

## Curso Profissional: Programador/a de Informática

PSD – 10.º ano: UFCD 0809 - Programação em C/C++ - fundamentos

### Ficha de Trabalho 11

Ano letivo 21/22

## Matrizes e vetores multidimensionais

Em C um vetor declarado com 2 dimensões, não é chamado de matriz, mas sim de vetor de vetores.

**Sintaxe (vetor multidimensional):** tipo nome [ dim1 ] [ dim2 ] ... [ dim n ] ;

**Exemplo:** char galo [ 3 ] [ 3 ] ;

		j ↓		
		0	1	2
i →	0	[ 0 ] [ 0 ]	[ 0 ] [ 1 ]	[ 0 ] [ 2 ]
	1	[ 1 ] [ 0 ]	[ 1 ] [ 1 ]	[ 1 ] [ 2 ]
	2	[ 2 ] [ 0 ]	[ 2 ] [ 1 ]	[ 2 ] [ 2 ]

Para percorrer a “matriz” linha a linha, fixa-se a linha (índice i) e faz-se variar a coluna (índice j). Por exemplo:

```
for ( i = 0; i<3; i++)
    for ( j = 0; j<3; j++)
        scanf ("%d", &m[i ] [j ] );
```

Para percorrer a “matriz” coluna a coluna, fixa-se a coluna (índice j) e faz-se variar a linha (índice i). Por exemplo:

```
for ( j = 0; j<3; j++)
    for ( i = 0; i<3; i++)
        scanf ("%d", &m[ i ][ j ] );
```

ou

```
for ( i = 0; i<3; i++)
    for ( j = 0; j<3; j++)
        scanf ("%d", &m[ j ][ i ] );
```

## Inicialização automática (dimensão 2):

### Sintaxe:

tipo nome  $\underbrace{[n^{\circ} \text{ de el.}]}_n \underbrace{[n^{\circ} \text{ de el.}]}_m = \{ \{ \text{valor 1, ..., valor n} \}, \dots, \{ \text{valor 1, ..., valor n} \} \};$

**Exemplo:** suponhamos que queremos inicializar um vetor de 3 caracteres com espaços:

```
char v [ 3 ] = { '\0', '\0', '\0' } ;
```

Para inicializar um vetor de 3 caracteres de dimensão 2, fazemos:

```
char galo [ 3 ] [ 3 ] = { { '\0', '\0', '\0' }, { '\0', '\0', '\0' }, { '\0', '\0', '\0' } } ;
```

```
# include <stdio.h>
# define DIM 3
main()
{
char galo[ DIM ][ DIM ] = { { '\0', '\0', '\0' }, { '\0', '\0', '\0' }, { '\0', '\0', '\0' } }; int i,j;
galo[0][0] = 'X' ; galo[1][1] = 'X' ;
galo[2][2] = 'O' ; galo[3][3] = 'O' ;
for ( i=0; i<DIM; i++)
{
for ( j=0; j<DIM; j++)
printf(“%c %c “, galo[i][j], j==DIM-1 ? '\0' : '|' );
if ( i!= DIM-1)
printf(“\n -----\n” );
}
}
```

**OUTPUT:**

```
X | | O
-----
|X|
-----
| | O
```

## EXERCÍCIOS RESOLVIDOS

**R1.** Dado uma matriz A definida como int A [5] [5];

Preenchê-la com 5x5 números inteiros introduzidos pelo utilizador e exibi-la, depois de lida, no monitor:

```
#include <stdio.h>
main()
{
int i,j,A[5] [5];
for (i=0; i<5; i++)
for ( j=0; j< 5;j++)
scanf(“%d”,&A[i][j]);
for(i=0; i<5; i++)
{
for ( j=0; j< 5;j++)
printf(“%d ”,A[i][j]);
printf(“\n”);
}
}
```

- R2.** Fazer um programa para ler uma matriz 3 x 5 de números inteiros e escrevê-la após ter multiplicado cada elemento por 2.

```
#include <stdio.h>
main()
{
    const int NL = 3, //número de linhas
           NC = 5,     //número de colunas
           K = 2;      //fator para multiplicação
    int i,j, M [NL][NC];

    //leitura da matriz

    for ( i=0;i<NL; i++)
        for ( j=0; j<NC; j++)
        {
            printf("Elemento da linha %d coluna %d : ",i,j);
            scanf("%d", &M[i][j]);
        }

    //cálculo da multiplicação

    for (i=0; i<NL; i++)
        for ( j=0; j< NC; j++)
            M[i][j] = M[i][j] * K;

    //exibição da matriz resultante
    printf ("Resultado: \n");
    for (i=0; i<NL; i++)
    {
        for ( j=0; j< NC; j++)
            printf("%d ", M[i][j]);
        printf("\n");
    }
}
```

- R3.** Dada uma matriz de 4 x 5 elementos inteiros, calcular a soma de cada linha, de cada coluna e de todos os seus elementos.

