



# Algoritmos e Programação II

Prof.<sup>a</sup> Noeli A. Pimentel Vaz  
Prof. Joilson dos Reis Brito



# Correção Lista 06

## MATRIZ - Exercícios - Lista 06

---

3 - Faça um programa que leia uma matriz 3x3 de números inteiros e escreva os elementos que tenham o número da linha igual ao número da coluna.

# MATRIZ - Exercícios - Lista 06

```
#define M 3
#define N 3
int main()
{
    int MatrizX[M][N],I,J;
    printf("Informe os elementos da Matriz X.\n");
    for(I=0;I<M;I++)
        for(J=0;J<N;J++)
        {
            printf("Elemento[%i][%i]:", I, J);
            scanf("%i", &MatrizX[I][J]);
        }
}
```

## MATRIZ - Exercícios - Lista 06

```
printf("\nConteúdo da Diagonal Principal da Matriz X:\n");
for(I=0;I<M;I++)
    for(J=0;J<N;J++)
        if(I==J)
            printf("%d ",MatrizX[I][J]);
printf("\n");
return 0;
}
```

## MATRIZ - Exercícios - Lista 06

---

4 - Faça um programa que leia uma matriz 3x5 de números inteiros, depois leia uma variável inteira que representa uma coluna da matriz e escreva todas as linhas desta coluna.

# MATRIZ - Exercícios - Lista 06

```
#define M 3
#define N 5
int main()
{
    int MatrizX[M][N],I,J,ColunaEscolhida;
    printf("Informe os elementos da Matriz X.\n");
    for(I=0;I<M;I++)
        for(J=0;J<N;J++)
        {
            printf("Elemento[%i][%i]:", I, J);
            scanf("%i", &MatrizX[I][J]);
        }
}
```

# MATRIZ - Exercícios - Lista 06

4 - Faça um programa que leia uma matriz 3x5 de números inteiros, depois leia uma variável inteira que representa uma coluna da matriz e escreva todas as linhas desta coluna.

```
printf("Qual coluna você quer que seja mostrada?");  
scanf("%d",&ColunaEscolhida);  
printf("\nConteúdo da coluna escolhida:\n");  
for(l=0;l<M;l++)  
    printf("%d ",MatrizX[l][ColunaEscolhida]);  
printf("\n");  
  
return 0;  
}
```



## MATRIZ - Exercícios - Lista 06

---

5 - Faça um programa que leia duas matrizes de 3X3, depois some seus valores em outra matriz 3X3 e por fim escreva a matriz resultante.

# MATRIZ - Exercícios - Lista 06

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define LINHAS
#define COLUNAS 3
int main()
{
    srand(time(NULL));
    int MatrizSoma[LINHAS][COLUNAS], I,J;
    int MatrizA[LINHAS][COLUNAS], MatrizB[LINHAS][COLUNAS];
    for(I=0;I<LINHAS;I++)
        for(J=0;J<COLUNAS;J++)
        {
            // preenche a matriz com números aleatórios
            MatrizA[I][J]=rand()%100;
            MatrizB[I][J]=rand()%100;
```

# MATRIZ - Exercícios - Lista 06

```
for(l=0;l<LINHAS;l++)
    for(J=0;J<COLUNAS;J++)
        MatrizSoma[l][J]=MatrizA[l][J]+MatrizB[l][J];

printf("\nConteúdo da Matriz Soma:\n");
for(l=0;l<M;l++)
{
    for(J=0;J<COLUNAS;J++)
        printf("%d ",MatrizSoma[l][J]);
    printf("\n");
}
printf("\n");
return 0;
}
```

## MATRIZ - Exercícios - Lista 06

---

7 - Faça um programa para gerar uma matriz 5X6 de números inteiros aleatórios.

Depois, o programa deve calcular e escrever:

- . A soma de cada uma das 5 linhas
- . A soma de cada uma das 6 colunas

# MATRIZ - Exercícios - Lista 06

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define LINHAS 5
#define COLUNAS 6
int main()
{
    srand(time(NULL));
    int Numeros[LINHAS][COLUNAS], I,J,SomaLinha, SomaColuna;
    for(I=0;I<LINHAS;I++)
        for(J=0;J<COLUNAS;J++)
        {
            // preenche a matriz com números aleatórios
            Numeros[I][J]=rand()%100;
        }
}
```

