

```
# coding: utf-8

''' 1. Faça uma função que dada uma matriz de inteiros de tamanho qualquer e um
número inteiro, conta e
retorna quantas vezes aquele número aparece na matriz. '''

def contar(M, n):
    c=0

    for linha in M:
        c+=list.count(linha, n)

    return c

''' 2. Faça uma função que, dada uma matriz qualquer, gera e retorna a matriz
transposta.'''

def transposta(M):
    T = []

    for j in range(len(M[0])):
        nova_linha=[]

        for i in range(len(M)):
            nova_linha.append(M[i][j])

        T.append(nova_linha)

    return T

''' 3. Faça uma função que dada uma matriz de inteiros, retorna a média de todos
os números da matriz.'''

def media(M):
    L=[]

    for i in range(len(M)):
        for j in range(len(M[0])):
            L.append(M[i][j])

    return sum(L)*1.0/len(L)

''' 4. Uma pista de Kart permite 10 voltas para cada um dos 6 corredores. Faça uma
função que receba como
entrada uma matriz 6 x 10 com os tempos em segundos dos corredores em cada volta.
A função deve
retornar uma tupla informando: De quem foi a melhor volta da prova, com qual tempo
e em que volta.
Assuma que os corredores tem tempos diferentes. (use a função min) '''

def melhor_volta(M):
    # uma linha corresponde a um corredor, com cada elemento sendo um tempo

    T = [] # lista com os menores tempos de cada corredor
    V = [] # lista com o numero da volta de melhor tempo de cada corredor

    for corredor in M:
        T.append(min(corredor))
        V.append(list.index(M, corredor))

    melhor_tempo= min(T)
    de_quem = list.index(T, melhor_tempo)
    qual_volta= list.index(M[de_quem], melhor_tempo)
```

```
    return ("corredor "+str(de_quem + 1), "melhor tempo = "+ str(melhor_tempo) + '
segundo(s)', "volta: "+str(qual_volta + 1))

def testeQ4():
    c=[[2, 3, 4, 5, 6, 7, 8, 9, 10, 11],
        [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], # linha 1, coluna 0, valor = 1
        [2, 3, 4, 5, 6, 7, 8, 9, 10, 11],
        [2, 3, 4, 5, 6, 7, 8, 9, 10, 11],
        [2, 3, 4, 5, 6, 7, 8, 9, 10, 11],
        [2, 3, 4, 5, 6, 7, 8, 9, 10, 11]]

    return melhor_volta(c)
```