



# Laboratório de Informática

## Vetor e Matriz – Aula 09

Profª Ms. Lília Marta Brandão Soussa Modesto

# Estrutura de dados – vetores

---

- A variável simples armazena somente um valor de cada vez.
- As matrizes são variáveis que armazenam vários valores ao mesmo tempo, que possuem os mesmos tipos de dados, sendo chamadas de estruturas de dados homogêneas.
- As matrizes também são chamadas de:  
tabelas, arrays, variáveis indexadas, variáveis compostas, variáveis subscritas ou arranjos.
- A Matriz unidimensional é chamada de **Vetor**.

# Estrutura de dados – vetores

**Matriz:**

		Coluna				
		↓				
		0	1	2	3	4
Linha →	0					
	1					
	2					
	3					

**Vetor:**

		Coluna				
		↓				
		0	1	2	3	4
Linha →	0					

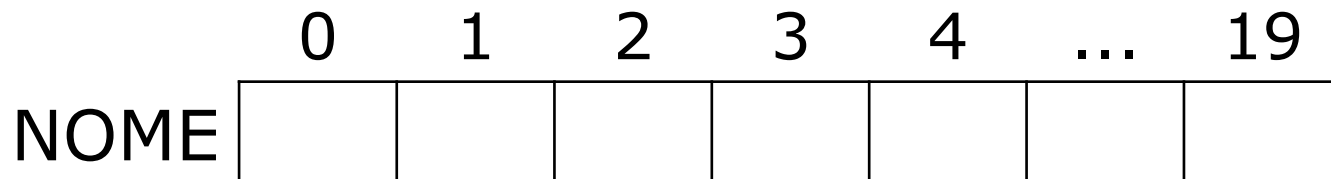
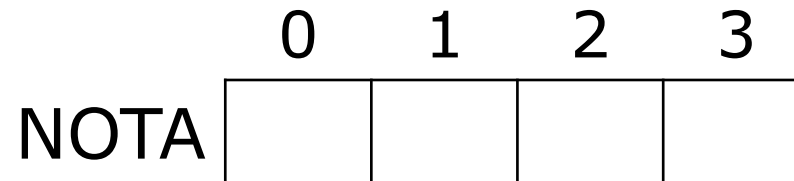
# Estrutura de dados – vetores

**Sintaxe:** tipo <nome vetor> [dimensão];

exemplos:

float NOTA [4]; (Obs: de 0 a 3)

char NOME [20]; (Obs: de 0 a 19)



# Estrutura de dados – vetores

## Exemplo 1:

Crie um programa que calcule e apresente a média geral de uma turma de 8 alunos.

0	1	2	3	4	5	6	7
4.5	6.5	8.0	3.5	6.0	7.0	6.5	6.0

MEDIA da TURMA = 6.0

# Estrutura de dados – vetores

```
#include <stdio.h>
#include <stdlib.h>
int main(void)
{
    system ("color f0");
    int I;
    float SOMA = 0, MEDTUR, MEDIA[8];
    printf ("\n Digite 8 medias: ");
    for (I = 0; I < 8; I++)
    {
        printf ("\n Digite a media %d: ", I+1);
        scanf ("%f", &MEDIA[I]);
        SOMA += MEDIA[I];
    }
    MEDTUR = SOMA / 8;
    printf ("\nMedia da Turma = %.2f \n\n", MEDTUR);
    system ("pause");
    return(0);
}
```

# Estrutura de dados – vetores

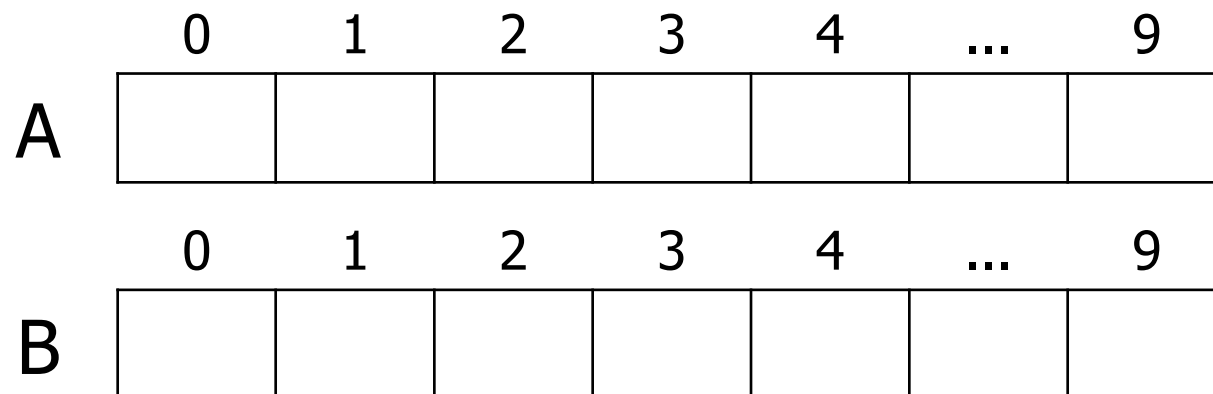
## Exemplo 2:

Crie um programa que:

Leia dez elementos de um vetor A;

Construir um vetor B de mesma dimensão,  
que é formado pela multiplicação dos  
elementos de A por 5;

Mostrar o conteúdo do vetor B.



# Estrutura de dados – vetores

```
#include <stdio.h>
#include <stdlib.h>
int main(void)
{
    system("color f0");
    int I, A[10], B[10];
    printf ("\n Digite 10 valores inteiros: \n ");
    for (I = 0; I < 10; I++)
    {
        printf ("\n Digite o %do.: ", I+1);
        scanf("%d", &A[I]);
        B[I] = A[I] * 5;
    }
    for (I = 0; I < 10; I++)
        printf ("\n B[%d] = %d\n", I, B[I]);
    system("pause");
    return(0);
}
```



# Exercício

---

- 1) Ler 10 elementos de uma matriz tipo vetor e apresentá-los.
- 2) Ler 8 elementos em uma matriz A tipo vetor. Construir uma matriz B de mesma dimensão com os elementos da matriz A multiplicados por 3. O elemento  $B[1]$  deverá ser implicado pelo elemento  $A[1] * 3$ , o elemento  $B[2]$  implicado pelo elemento  $A[2] * 3$  e assim por diante, até 8. Apresentar a matriz B.
- 3) Ler duas matrizes A e B do tipo vetor com 20 elementos. Construir uma matriz C, onde cada elemento de C é a subtração do elemento correspondente de A com B. Apresentar a matriz C.

# Exercício

---

- 4) Ler 15 elementos de uma matriz A do tipo vetor. Construir uma matriz B de mesmo tipo, observando a seguinte lei de formação: "Todo elemento de B deverá ser o quadrado do elemento de A correspondente". Apresentar as matrizes A e B.
- 5) Ler uma matriz A do tipo vetor com 15 elementos. Construir uma matriz B de mesmo tipo, sendo que cada elemento da matriz B seja o fatorial do elemento correspondente da matriz A. Apresentar as matrizes A e B.
- 6) Ler duas matrizes A e B do tipo vetor com 15 elementos cada. Construir uma matriz C, sendo esta a junção das duas outras matrizes. Desta forma, C deverá ser o dobro de elementos, ou seja, 30. Apresentar a matriz C.

# Exercício

---

- 7) Ler 20 elementos de uma matriz A tipo vetor e construir uma matriz B de mesma dimensão com os mesmos elementos da matriz A, sendo que deverão estar invertidos. Ou seja, o primeiro elemento de A passa a ser o último de B, o segundo elemento passa a ser o penúltimo de B e assim por diante. Apresentar as matrizes A e B.
- 8) Ler três matrizes (A, B e C) de uma dimensão com 5 elementos cada. Construir uma matriz D, sendo esta a junção das três outras matrizes. Desta forma D deverá ter o triplo de elementos, ou seja, 15. Apresentar os elementos da matriz D.

# Exercício

---

- 9) Elaborar um programa que efetue o cálculo de uma tabuada de um número qualquer e armazene os resultados em uma matriz A de uma dimensão para 10 elementos. Apresentar os valores armazenados na matriz.
- 10) Ler 6 elementos (valores inteiros) para as matrizes A e B de uma dimensão do tipo vetor. Construir as matrizes C e D de mesmo tipo e dimensão, sendo que a matriz C deverá ser formada pelos elementos de índice ímpar das matrizes A e B, e matriz D deverá ser formada pelos elementos de índice par das matrizes A e B. Apresentar as matrizes C e D.