# MATRIZ - Exercícios - Lista 06

```
for(l=0;l<LINHAS;l++)
{
    SomaLinha = 0;
    for(J=0;J<COLUNAS;J++)
        SomaLinha += Numeros[l][J];
    printf("Soma da linha %d: %d\n",l,SomaLinha);
}
for(J=0;J<COLUNAS;J++)
{
    SomaColuna = 0;
    for(l=0;l<LINHAS;l++)
        SomaColuna += Numeros[l][J];
    printf("Soma da coluna %d: %d\n",J,SomaColuna);
}
return 0;
}
```

## MATRIZ - Exercícios - Lista 06

---

9 - Desenvolva um programa que leia uma matriz 3 x 3 com números inteiros e mostre uma mensagem dizendo se a matriz digitada é simétrica ou não. Uma matriz só pode ser simétrica se  $A[i,j] = A[j,i]$ .

## MATRIZ - Exercícios - Lista 06

```
#include<stdio.h>  
#define LINHAS 3  
#define COLUNAS 3  
int main()  
{  
    int Matriz[LINHAS][COLUNAS], I, J, Flag=0;  
    for(I=0;I<LINHAS;I++)  
        for(J=0;J<COLUNAS;J++)  
            scanf("%d",&Matriz[I][J]);
```



# MATRIZ - Exercícios - Lista 06

```
for(I=0;I<LINHAS;I++)
{
    for(J=0;J<COLUNAS;J++)
        if(Matriz[I][J] != Matriz[J][I])
            Flag = 1;
}
if(Flag == 1)
    printf("A matriz não é simétrica");
else
    printf("A matriz é simétrica");

return 0;
}
```

## MATRIZ - Exercícios - Lista 06

---

10 - Desenvolva um programa que gere um conjunto de números inteiros para preencher uma matriz 10X10 e a partir daí, gere um vetor de 10 elementos com os maiores elementos de cada linha e outro vetor de 10 elementos com os menores elementos de cada coluna.

# MATRIZ - Exercícios - Lista 06

```
#include<stdio.h>
#include<stdlib.h>
#include<locale.h>
#include<time.h>
#define M 10
#define N 10
int main()
{
    setlocale(LC_ALL, "Portuguese");
    int MatrizX[M][N],I,J,Aux,MaioresLinhas[M],MenoresColuna[M];
    srand(time(NULL));
    for(I=0;I<M;I++)
        for(J=0;J<N;J++)
            MatrizX[I][J]=rand()%100;
```

# MATRIZ - Exercícios - Lista 06

**\\ Preencher vetor MaioresLinhas com o primeiro elemento de cada linha**

**for(I=0;I<M;I++)**

**MaioresLinhas[I]=MatrizX[I][0];**

**\\ Preencher vetor MenoresColuna com o primeiro elemento de cada linha**

**for(J=0;J<N;J++)**

**MenoresColuna[J]=MatrizX[0][J];**

**\\ Encontrar maiores valores de cada linha e menores valores de cada coluna**

**for(I=0;I<M;I++)**

**for(J=0;J<N;J++)**

**{**

**if(MatrizX[I][J]>MaioresLinhas[I])**

**MaioresLinhas[I] = MatrizX[I][J];**

**if(MatrizX[I][J]<MenoresColuna[J])**

**MenoresColuna[J] = MatrizX[I][J];**

**}**

# MATRIZ - Exercícios - Lista 06

```
printf("\nConteúdo da Matriz gerada:\n");
for(l=0;l<M;l++)
{
    for(J=0;J<N;J++)
        printf("%2i ",MatrizX[l][J]);
    printf("\n");
}
printf("\nMaiores valores de cada linha:\n");
for(l=0;l<M;l++)
    printf("%2i ",MaioresLinhas[l]);
printf("\nMenores valores de cada coluna:\n");
for(J=0;J<N;J++)
    printf("%2i ",MenoresColuna[J]);
printf("\n");
return 0;
}
```



