

# Algoritmos e Programação de Computadores II (APC II)

## Aula 2 – Revisão: Vetores e Matrizes

Profa. Carla Gonçalves Pelissoni





# Vetores

**Vetores =**

**Variáveis Compostas Unidimensionais  
Homogêneas**

- Correspondem a posições de memória, identificadas por um mesmo nome, individualizadas por índices e cujo conteúdo é do mesmo tipo.

# Vetores

- **Exemplo:**
  - Conjunto de notas de 100 alunos de uma disciplina.
  - Associa-se o identificador **Notas** que passará a identificar não uma única posição de memória, mas 100 posições.
  - A referência ao *n-ésimo* elemento do conjunto será identificado pela notação **Notas[i]**, onde *i* é um número inteiro ou uma variável numérica contendo um valor inteiro que identifica a posição do elemento no vetor.



# Vetores

## Exemplo:

Notas[0] = 9.0

Notas[2] = 5.5

Pode-se usar um índice (i) para se ter acesso a qualquer posição do vetor.

i = 5, Notas[i] = 4.0

Notas

9.0	0
8.0	1
5.5	2
6.5	3
2.0	4
4.0	5



# Vetores

- Outros exemplos:

Nome	
'J'	0
'O'	1
'A'	2
'O'	3

Idade	
18	0
54	1
40	2
68	3
20	4
4	5



# Vetores

- Declaração de uma Variável do Tipo Vetor (em Linguagem C):

tipo nome[TAMANHO];

- Onde:
  - **nome** é o nome que se dá ao conjunto
  - **tipo** é o tipo que se quer associar o conjunto (*int*, *float*, *char*, por exemplo)
  - **TAMANHO** é a quantidade máxima de elementos do vetor.
- **Exemplo:**

float Notas[10];

limite inferior = notas[0]

limite superior= notas[9]



# Vetores

- **Acessando** um elemento do vetor
  - É preciso utilizar um *índice sempre!*
  - Notas[2]: representa o terceiro elemento do vetor (primeiro elemento é o de posição 0).
  - Pode-se usar uma variável para representar o índice para se ter acesso a qualquer posição do vetor. Como por exemplo, variável *i*: **Notas[i]**.
  - Se  $i = 5$ , estará sendo feito o acesso à 6ª posição do vetor Notas (**pois o 1º índice é sempre 0**).



# Vetores

- **Armazenando** dados no vetor (Exemplo: armazena 10 elementos)

```
.....  
for (i=0; i<10; i++)  
{  
    cout << "Digite a nota do aluno:";  
    cin >> Notas[i];  
}
```





# Vetores

- **Utilizando** dados do vetor de 10 elementos (soma dos elementos do vetor e cálculo da média aritmética):

```
....  
soma=0;  
for (i=0; i<10; i++)  
{  
    soma = soma + Notas[i];  
}  
media = soma/10;  
cout << "Media das notas: " << media;  
....
```



# Vetores

- **Mostrando** na tela os dados de um vetor (Exemplo: 10 elementos)

```
.....  
for (i=0; i<10; i++)  
{  
    cout << "\nElemento " << i << ": ";  
    cout << Notas[i];  
}
```



# Exemplo 1

- Escrever um programa em C que leia um conjunto de 10 notas, armazene-as num vetor Notas e calcule a média aritmética dessas notas.



# Exemplo 1

```
#include<iostream.h>
main()
{
    int i;
    float Notas[10], media, soma=0;

    /* Faz a leitura de cada nota e a somatória com a
       variável soma. */
    for (i=0; i < 10; i++)
    {
        cout << "\nDigite uma nota: ";
        cin >> Notas[i];
        soma = soma + Notas[i];
    }
    media = soma/10;
    cout << "Media da turma = " << media;
}
```



## Exemplo 2

- Fazer um programa em C que leia um conjunto de 50 notas e armazene-as num vetor. Calcule e escreva a quantidade de alunos com nota superior à 7.0.





