

Computação – Informática

Linguagem C

“Matrizes em C”

Aula N. 05

Prof. SIMÃO

Estrutura de Dados Homogênea “matrizes”

Uma matriz computacional é um “variável composta” capaz de armazenar uma quantidade determinada de valores em (duas) dimensões.

Uma matriz computacional é inspirada na matriz matemática, que também é capaz de armazenar um conjunto de valores em duas dimensões.

1	3	7
0	1	0
1	9	9

Declaração de Matrizes em algoritmos – compatível com a linguagem C.

<tipo-básico> NOME_MATRIZ [Número1] [Número2];

Onde:

Número1 e *Número2* representam as dimensões da matriz

< tipo-básico > - representa qualquer um dos tipos básicos

Exemplo 1.1

```
#include <stdio.h>
void main ( )
{
    float matriz [2] [2];

    float media, quant;

    matriz [0][0] = 9.0;
    matriz [0][1] = 8.0;
    matriz [1][0] = 7.0;
    matriz [1][1] = 7.0;

    quant = matriz[0][0] + matriz[0][1]
            + matriz[1][0] + matriz[1][1];

    media = quant / 4.0;
    printf ( " A média dos 4 números é: %f.", media );
}
```



Índices	0	1
0	9.0	8.0
1	7.0	7.0

Exemplo 1.2

```
#include <stdio.h>
void main ( )
{
    float matriz [2][2];

    float media, quant;

    printf ( “ Digite o 1o número: ” );
    scanf ( “%f”, &matriz[0][0] );

    printf ( “ Digite o 2o número: ” );
    scanf ( “%f”, &matriz[0][1] );

    printf ( “ Digite o 3o número: ” );
    scanf ( “%f”, &matriz[1][0] );

    printf ( “ Digite o 4o número: ” );
    scanf ( “%f”, &matriz[1][1] );

    quant = matriz [0][0] + matriz [0][1] + matriz [1][0] + matriz [1][1];

    media = quant / 4.0;
    printf ( “ A média dos 4 números é: %f.”, media );
}
```

Exemplo 1.3

```
#include <stdio.h>

void main ( )
{
    float matriz [4][2];

    float  media, quant;
    int    cont1, cont2;
    quant = 0;

    for ( cont1 = 0; cont1 <= 3; cont1 = cont1 + 1 )
    {
        for ( cont2 = 0; cont2 <= 1; cont2 = cont2 + 1 )
        {
            printf ( "Digite o número de posição %i, %i da matriz::", cont1, cont2 );
            scanf ( "%f", &matriz [ cont1 ] [ cont2 ] );

            quant = quant + matriz[ cont1 ] [ cont2 ] ;
        }
    }

    media = quant / 8.0;
    printf ( " A média dos 8 números da matriz é: %f.", media);
}
```

Declaração de Vetores e Matrizes

Exemplos:

```
int main()
{
    float parc_aluno[2];

    float notas_alunos[5][2];

    char nome[40];

    char nomes_turma[5][40];

    return 0;
}
```

Exercício

- Elaborar um programa que leia as duas parciais de cada aluno de uma turma de 30 alunos, classificando cada um como aprovado, em exame ou reprovado.
 - O aluno que obtiver média maior ou igual a 7 será aprovado.
 - O aluno que obtiver 10 na segunda parcial também será aprovado.
 - O aluno que obtiver média maior ou igual a 5 irá para exame.
 - O aluno que obtiver média maior ou igual a P também irá para exame, sendo P calculado como $1,5 \times \text{média_da_turma}$.

Solução

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int cont;
    float parc_aluno[2];

    float med_aluno = 0.0;
    float med_turma = 0.0;
    float soma      = 0.0;
    float parametro = 0.0;

    float notas_alunos[5][2];

    for (cont = 0; cont < 5; cont++)
    {
        printf ("Primeira nota: \n");
        scanf ("%f", &parc_aluno[0]);
        printf ("Segunda nota: \n ");
        scanf ("%f", &parc_aluno[1]);

        notas_alunos[cont][0] = parc_aluno[0];
        notas_alunos[cont][1] = parc_aluno[1];
        med_aluno = ( parc_aluno[0] + parc_aluno[1] )/2.0;
        soma = soma + med_aluno;
        printf("\n");
    }

    med_turma = soma / 5.0;
    parametro = (1.5)*med_turma;
    printf ("A média da turma é: %f \n", med_turma);
    printf ("O parametro é : %f \n \n", parametro);
```