# Correção APA1

# APA1 - Questão 1

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>
#define PRODUTOS 10
int main()
{
    float Precos[PRODUTOS],Maiscaro, MaisBarato,PrecoMedio,TotalPrecos;
    int l;
    setlocale(LC_ALL,"Portuguese");
    // Ler Preços
    printf("Digite os preços dos %d produtos\n",PRODUTOS);
    for(l=0;l<PRODUTOS;l++)
        scanf("%f",&Precos[l]);
```

# APA1 - Questão 1

**// Calcular valores**

**MaisBarato = Precos[0];Maiscaro=Precos[0];TotalPrecos=0;**

**for(l=0;l<PRODUTOS;l++)**

**{**

**if (Precos[l] > Maiscaro)**

**Maiscaro = Precos[l];**

**if (Precos[l] < MaisBarato)**

**MaisBarato = Precos[l];**

**TotalPrecos += Precos[l];**

**}**

**PrecoMedio = TotalPrecos / PRODUTOS;**



# APA1 - Questão 1

---

```
printf("Mais caro: %.2f\n",Maiscaro);
printf("Mais barato: %.2f\n",MaisBarato);
printf("Preço médio: %.2f\n",PrecoMedio);
printf("Acima do médio:\n");
for(l=0;l<PRODUTOS;l++)
    if (Precos[l] > PrecoMedio)
        printf("%.2f\n",Precos[l]);
return 0;
}
```

# APA1 - Questão 2

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>
#define TAMANHO 10
int main()
{ char Gabarito[TAMANHO],Respostas1[TAMANHO],Respostas2[TAMANHO];
  int I,J;
  float NotaAluno1,NotaAluno2;
  setlocale(LC_ALL,"Portuguese");
  // Ler Gabarito
  printf("Gabarito\n");
  for(I=0;I<TAMANHO;I++)
  {
    fflush(stdin);
    scanf("%c",&Gabarito[I]);
  }
```

# APA1 - Questão 2

**// Ler respostas aluno 1**

```
printf("Respostas aluno 1:\n");  
for(l=0;l<TAMANHO;l++)  
{  
    fflush(stdin);  
    scanf("%c",&Respostas1[l]);  
}
```

**// Ler respostas aluno 2**

```
printf("Respostas aluno 2:\n");  
for(l=0;l<TAMANHO;l++)  
{  
    fflush(stdin);  
    scanf("%c",&Respostas2[l]);  
}
```

# APA1 - Questão 2

## \\ Calcular notas dos alunos

```
NotaAluno1 = 0;NotaAluno2=0;
for(l=0;l<TAMANHO;l++)
{
    if (Respostas1[l] == Gabarito[l])
        NotaAluno1 ++;
    if (Respostas2[l] == Gabarito[l])
        NotaAluno2 ++;
}
printf("Nota Aluno 1: %.1f\n",NotaAluno1);
printf("Nota Aluno 2: %.1f\n",NotaAluno2);
returno 0;
}
```



# APA2 - Erros Frequentes

Não utilizar for para ler vetor

Não usou constante para definir tamanho de vetor

Utilizou um vetor para armazenar os preços acima da média  
inicializou o mais barato com 0

# MULTIPLICAÇÃO DE MATRIZ

Vamos multiplicar as matrizes  $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$  e  $B = \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix}$  para entender como se obtém cada elemento  $c_{ij}$ :

- 1ª linha e 1ª coluna

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} \boxed{1 \cdot (-1) + 2 \cdot 4} & \quad \quad \quad \\ \quad \quad \quad & \quad \quad \quad \end{bmatrix} \quad c_{11}$$

- 1ª linha e 2ª coluna

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} 1 \cdot (-1) + 2 \cdot 4 & \boxed{1 \cdot 3 + 2 \cdot 2} \\ \quad \quad \quad & \quad \quad \quad \end{bmatrix} \quad c_{12}$$

# MULTIPLICAÇÃO DE MATRIZ

- 2ª linha e 1ª coluna

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} 1 \cdot (-1) + 2 \cdot 4 & 1 \cdot 3 + 2 \cdot 2 \\ \boxed{3 \cdot (-1) + 4 \cdot 4} & \end{bmatrix}$$

$c_{21}$

- 2ª linha e 2ª coluna

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} 1 \cdot (-1) + 2 \cdot 4 & 1 \cdot 3 + 2 \cdot 2 \\ 3 \cdot (-1) + 4 \cdot 4 & \boxed{3 \cdot 3 + 4 \cdot 2} \end{bmatrix}$$

$c_{22}$

Assim,  $A \cdot B = \begin{bmatrix} 7 & 7 \\ 13 & 17 \end{bmatrix}$ .

