

## Matrizes

1. Faça um programa que recebe as dimensões de uma matriz e seus elementos. O programa deve mostrar a soma dos números ímpares da matriz.
2. Construa um programa que recebe os valores de uma matriz com dimensões definidas pelo usuário. O programa deve mostrar o resultado da multiplicação da matriz pelo maior elemento dela.
3. Construa um programa que recebe os elementos de uma matriz de dimensões definidas pelo usuário. O programa deve mostrar o maior e menor elemento de cada linha da matriz.
4. Uma empresa de ônibus oferece transporte entre 5 cidades. Cada cidade tem uma distância para as outras 4 cidades. Faça um programa que recebe o nome de cada uma das 5 cidades e utiliza uma matriz para armazenar as distâncias entre cada par de cidades. O usuário do programa deve digitar uma cidade de origem e seu programa deve mostrar a cidade mais distante e a cidade mais próxima da cidade de origem.
5. Construa uma função `soma_diagonal(matriz)` que tem como parâmetro uma matriz quadrada. A função deve retornar a soma dos elementos da diagonal secundária da matriz.
6. Faça um programa que recebe os elementos uma matriz de dimensões definidas pelo usuário e verifica se é um quadrado mágico. Em um quadrado mágico, a soma de cada linha, a soma de cada coluna, a soma da diagonal principal e a soma da diagonal secundária são iguais.
7. Defina uma função `inverter_linhas(mat)` que tem como parâmetro uma matriz. A função deve retornar a matriz com os elementos de cada linha invertidos. Por exemplo, se a matriz original for `[[1,2,3],[4,5,6]]`, a função deve retornar a matriz `[[3,2,1],[6,5,4]]`.
8. Defina a função `troca_linhas(mat, indice_lin1, indice_lin2)` que tem como parâmetros uma matriz e dois índices de linha, respectivamente. A função deve trocar as linhas `indice_lin1` e `indice_lin2` e retornar a nova matriz. Por exemplo, se a matriz original for `[[1,2],[3,4],[5,6]]`, `indice_lin1` for 0 e `indice_lin2` for 2, então sua função deve retornar a matriz `[[5,6],[3,4],[1,2]]`.
9. Defina a função `transposta(matriz)` que tem como parâmetro uma matriz. A função deve retornar a transposta da matriz.
10. Construa uma função `ordena_intervalos(lista_intervalos)` que recebe uma lista de intervalos. Cada intervalo é uma lista de dois elementos da forma `[inicio, fim]`. A função deve retornar uma lista com os intervalos ordenados de forma crescente pelo tempo de início. Por exemplo, se a lista for `[[1, 3], [5, 7], [2, 4], [6, 8]]`, então sua função deve retornar a lista `[[1, 3], [2, 4], [5, 7], [6, 8]]`.
11. Construa uma função `tem_sobreposicao(lista_intervalos)` que recebe uma lista de intervalos como na questão anterior. A sua função deve retornar `True` caso exista

sobreposição de intervalos, e deve retornar **False**, caso contrário. Por exemplo, se a lista for `[[1, 3], [5, 7], [2, 4], [6, 8]]`, então sua função deve retornar **True** pois os intervalos `[1, 3]` e `[2, 4]` possuem sobreposição.

12. Mostre uma definição para a função `menor_distancia(ponto, lista_pontos)` que recebe as coordenadas  $(x, y)$  de um ponto no espaço cartesiano e uma lista de pontos, respectivamente. O parâmetro `ponto` é uma lista com dois elementos `[x, y]`. Uma lista `lista_pontos` com os pontos  $(x_1, y_1)$ ,  $(x_2, y_2)$  e  $(x_3, y_3)$  deve ser da forma `[[x1, y1], [x2, y2], [x3, y3]]`. É importante observar que `lista_pontos` pode ter qualquer quantidade de pontos. A função deve retornar a distância  $d$  de  $(x, y)$  para o ponto  $(x_i, y_i)$  em `lista_pontos` que está mais próximo de  $(x, y)$ . Em seguida, defina a função `vizinhos_mais_proximos(ponto, lista_pontos)` que deve retornar uma lista com todos os pontos de `lista_pontos` com distância  $d$  de  $(x, y)$ . Além disso, sua função deve retornar uma lista com o mesmo formato da lista `lista_pontos`.
13. Uma indústria fabrica 10 modelos diferentes de televisores. Cada modelo tem uma quantidade de teclas e de alto-falantes. Além disso, a indústria planeja uma quantidade de produção de cada modelo para 12 meses. Por exemplo, um dos modelos pode ter 12 teclas e 2 alto-falantes e a empresa planeja produzir 800 no primeiro mês e 1500 no segundo mês, 700 no terceiro mês, etc. A empresa necessita saber quantos alto-falantes e quantas teclas serão necessárias para a produção em cada um dos 12 meses. Faça um programa que utiliza uma matriz de 2 linhas e 10 colunas para armazenar os valores de quantas teclas e quantos alto-falantes são necessários para cada modelo. Além disso, o programa deve utilizar uma segunda matriz para guardar a produção de cada modelo em cada um dos 12 meses. No final, o programa deve mostrar a quantidade de teclas e alto-falantes que serão necessários em cada mês. **Dica:** utilize multiplicação de matrizes.
14. Defina uma função `permutacoes(lista)` que retorna uma lista com todas as permutações do parâmetro `lista`. Por exemplo, `permutacoes([1, 2, 3])` deve retornar `[[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 1, 2], [3, 2, 1]]`.