

# Linguagem C Matrizes



Professora: Emanoeli Madalosso emanoelim@utfpr.edu.br

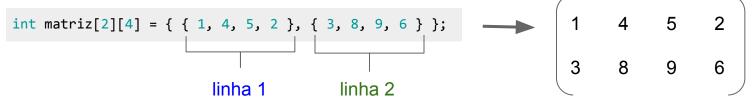


- Uma matriz é um vetor multidimensional;
- Exemplo: declarar uma matriz bidimensional de números inteiros de tamanho 2x4 (2 linhas e 4 colunas):

```
int matriz[2][4];
```



- Atribuição de valores:
  - Inicialização:





- Atribuição de valores:
  - Acesso através dos índices:

```
int matriz[2][4];

matriz[0][0] = 1;
matriz[0][1] = 4;
matriz[0][2] = 5;
matriz[0][3] = 2;
matriz[1][0] = 3;
matriz[1][1] = 8;
matriz[1][2] = 9;
matriz[1][3] = 6;
```

Leitura de uma matriz:

```
int matriz[2][4];
int l, c;

for(l=0; 1<2; l++) {
    for(c=0; c<4; c++) {
        printf("Informe o elemento %d, %d: ", l, c);
        scanf("%d", &matriz[l][c]);
    }
}</pre>
```



Saída de uma matriz:

```
int matriz[2][4];
int 1, c;
for(1=0; 1<2; 1++) { // leitura
    for(c=0; c<4; c++) {
        printf("Informe o elemento %d, %d: ", 1, c);
        scanf("%d", &matriz[1][c]);
for(1=0; 1<2; 1++) { // Saída
    for(c=0; c<4; c++) {
        printf("%d\t", matriz[l][c]);
    printf("\n"); -
```

Mostrar as linhas separadas

 Assim como em vetores, os elementos da matriz ficam alocados de maneira contígua na memória do computador.



#### • Exemplo:

0, 0

1	9	0
8	5	4
3	7	2

2, 2

• • •

10000	1
10004	9
10008	0
10012	8
10016	5
10020	4
10024	3
10028	7
10032	2

Uma linha "em baixo" da outra

... 8

- Observações:
  - Cuidado ao inicializar a matriz:

```
int mat[2][2] = \{ \{1, 2\}, \{3, 4\} \}; // \text{ duas dimensões informadas - válido } // int <math>mat[][2] = \{ \{1, 2\}, \{3, 4\} \}; // \text{ dimensão mais à esquerda omitida - válido } // int <math>mat[2][] = \{ \{1, 2\}, \{3, 4\} \}; // \text{ dimensão mais à direita omitida - ERRO } // int <math>mat[][] = \{ \{1, 2\}, \{3, 4\} \}; // \text{ nenhuma dimensão informada - ERRO }
```