# Estrutura de dados – matrizes

## Sintaxe:

tipo <nome matriz> [dimensão linha][dimensão coluna];

## Exemplos:

```
float NOTA [30][4];
```

```
char NOME [10][40];
```

NOTA

	0	1	2	3
0				
1				
...				
29				

NOME

	0	1	2	3	...	39
0						
1						
...						
9						

# Estrutura de dados – matrizes

## Exemplo 3:

Crie um programa que:

Leia duas matrizes A e B com 5 linhas e 3 colunas;

Construir uma matriz C de mesma dimensão, que é formada pela soma dos elementos de A com B;

Mostrar o conteúdo da matriz C.

# Estrutura de dados – matrizes

```
#include <stdio.h>
#include <stdlib.h>
int main(void)
{
    system("color f0");
    int I, J, A[5][3], B[5][3], C[5][3];
    printf ("\n Digite 15 números para A: ");
    for (I = 0; I < 5; I++)
        for (J = 0; J < 3; J++)
        {
            printf ("\n Digite o A[%d][%d]: ", I, J);
            scanf("%d", &A[I][J]);
        }
    printf ("\n Digite 15 números para B: ");
```

# Estrutura de dados – matrizes

```
for (I = 0; I < 5; I++)
    for (J = 0; J < 3; J++)
    {
        printf ("\n Digite B[%d][%d]: ", I, J);
        scanf("%d", &B[I][J]);
    }
for (I = 0; I < 5; I++)
    for (J = 0; J < 3; J++)
    {
        C[I][J] = A[I][J] + B[I][J];
        printf ("\n C[%d][%d] = %d ", I, J, C[I][J]);
    }
printf("\n\n");
system("pause");
return(0);
```

```
}
```



# Estrutura de dados – matrizes

## Exemplo 4:

Crie um programa que:

- Leia 10 nomes de pessoas;

- Mostre-os na tela.

Considere que cada nome tenha 40 caracteres de comprimento.

# Estrutura de dados – matrizes

```
#include <stdio.h>
#include <stdlib.h>
int main(void)
{
    system("color f0");
    int I;
    char NOME [10] [40];
    for (I = 0; I < 10; I++)
    {
        printf ("\n Digite o %2do. nome: ", I+1);
        fflush(stdin);
        fgets(NOME [I], 40, stdin);
    }
    for (I = 0; I <= 9; I++)
        printf ("\n\n %2do. Nome: %s", I+1, NOME[I]);
    system("pause");
    return(0);
}
```

# Exercício

---

- 11) Ler dois vetores  $A$  e  $B$ , cada um com 7 elementos. Construir uma matriz  $C$  de duas dimensões. A primeira coluna deve ser formada pelos elementos de  $A$  e a segunda coluna deve ser formada pelos elementos de  $B$ .
- 12) Ler dois vetores  $A$  e  $B$ , cada um com 12 elementos. Construir uma matriz  $C$  de duas dimensões. A primeira coluna de  $C$  deve ser formada pelos elementos de  $A$  multiplicados por 2 e a segunda coluna deve ser formada pelos elementos de  $B$  subtraídos de 5.

# Exercício

---

- 13) Ler um vetor A de 10 elementos. Construir uma matriz B de duas dimensões com três colunas. A primeira coluna de B é formada pelos elementos de A somados com mais 5, a segunda coluna é formada pelo valor do cálculo da fatorial de cada elemento correspondente de A e a terceira e última coluna deve ser formada pelos quadrados dos elementos correspondentes de A.
- 14) Ler uma matriz A de 5 linhas e 5 colunas. Construir uma matriz B de mesma dimensão, sendo que cada elemento de B deverá ser o dobro de cada elemento correspondente de A, exceto para os valores situados na diagonal principal, os quais deverão ser o triplo de cada elemento correspondente de A. Imprimir B.

# Matriz

Construa um programa que receba uma matriz quadrada de dimensão 4X4 e classifique a matriz como SUPERIOR ou INFERIOR. Uma matriz é SUPERIOR se a soma dos elementos acima da diagonal é maior ou igual a soma dos elementos embaixo da diagonal; caso contrário diz-se que é uma matriz inferior. Veja exemplo na figura a seguir:

1	3	5	8
2	5	6	8
3	3	8	1
0	2	1	3

Soma  : 31

Soma  : 11

**MATRIZ SUPERIOR**

1	3	5	1
2	5	3	2
3	3	8	1
9	6	1	3

Soma  : 15

Soma  : 24

**MATRIZ INFERIOR**