

## Atividades ADNP Semana 07 :: Matrizes

### Instruções Gerais

- Faça cada exercício em uma função distinta.
  - Utilize a extensão .c e o compilador de gcc.
  - Utilize o editor de sua preferência: Code Blocks, VS Code, Dev C++, etc.
1. Escreva uma função que imprime o conteúdo de uma matriz ao contrário, isto é, do último elemento ao primeiro. Considerando a matriz do exemplo anterior, a função imprimiria do 9 ao 1.

**void printMatRev(int rows, int cols, int m[rows][cols])**

Exemplo:

```
int mat[4][3] = {
    { 1, 2, 3},
    { 4, 5, 6},
    { 7, 8, 9},
    {10,11,12}
};
printMatRev(4, 3, mat); // deve imprimir do 12 ao 1
```

2. Escreva uma função que encontra e imprime o maior e o menor valores contidos em uma matriz.

**void findMinMax(int rows, int cols, int m[rows][cols])**

3. Escreva uma função recebe uma matriz e coloca as somas de cada linha na última coluna.

**void findTotals(int rows, int cols, int m[rows][cols])**

Exemplo:

```
int mat[4][5] = {
    { 1, 2, 3, 4, 0},
    { 2, 4, 6, 8, 0},
    { 3, 6, 9, 12, 0},
    { 4, 8, 12,16, 0} // a última coluna recebe a soma da linha
};
findTotals(4, 5, mat);
```

```
// Após a chamada da função, a matriz deve estar com o seguinte conteúdo:
// ** O último elemento de cada linha recebe a soma dos demais elementos **
// { 1, 2, 3, 4, 10}      1 + 2 + 3 + 4 = 10
// { 2, 4, 6, 8, 20}      2 + 4 + 6 + 8 = 20
// { 3, 6, 9, 12,30}      ...
// { 4, 8, 12,16,40}
```

4. Escreva uma função que realiza a adição de duas matrizes m1 e m2, colocando resultado em m3. Considere que todas as matrizes possuem as mesmas dimensões r x c.

```
void sum(int r, int c, int m1[r][c], int m2[r][c], int m3[r][c]).
```

Exemplo para matrizes 3x2:

$$\begin{array}{cc} m1 & m2 \\ \begin{bmatrix} 1 & 3 \\ 1 & 0 \\ 1 & 2 \end{bmatrix} & + \begin{bmatrix} 0 & 0 \\ 7 & 5 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 1+0 & 3+0 \\ 1+7 & 0+5 \\ 1+2 & 2+1 \end{bmatrix} = \begin{array}{cc} m3 \\ \begin{bmatrix} 1 & 3 \\ 8 & 5 \\ 3 & 3 \end{bmatrix} \end{array}$$

5. Escreva uma função que monta a transposta da matriz m1 em m2. Observe que deve haver uma compatibilidade entre as dimensões das matrizes.

```
void transpose(int r1, int c1, int m1[r1][c1], int r2, int c2, int m2[r2][c2]).
```

Exemplo para matriz 2x3:

$$\begin{array}{cc} m1 & m2 \\ \begin{bmatrix} 1 & 2 & 3 \\ 0 & -6 & 7 \end{bmatrix}^T & = \begin{bmatrix} 1 & 0 \\ 2 & -6 \\ 3 & 7 \end{bmatrix} \end{array}$$

6. Escreva uma função que verifica se uma matriz é simétrica e devolve 1 (true) ou 0 (false). Em uma matriz simétrica os valores são espelhados em relação à diagonal principal. Considere que a matriz é quadrada, com dimensões d.

```
int checkSymmetric(int d, int m[d][d]).
```

Exemplo para matriz 3x3

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 8 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 8 \end{bmatrix}$$