

Matrizes

Professores(as):

Virgínia Fernandes Mota

João Eduardo Montandon de Araujo Filho

Leandro Maia Silva



- Assim como os vetores, as matrizes são estruturas de dados homogêneas. Podem ser construídas dos diversos tipos básicos primitivos (real, inteiro, caractere).
- Principal diferença em relação aos vetores (unidimensionais): possui uma ou mais dimensões adicionais.
- Maioria dos casos: utiliza-se matrizes bidimensionais.

- São utilizadas quando os dados homogêneos necessitam de uma estruturação com mais de uma dimensão.
- Exemplos:
 - Programar um jogo de xadrez (o tabuleiro é naturalmente bidimensional).
 - Estrutura para guardar caracteres de um livro (três dimensões: 2 para representar os caracteres de uma página e uma terceira para indicar as páginas).
 - Problemas matemáticos matriciais.
 - Processamento de imagens: Uma imagem pode ser vista como uma matriz de pixels.

Matrizes: Declaração

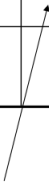
- A sintaxe para declaração de uma variável deste tipo é semelhante a declaração dos vetores. Considera-se porém a quantidade de elementos da outra dimensão:
tipo identificador[qtde_linhas][qtde_colunas];
- Nesse caso, os índices variam de 0 até $(qtde_linhas-1)$ para as linhas e 0 até $(qtde_colunas-1)$ para as colunas.
- Exemplos:
 - `int matriz[3][4];` //Matriz de 2 dimensões com elementos do tipo int
 - `float matriz_real[4][4][6];` //Matriz de 3 dimensões com elementos do tipo float
 - `char matriz_char[3][3];` //Matriz de 2 dimensões com elementos do tipo char

- Caso geral para declaração de uma matriz é:
tipo identificador[dim1][dim2]...[dimn];
- Assim temos uma matriz n-dimensional (n dimensões).
- Exemplo:
 - `int matriz[3][4][5][3];`
 - Os índices variam de 0 a 2 para a primeira dimensão, 0 a 3 para a segunda dimensão, 0 a 4 para a terceira dimensão e 0 a 2 para a quarta dimensão.

Matrizes: Declaração

Representação: Matriz 3x4

	0	1	2	3
0	0 0	0 1	0 2	0 3
1	1 0	1 1	1 2	1 3
2	2 0	2 1	2 2	2 3



mat[1][2]

- A atribuição a um valor na matriz é feito explicitando a posição da matriz em que o valor deverá ser atribuído.
- Exemplo:

```
float NUM[2][2];  
NUM[0][0] = 3.6;  
NUM[0][1] = 0.0;  
NUM[1][0] = 4.7;  
NUM[1][1] = 1.5;
```

- Os elementos das matrizes são referenciados individualmente por meio de índices (iniciando de zero) entre colchetes.
- Seja a matriz do exemplo anterior:
float A, B;
A = NUM[0][0];
B = NUM[1][0];
- Assim, A passa a ter o valor 3.6 e B o valor 4.7.

Matrizes: Exemplo

O programa a seguir, inicializa com zeros os elementos de uma matriz inteira **n** de 5 linhas e 4 colunas e imprime.

```
1 #include <stdio.h>
2
3 #define SUCESSO 0
4
5 int main(int argc, char ** argv) {
6     int n[5][4], i, j;
7     for (i = 0; i < 5; i++)
8         for (j = 0; j < 4; j++)
9             n[i][j] = 0;
10    printf("Matriz \n");
11    for (i = 0; i < 5; i++){
12        printf("\n Linha %2d \n", i);
13        for (j = 0; j < 4; j++)
14            printf("%d ", n[i][j]);
15    }
16    return SUCESSO;
17 }
```

Matrizes: Exemplo

O programa abaixo inicializa os elementos de uma matriz **m** com os valores iguais a soma dos índices de cada elemento e imprime cada valor.

```
1 #include <stdio.h>
2
3 #define SUCESSO 0
4
5 int main(int argc, char ** argv) {
6     int m[3][2], i, j;
7     for (i=0; i < 3; i++)
8         for (j=0; j < 2; j++) {
9             m[i][j] = i+j;
10             printf("i=%d j=%d elemento=%d \n", i, j, m[i][j]);
11         }
12     return SUCESSO;
13 }
```

- Matrizes serão passadas para subrotinas da mesma forma como vetores.
- Os valores das dimensões podem ser informados ou omitidos.
- Exemplo: Um procedimento para imprimir uma matriz teria a seguinte declaração:
imprimeMatriz(float mat[][3], int lin, int col)
ou
imprimeMatriz(float mat[3][3], int lin, int col)
- A declaração em C segue a mesma regra da pseudolinguagem, mas apenas a primeira dimensão pode ser omitida.
- Havendo mais dimensões, a mesma regra deverá ser seguida.