```
for (cont = 0; cont < 5; cont++)
{
    med_aluno = ( notas_alunos [cont][0] +
                 notas_alunos [cont][1] )/2.0;

    if ( (med_aluno >= 7) || (10 == notas_alunos[cont][1]) )
    {
        printf ("Aluno %d aprovado! \n", cont);
    }
    else
    {
        if ( (med_aluno >= 5) || (med_aluno >= parametro) )
        {
            printf ("Aluno %d em exame! \n", cont );
        }
        else
        {
            printf ("Aluno %d reprovado! \n", cont);
        }
    }
}

system ("pause");
return 0;
}
```

Solução

```
#include <stdio.h>
#include <stdlib.h>
int main ( )
{
    const int num_alunos = 5;
    int cont;
    float parc_aluno [ 2 ];

    float med_aluno = 0.0;
    float med_turma = 0.0;
    float soma      = 0.0;
    float parametro = 0.0;

    float notas_alunos [num_alunos][2];

    for (cont = 0; cont < num_alunos; cont++)
    {
        printf ("Primeira nota: \n");
        scanf ("%f", &parc_aluno[0]);
        printf ("Segunda nota: \n ");
        scanf ("%f", &parc_aluno[1]);

        notas_alunos[cont][0] = parc_aluno[0];
        notas_alunos[cont][1] = parc_aluno[1];
        med_aluno = ( parc_aluno[0] + parc_aluno[1] )/2.0;
        soma = soma + med_aluno;
        printf("\n");
    }

    med_turma = soma / num_alunos;
    parametro = (1.5)* med_turma;
    printf ("A média da turma é: %f \n", med_turma);
    printf ("O parametro é : %f \n \n", parametro);
```

```
for (cont = 0; cont < num_alunos ; cont++)
{
    med_aluno = ( notas_alunos [cont][0] +
                 notas_alunos [cont][1] )/2.0;

    if ( (med_aluno >= 7) || (10 == notas_alunos[cont][1]) )
    {

        printf ("Aluno %d aprovado! \n", cont);

    }
    else
    {
        if ( (med_aluno >= 5) || (med_aluno >= parametro) )
        {

            printf ("Aluno %d em exame! \n", cont );

        }
        else
        {

            printf ("Aluno %d reprovado! \n", cont);

        }
    }
}

system ("pause");
return 0;
}
```

Exercício 2

- Elaborar um programa que leia **o nome** e as duas parciais de cada aluno de uma turma de 30 alunos, classificando cada um como aprovado, em exame ou reprovado.
 - O aluno que obtiver média maior ou igual a 7 será aprovado.
 - O aluno que obtiver 10 na segunda parcial também será aprovado.
 - O aluno que obtiver média maior ou igual a 5 irá para exame.
 - O aluno obtiver média maior ou igual a P também irá para exame, sendo P calculado como $1,5 \times \text{média_da_turma}$.

Solução

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main()
{
    const int num_alunos = 5;    int cont;
    float parc_aluno [ 2 ];

    float med_aluno = 0;  float med_turma = 0;
    float soma      = 0;  float parametro = 0;

    float notas_alunos[num_alunos][2];
    char nome[140];
    char nomes_turma[num_alunos][140];

    for ( cont = 0; cont < num_alunos; cont++ )
    {
        printf ("Nome do aluno: \n");
        fflush (stdin);
        gets (nome);           // scanf("%s", nome);
        printf ("Primeira nota: \n");
        scanf ("%f", &parc_aluno[0]);
        printf ("Segunda nota: \n ");
        scanf ("%f", &parc_aluno[1]);

        // nomes_turma[cont] = nome;
        strcpy ( nomes_turma[cont], nome );

        notas_alunos[cont][0] = parc_aluno[0];
        notas_alunos[cont][1] = parc_aluno[1];
        med_aluno = (parc_aluno[0] + parc_aluno[1])/2;
        soma = soma + med_aluno;
        printf("\n");
    }
}
```

```
med_turma = soma / num_alunos;
parametro = (1.5)*med_turma;

printf ("A média da turma é: %f \n", med_turma);
printf ("O parametro é : %f \n \n", parametro);

for (cont = 0; cont < num_alunos; cont++)
{
    med_aluno = ( notas_alunos[cont][0] +
                 notas_alunos[cont][1] ) / 2;

    if ( (med_aluno >= 7) || (10 == notas_alunos[cont][1] ) )
    {
        printf ("Aluno %s aprovado! \n", nomes_turma[cont] );
    }
    else
    {
        if ( (med_aluno >= 5) || (med_aluno > parametro) )
        {
            printf ("Aluno %s em exame! \n", nomes_turma[cont] );
        }
        else
        {
            printf ("Aluno %s reprovado! \n", nomes_turma[cont] );
        }
    }
}

system ("pause");

return 0;
}
```

O comando (função) strcmp.

```
#include <stdio.h>
#include <string.h>

int main()
{
    int quant = 3;
    char umnome[50];
    char nomes[quant][50];

    for ( int i = 0; i < quant; i++ )
    {
        puts (" Informe um nome: ");
        fflush (stdin);
        gets (nomes[i]);
    }

    puts (" Digite um nome a ser pesquisado: ");
    fflush (stdin);
    gets (umnome);

    int ok = 0;
    for (int i = 0; i < quant; i++)
    {
        int aux;
        aux = strcmp (nomes[i], umnome);
        if (aux == 0)
        {
            ok = 1;
            puts("Achei!");
            break;
        }
    }
}
```

```
if (ok == 0)
{
    puts ("Ops, não achei");
}

fflush (stdin);
getchar();

return 0;
}
```

Estudem outras funções,
como *strcpy* e *strlen*.