Obs: utiliza um vetor para armazenar o resultado da soma de cada linha e outro para a soma de cada coluna.

```
#include <stdio.h>
main()
{
    int M [4][5], L[4], C[5], I, J, SOMA;

    // leitura da matriz
    for (i=0; i<4; i++)
        for ( j=0; j< 5; j++)
        {
            printf("Elemento da linha %d coluna %d : ",i, j);
            scanf("%d", &M[i][j]);
        }
}
```

```

//cálculo da soma dos elementos de cada linha
for (i=0; i<4; i++)
{
    L[i] = 0;
    for ( j=0; j< 5; j++)
        L[i] = L[i] + M[i][j];
}
//cálculo da soma dos elementos de cada coluna
for ( j=0; j< 5; j++)
{
    C[j] = 0;
    for (i=0; i<4; i++)
        C[j] = C[j] + M[i][j];
}
// cálculo da soma de todos os elementos da matriz

SOMA= 0;
for (i=0; i<=4; i++)
    for ( j=0; j<= 5; j++)
        SOMA = SOMA + M[i][j];

// exibição dos resultados

for (i=0; i<4; i++)
    printf ("Soma da Linha: %d= %d\n",i, L[i]);
for ( j=0; j< 5; j++)
    printf("\nSoma da Coluna %d: %d\n",j, C[j]);
printf("\nSoma da Matriz: %d", SOMA);
}

```

## EXERCÍCIOS:

- 1- Escreve um programa que leia as notas de exame (30% da nota) e de frequência (70% da nota) de 5 alunos e que calcule e exiba a nota final obtida por cada um deles.
- 2- Dada uma matriz A com 3 x 4 elementos reais calcular e exibir o maior elemento da mesma.
- 3- Dadas duas matrizes A e B, com 2 x 3 elementos inteiros cada, gerar e exibir uma matriz C do mesmo tamanho, que resultará da soma da matriz A com a matriz B.
- 4- Faz um programa que leia uma matriz de ordem 3 x 5 de elementos inteiros, calcular e exibir:
  - a) A soma dos elementos da matriz;
  - b) A média dos elementos da matriz;

- 5- Dada uma matriz quadrada (i.é.  $n^{\circ}$  de linhas é = ao  $n^{\circ}$  de colunas) de ordem N, de elementos inteiros, exibir os elementos da diagonal principal, isto é, os elementos onde  $i = j$ . **Obs:** N será lido ( $N \leq 10$ ).
- 6- Faz um programa que crie um array tridimensional  $5 \times 7 \times 3$ , onde o conteúdo de cada elemento é igual a soma de seus índices.
- 7- Elabora um programa que ordene um vetor com N elementos do tipo inteiro por ordem decrescente.
- 8- Elabora um programa que dado um vetor com N elementos:
  - a) O inverta e todos os elementos cujo conteúdo seja um número par devem ser substituídos pelo valor 0 (zero).
- 9- Elabora um programa que determine e imprima a soma dos elementos positivos e negativos duma matriz.
- 10- Num concurso para admissão de empregados de um escritório, os 100 candidatos selecionados por entrevista, terão de prestar seguidamente uma prova de dactilografia, me que o  $n^{\circ}$  de palavras é fornecido e é igual para todos os candidatos.

Pretende-se escrever um programa que dado o nome, o número de palavras e o número de erros de cada candidato determine a pontuação obtida sabendo que:

$$\text{PONTUAÇÃO} = (\text{número de palavras} - 10 * \text{número de erros}) / 5$$

O resultado do teste tem os seguintes valores:

Reprovado se a pontuação for  $< 45$

Aprovado se a pontuação for  $> 52$

Novo teste se a pontuação for  $\geq 45$  e pontuação  $\leq 52$

Deve ser dado a conhecer o nome, o número de palavras

**Nota:** Utiliza arrays unidimensionais para armazenar o nome, o número de erros, a pontuação e o resultado de cada aluno.

- 11- Para uma matriz A de  $2 \times 3$  elementos, do tipo inteiro:
  - a. Ler a matriz A;
  - b. Determinar T, matriz transposta de A;
  - c. Imprimir. Por linhas, a matriz A e a matriz T