Criar uma função que receba uma matriz 2×3 de números reais e retorne a média dos valores da matriz. Crie uma função principal que chame a subrotina e imprima a média.

Exemplo

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 #define SUCESSO      0
5 #define NUM_LINHAS   2
6 #define NUM_COLUNAS  3
7
8 float mediaMatriz(float m[NUM_LINHAS][NUM_COLUNAS]) {
9     int i, j;
10    float media = 0;
11    for( i = 0; i < NUM_LINHAS; i++)
12        for(j = 0; j < NUM_COLUNAS; j++)
13            media += m[i][j];
14    return media / (1.0 * NUM_LINHAS * NUM_COLUNAS);
15 }
16
17 int main(int argc, char ** argv) {
18     float mat[NUM_LINHAS][NUM_COLUNAS] = {
19         {3.4, 5.6, 4.0},
20         {2.0, 1.1, 4.9}
21     };
22     float media = mediaMatriz(mat);
23     printf("A media da matriz foi %.2f", media);
24     return SUCESSO;
25 }
```

Faça um programa para exibir a soma de duas matrizes quadradas 3×3 . Deverá ser criado um procedimento para ler uma matriz (será chamado duas vezes com parâmetros diferentes) e um segundo procedimento que irá imprimir a soma das matrizes passadas como parâmetro.

Exemplo

```
1 #include <stdio.h>
2 #define NUM_LINHAS 3
3 #define NUM_COLUNAS 3
4
5 void leMatriz(int m[NUM_LINHAS][NUM_COLUNAS]) {
6     int i, j;
7
8     for(i = 0; i < NUM_LINHAS; i++)
9         for(j = 0; j < NUM_COLUNAS; j++)
10             scanf("%d", &m[i][j]);
11 }
12
13 void somaMatriz(int m1[NUM_LINHAS][NUM_COLUNAS],
14                 int m2[NUM_LINHAS][NUM_COLUNAS]) {
15
16     int soma[NUM_LINHAS][NUM_COLUNAS], i, j;
17
18     for(i = 0; i < NUM_LINHAS; i++) {
19         for(j = 0; j < NUM_COLUNAS; j++) {
20             soma[i][j] = m1[i][j] + m2[i][j];
21         }
22     }
23     printf("A soma das matrizes eh: ");
24     for(i = 0; i < NUM_LINHAS; i++){
25         printf("\n");
26         for(j = 0; j < NUM_COLUNAS; j++) {
27             printf("%d ", soma[i][j]);
28         }
29     }
30 }
31 // Continua no próximo slide
```

Exemplo

```
1
2 // Continuação
3
4 #define SUCESSO 0
5
6 int main( int argc , char ** argv ) {
7
8     int m1[NUM_LINHAS][NUM_COLUNAS] , m2[NUM_LINHAS][NUM_COLUNAS];
9
10    printf( "Digite sua primeira matriz" );
11    leMatriz( m1 );
12    printf( "Digite sua segunda matriz" );
13    leMatriz( m2 );
14    somaMatriz( m1, m2 );
15
16    // Se chegou até aqui é porque correu tudo bem
17    return SUCESSO;
18 }
```


1. Faça um programa que leia uma matriz 5×5 e um procedimento que imprima sua transposta.
2. Crie uma função que receba uma matriz de números reais e retorne a soma dos elementos desta matriz.
3. Faça um programa que leia um vetor de dimensão 5 e uma matriz quadrada de dimensão 5. Crie um procedimento que multiplique o vetor pela matriz. Imprima o resultado.
4. Faça um programa para ler a quantidade de um total de 5 produtos que uma empresa tem em suas 7 lojas e imprimir em uma tabela:
 - a) o total de cada produto nestas lojas
 - b) a loja que tem menos produtos