Fazer exercícios:

os da apostila, por exemplo.

Exercício

- a) **Transforme os algoritmos dos slides seguintes em programas em linguagem C.**

Exemplo 1.1

Algoritmo em estilo C 'Exemplo Matriz...'.
Início

Real matriz[2][2];

Real media, quant;

matriz[0][0] ← 9.0;

matriz[0][1] ← 8.0;

matriz[1][0] ← 7.0;

matriz[1][1] ← 7.0;

quant ← matriz[0][0] + matriz[0][1]
+ matriz[1][0] + matriz[1][1];

media ← quant / 4.0;

Imprima (" A média dos 4 números é: %f.", media);

Fim.



Índices	0	1
0	9.0	8.0
1	7.0	7.0

Exemplo 1.2

Algoritmo 'Exemplo Matriz...'.
Início

Real matriz [2] [2];

Real media, quant;

Imprima ("Digite o 1o número:");

Leia (matriz [0][0]);

Imprima ("Digite o 2o número:");

Leia (matriz [0][1]);

Imprima ("Digite o 3o número:");

Leia (matriz [1][0]);

Imprima ("Digite o 4o número:");

Leia (matriz [1][1]);

quant ← matriz [0][0] + matriz[0][1] + matriz[1][0] + matriz[1][1];

media ← quant / 4.0;

Imprima (" A média dos 4 números é: %f.", media);

Fim.

Exemplo 1.3

Algoritmo 'Exemplo Matriz 2...'

Início

Real matriz[4][2];

Real media, quant;

Inteiro cont1, cont2;

quant ← 0;

para cont1 de 0 até 3 **passo** 1 **faça**

para cont2 de 0 até 1 **passo** 1 **faça**

Imprima ("Digite o número de posição %i, %i da matriz::", cont1, cont2);

Leia (matriz [cont1][cont2]);

 quant ← quant + matriz[cont1][cont2] ;

fim-para

fim-para

média ← quant / 8.0;

Imprima (" A média dos 8 números da matriz é: %f.", media);

Fim.

Exercício

- a) Elabore um programa que registra os nomes de 50 alunos (*i.e.* matriz de caracter de 50 por 150) e as quatro notas de cada um dos alunos (*i.e.* matriz de reais de 50 por 4).
- b) O programa deve ainda fornecer os nomes dos alunos que tem **nota média** maior que a média da turma.

Algoritmo "Nomes e notas"

// Principal

Início

Caracter nomes[50][150];

Real notas[50][4];

Real medias[50];

Real soma, somag, mediat;

Inteiro cont, cont2;

somag ← 0;

para cont de 0 até 49 passo 1 faça

 imprima ("O nome do aluno:");

 leia (nomes[cont]);

 imprima ("As notas do aluno:");

 soma ← 0;

 para cont2 de 0 até 3 passo 1 faça

 repita

 ...

 leia(notas[cont][cont2]);

 ...

 até ((notas[cont][cont2] >= 0) e (notas[cont][cont2] <=10))

 soma ← soma + notas[cont][cont2]

 fim-para

 medias[cont] ← soma / 4;

 somag ← somag + medias[cont];

fim-para

mediat ← somag / 50

para cont de 0 até 49 passo 1 faça

 se (medias[cont] > mediat) então

 Imprima ("O aluno", nomes[cont], "tem média maior que a média da turma que é ", mediat);

 fim-se

fim-para

Fim

Exercícios C

- a) Uma matriz **M** pode ser transposta em uma matriz **N** onde o número de linhas/colunas da primeira passa a ser o número de colunas/linhas da segunda. Faça um programa que:
- Leia uma matriz **M** (o usuário deve fornecer a quantidade de linhas e a quantidade de colunas que não deverão exceder 100 linhas e 80 colunas).
 - Calcule a transposta **N** da matriz **M**.
 - Imprima a transposta calculada.
- b) Elabore um programa que multiplique duas Matrizes **M** e **N** (10x10) fornecidas pelo usuário. O programa deve:
- Ler as matrizes **M** e **N**.
 - Calcular a multiplicação e armazenar na matriz **Mult**.
 - Imprimir o resultado.
- c) Elabore um programa que manipule os dados para uma sala de 40 alunos. O programa deve receber como entrada (via leitura)
- Os nomes dos alunos.
 - As três parciais.
 - A frequência de cada aluno.
- O programa deve calcular e imprimir
- O nome.
 - A média final.
 - A frequência.
 - A maior parcial e a menor parcial.