis posições.

Faça um programa que, dado um inteiro  $n$  e uma matriz quadrada de ordem  $n$ , cujos elementos são todos inteiros positivos, imprime uma tabela onde os elementos são listados em ordem decrescente, acompanhados da indicação de linha e coluna a que pertencem. Havendo repetições de elementos na matriz, a ordem é irrelevante. Utilize, obrigatoriamente, a função **MAX**.

Por exemplo, no caso da matriz acima, a saída poderia ser:

Elem	Linha	Coluna
8	1	2
7	0	1
5	2	0
4	2	2
3	0	0
3	2	1
2	1	1
1	0	2
1	1	0

(24) Escreva uma função que recebe como parâmetros uma matriz de caracteres  $NOMES_{m \times n}$ , os índices  $i$  e  $j$  de duas linhas e que troca os elementos da linha  $i$  com os da linha  $j$ .

Escreva uma função que recebe como parâmetros uma matriz  $NOMES_{m \times n}$ , os índices  $i$  e  $j$  de duas linhas e que devolve valor 1 se o nome na linha  $i$  é maior ou igual ao da linha  $j$  (ordem alfabética) e 0 caso contrário.

São dados  $n$  nomes que serão armazenados em uma matriz  $NOMES_{m \times n}$ . Coloque os nomes dessa matriz em ordem alfabética usando as funções descritas acima.

(25)(FEA 88) Escreva uma função que recebe como parâmetros uma matriz real  $A_{m \times n}$ , e uma posição  $(i,j)$  da matriz, e calcula a média aritmética dos vizinhos de  $(i,j)$ , ou seja, a média entre  $A_{i-1,j}$ ,  $A_{i+1,j}$ ,  $A_{i,j+1}$ ,  $A_{i,j-1}$ . Desconsidere os vizinhos que não pertencem a matriz (por exemplo, os vizinhos de (0, 0) são somente (0,1) e (1,0) ).

Escreva uma função que recebe como parâmetro uma matriz real  $A_{m \times n}$  e devolve uma matriz  $A^{média}$ , onde  $A_{i,j}^{média}$  é a média aritmética dos vizinhos de  $(i,j)$ . Para isto, utilize a função anterior.

Escreva um programa que lê uma matriz real  $A_{m \times n}$ , e um número inteiro  $k$ ; utilizando a função do item anterior, o programa deve transformar a matriz  $k$  vezes, imprimindo a matriz inicial e depois de cada transformação.

(26) Dizemos que uma matriz  $A_{n \times n}$  é um quadrado latino de ordem  $n$  se em cada linha e em cada coluna aparecem todos os inteiros  $1,2,3,\dots,n$  (ou seja, cada linha e coluna é permutação dos inteiros  $1,2,\dots,n$ ).

Exemplo:  $[3\ 1\ 4\ 2\ 4\ 1\ 1\ 3\ 2\ 2\ 3\ 4\ 4\ 2\ 3\ 1]$

A matriz acima é um quadrado latino de ordem

Escreva uma função que recebe como parâmetros um inteiro  $n$  e um vetor  $V$  com  $n$

inteiros e verifica se em  $V$  ocorrem todos os inteiros de 1 a  $n$ .

Usando a função acima, verifique se uma dada matriz inteira  $A_{n \times n}$  é um quadrado latino de ordem  $n$ .

(27) Verificar se uma matriz quadrada de ordem  $N$  forma um quadrado mágico, cuja soma das linhas, das colunas e das diagonais são iguais.

- (28) Uma imagem em preto e branco, de tamanho  $m \times n$ , pode ser representada por uma matriz cujos elementos assumem valores no conjunto  $\{0,1\}$ . Dado um padrão representado por uma matriz  $3 \times 3$  também assumindo valores em  $\{0,1\}$ , escreva um programa que determine se o padrão existe ou não na imagem. Exemplo:

Matriz	Padrão
0 0 1 0 0 0 0 1 0	0 1 0
0 1 1 1 1 1 1 0 0	1 1 1
0 0 1 0 1 1 1 0 0	0 1 0
0 0 0 0 0 0 0 0 0	
0 0 0 0 1 0 1 0 0	
0 0 0 0 0 1 0 0 0	

- (29) Escreva um programa que receba duas matrizes  $A_{i \times j}$  e  $A_{j \times k}$  e imprima a matriz  $C_{i \times k} = A \times B$ .
- (30) Escreva um programa que leia 21 valores inteiros, sendo que os primeiros 20 valores devem ser armazenados no vetor  $V$ . Em seguida, o programa deve determinar se o 21º valor lido está no vetor, isto é, se pelo menos um dos elementos do vetor é igual ao 21º valor lido. Se estiver, o programa deve imprimir a posição (o índice) em que o valor foi encontrado. Se não estiver, deve imprimir a mensagem: "O valor chave não foi encontrado".
- (31) Modifique a questão acima considerando que os 20 primeiros valores lidos são distintos. Antes de fazer a busca no vetor, o programa deve ordenar os dados no vetor e então fazer uma busca binária. Se o valor da chave estiver no vetor, o programa deve imprimir a posição (o índice) em que o valor foi encontrado. Se não estiver, deve imprimir a mensagem: "O valor chave não foi encontrado".