# MULTIPLICAÇÃO DE MATRIZ

---

Faça um programa que leia duas matrizes (A,B) de 3X3 , depois calcule a matriz R que deve conter a multiplicação das matrizes A e B ( $R = A * B$ ), por fim escreva a matriz resultante.



# MULTIPLICAÇÃO DE MATRIZ

1- Faça um programa que leia duas matrizes (A,B) de 3X3 , depois calcule a matriz R que deve conter a multiplicação das matrizes A e B ( $R = A * B$ ), por fim escreva a matriz resultante.

A multiplicação de matrizes é feita da seguinte forma:

$$R[0][0] = A[0][0] * B[0][0] + A[0][1] * B[1][0] + A[0][2] * B[2][0]$$

$$R[0][1] = A[0][0] * B[0][1] + A[0][1] * B[1][1] + A[0][2] * B[2][1]$$

$$R[0][2] = A[0][0] * B[0][2] + A[0][1] * B[1][2] + A[0][2] * B[2][2]$$

$$R[1][0] = A[1][0] * B[0][0] + A[1][1] * B[1][0] + A[1][2] * B[2][0]$$

$$R[1][1] = A[1][0] * B[0][1] + A[1][1] * B[1][1] + A[1][2] * B[2][1]$$

$$R[1][2] = A[1][0] * B[0][2] + A[1][1] * B[1][2] + A[1][2] * B[2][2]$$

$$R[2][0] = A[2][0] * B[0][0] + A[2][1] * B[1][0] + A[2][2] * B[2][0]$$

$$R[2][1] = A[2][0] * B[0][1] + A[2][1] * B[1][1] + A[2][2] * B[2][1]$$

$$R[2][2] = A[2][0] * B[0][2] + A[2][1] * B[1][2] + A[2][2] * B[2][2]$$

# MULTIPLICAÇÃO DE MATRIZ

$$\begin{aligned} R[0][0] = & A[0][0] * B[0][0] + \\ & A[0][1] * B[1][0] + \\ & A[0][2] * B[2][0] \end{aligned}$$

$$\begin{aligned} R[0][1] = & A[0][0] * B[0][1] + \\ & A[0][1] * B[1][1] + \\ & A[0][2] * B[2][1] \end{aligned}$$

$$\begin{aligned} R[0][2] = & A[0][0] * B[0][2] + \\ & A[0][1] * B[1][2] + \\ & A[0][2] * B[2][2] \end{aligned}$$

# MULTIPLICAÇÃO DE MATRIZ

$$\begin{aligned} R[1][0] = & A[1][0] * B[0][0] + \\ & A[1][1] * B[1][0] + \\ & A[1][2] * B[2][0] \end{aligned}$$

$$\begin{aligned} R[1][1] = & A[1][0] * B[0][1] + \\ & A[1][1] * B[1][1] + \\ & A[1][2] * B[2][1] \end{aligned}$$

$$\begin{aligned} R[1][2] = & A[1][0] * B[0][2] + \\ & A[1][1] * B[1][2] + \\ & A[1][2] * B[2][2] \end{aligned}$$

# MULTIPLICAÇÃO DE MATRIZ

$$\begin{aligned} R[2][0] = & A[2][0] * B[0][0] + \\ & A[2][1] * B[1][0] + \\ & A[2][2] * B[2][0] \end{aligned}$$

$$\begin{aligned} R[2][1] = & A[2][0] * B[0][1] + \\ & A[2][1] * B[1][1] + \\ & A[2][2] * B[2][1] \end{aligned}$$

$$\begin{aligned} R[2][2] = & A[2][0] * B[0][2] + \\ & A[2][1] * B[1][2] + \\ & A[2][2] * B[2][2] \end{aligned}$$

# MATRIZ - Exercícios

...

```
for(I=0;I<M;I++)
```

```
    for(J=0;J<N;J++)
```

```
    {
```

```
        MatrizR[I][J] = 0;
```

```
        for(K=0;K<M;K++)
```

```
            MatrizR [I][J]=MatrizR[I][J]+MatrizA[I][K]*MatrizB[K][J];
```

```
    }
```

...