

IF61C—Fundamentos de Programação 1

Lista de exercícios 10 - Matrizes

Dica: quando necessário, preencha as matrizes com números aleatórios!

1. (*2) Implemente um programa que preencha uma matriz de 6 linhas e 5 colunas de números inteiros e adicione um valor `num` (lido do teclado) a todos os valores divisíveis por 3.
2. (*) Inicialize uma matriz de dimensão 10×10 . Calcule a média dos elementos da diagonal principal e a quantidade ímpares na diagonal secundária. Dica: para selecionar a diagonal principal/secundária, use a informação dos índices da matriz.
3. (*) Implemente um programa que leia uma matriz de inteiros com 7 linhas e 5 colunas. Calcule a soma dos elementos de cada linha e a soma dos elementos de cada coluna, armazenando tais somas nos vetores `SL[7]` e `SC[5]`.
4. (*) Implemente um programa que declare uma matriz quadrada de 10 linhas por 10 colunas e verifique se esta é simétrica ($A(i, j) = A(j, i)$).
5. (**) Faça um programa que: (a) leia os números da primeira linha de uma matriz, A , de dimensão 5×5 ; (b) determine o valor dos demais elementos desta matriz, sabendo que os elementos das demais linhas consistem no resultado da multiplicação de cada elemento da linha anterior pelo valor 2; (c) Exiba a matriz criada. Dica: observe que você vai ter todos os elementos com índice $0c$, em que c varia de 0 a 4. Com isso, $A[1c] = A[0c] * 2$, $A[2c] = A[1c] * 2$, etc... Procure explorar este padrão no laço de repetição!
6. (**) Faça um programa que obtenha do usuário uma matriz quadrada (dimensão máxima de 10×10) e troque o maior elemento de cada linha com o elemento da diagonal. Dica: lembre-se de reinicializar o valor de maior a cada mudança de linha.
7. (**) Faça um programa que leia elementos de uma matriz quadrada de ordem n e armazene o elemento somente se A_{ij} é maior que todos os elementos anteriores. Caso seja menor, imprima um aviso na tela e peça outro número ao usuário.
8. (**) Crie um programa que leia uma quantidade arbitrária de números e imprima, para cada número entre 0 e 100, a quantidade de vezes que o número foi lido. Dica: crie um vetor de 100 posições e, na posição x , armazene o número de vezes que o número x apareceu. Lembre-se de inicializar todas as posições com 0 antes de utilizá-las como contador.
9. (***) Dizemos que uma matriz inteira A ($n \times n$) é uma matriz de permutação se em cada linha e em cada coluna houver $n-1$ elementos nulos e um único elemento igual a 1. Dada uma matriz inteira A ($n \times n$) verificar se A é de permutação. Dica: você deve pensar em uma estratégia para contar o número de valores nulos e não nulos! Assim como no exercício anterior, lembre-se de reinicializar a contagem a cada mudança de linha.
10. (***) Faça um programa para gerar um triângulo de Pascal de N linhas em uma matriz de tamanho $N \times N$.

11. (***) Dizemos que uma matriz quadrada inteira é um quadrado mágico se a soma dos elementos de cada linha, a soma dos elementos de cada coluna e a soma dos elementos das diagonais principal e secundária são todos iguais. Exemplo:

```
8 0 7
4 5 6
3 10 2
```

é um quadrado mágico pois $8 + 0 + 7 = 4 + 5 + 6 = 3 + 10 + 2 = 8 + 4 + 3 = 0 + 5 + 10 = 7 + 6 + 2 = 8 + 5 + 2 = 3 + 5 + 7$ Dada uma matriz quadrada A ($n \times m$), verificar se A é um quadrado mágico. Dica: observe que, assim que uma das somas for diferente, você já pode sair do laço!

Revisão

1. (*) Faça um programa que preenche uma matriz 15×9 com o produto dos índices (de linha e coluna) de cada posição.
2. (*) Dada uma matriz real A ($m \times n$), verificar se existem elementos repetidos em A . Dica: note que o exercício pede apenas se existem elementos repetidos, e não quantos ou quais. Para reduzir o processamento, você pode parar a comparação assim que encontrar o primeiro número repetido! Explore o uso do `break`!
3. (*) Dadas duas matrizes inteiras A e B de ordem 4×3 , fazer um programa que gere uma matriz booleana C , tal que o elemento C_{ij} seja 1 se os elementos nas posições respectivas das matrizes A e B forem iguais, e 0 caso contrário. Exibir as matrizes A , B e C .
4. (**) Implemente um programa que leia uma matriz 10×5 , calcule e imprima a soma dos elementos pares de cada linha e a soma dos elementos ímpares de cada coluna.
5. (**) Dada uma matriz A ($n \times m$) imprimir o número de linhas e o número de colunas **nulas** da matriz (uma linha/coluna nula é aquela em que a soma dos elementos é zero). Dica: você deve criar dois contadores (um para o número de linhas e outro para o número de colunas nulas), os quais devem ser incrementados apenas se uma condição específica for satisfeita.
6. (**) Dada uma matriz de valores do tipo real A com m linhas e n colunas e um vetor real V com n elementos, determinar o produto de A por V . Dica: identifique o padrão dos índices das posições que devem ser multiplicadas.
7. (***) Crie um programa que leia uma quantidade arbitrária de números entre 0 e 100 e imprima cada número encontrado a mesma quantidade de vezes em que foi encontrado. Exemplo: se o número 5 foi encontrado 4 vezes, então imprima o número 5 quatro vezes. Observe o resultado e compare com os números lidos (esse comportamento corresponde a um algoritmo de ordenação e é chamado de count-sort).