## Exemplo 2

```
#include<iostream.h>
main()
{
    int i;
    float Notas[50], cont=0;
    for (i=0; i < 50; i++)
    {
        cout << "\nDigite uma nota: ";
        cin >> Notas[i];
        if(Notas[i] > 7)
            cont++;
    }
    cout << "Quantidade de notas maior que 7 = " <<
cont;
}
```





## Exemplo 3

- Faça um programa em C para ler 5 números inteiros quaisquer e armazená-los em um vetor. Em seguida, imprimir o vetor original e o mesmo em ordem inversa.
- Por exemplo:

vetor original:        5   -3   6   67   -10

ordem inversa:    -10   67   6   -3   5



## Exemplo 3

```
#include <iostream.h>
main() {
    int vetor[5], i;
    // Leitura dos elementos do vetor
    for(i=0; i<5; i++){
        cout << "\nDigite um numero inteiro: ";
        cin >> vetor[i];
    }
    // Escrita do vetor original
    cout << "O vetor original é: ";
    for(i=0; i<5; i++)
        cout << vetor[i] << " ";
    // Escrita do vetor em ordem inversa
    cout << "O vetor em ordem inversa é: ";
    for(i=4; i>=0; i--)
        cout << vetor[i] << " ";
}
```





# Matrizes

- **Matrizes = Variáveis Compostas Bidimensionais Homogêneas**
- Conjunto de dados referenciado por um mesmo nome e que necessita de **dois índices** para ter seus elementos individualizados.
- MATRIZES:

$$\begin{array}{cccccc} \mathbf{A} = & a_{00} & a_{01} & a_{02} & \dots & a_{0n-1} \\ & a_{10} & a_{11} & a_{12} & \dots & a_{1n-1} \\ & a_{20} & a_{21} & a_{22} & \dots & a_{2n-1} \\ & \dots & \dots & \dots & \dots & \dots \end{array} \Rightarrow \begin{array}{l} A_{m \times n} \\ m = \text{linha} \\ n = \text{coluna} \end{array}$$





# Matrizes

- Usa-se sempre **2 índices**: um para percorrer as linhas e outro para as colunas (geralmente, são chamados de *i* e *j*).
- **Exemplo**: matriz de números para cálculos matemáticos.

$$\text{Matriz}_{3 \times 3} = \begin{matrix} 199 & 35 & 57 \\ 29 & 10 & 6 \\ 30 & 0 & 27 \end{matrix}$$

$$\text{Matriz}[0][1] = 35$$

$$\text{Matriz}[1][2] = 6$$

$$\text{Matriz}[i][j] = 30, \text{ supondo } i=2 \text{ e } j=0$$





# Matrizes

- Se o número de linhas é igual a  **$m$** , o índice que percorre as linhas varia de **0 a  $m-1$** .
- Se o número de colunas é igual a  **$n$** , o índice que percorre as colunas varia de **0 a  $n-1$** .
- Por exemplo  $\text{Mat}_{3 \times 4}$ : o índice para linha varia de 0 a 2 e o índice para coluna de 0 a 3.





# Matrizes

- **Declaração:**

Tipo NomeMatriz[MAX\_LINHAS][MAX\_COLUNAS];

- **Exemplo:**

```
float Matriz[3][8];
```

primeira posição: Matriz[0][0];

última posição: Matriz[2][7];





# Matrizes

- **Armazenando** elementos em uma matriz 3x4.

```
//Percorre as linhas
for (i=0; i < 3; i++)
{
    //Percorre as colunas
    for (j=0; j < 4; j++)
    {
        cout << "\nDigite um valor: ";
        cin >> Matriz[i][j];
    }
}
```





# Matrizes

- **Mostrando na tela os elementos de uma matriz 3x4.**

```
cout << "\nOs elementos da matriz são: ";  
//Percorre as linhas  
for (i=0; i < 3; i++)  
{  
    //Percorre as colunas  
    for (j=0; j < 4; j++)  
    {  
        cout << Matriz[i][j] << "  ";  
    }  
    cout << "\n";  
}
```



# Exemplo

- O programa a seguir lê um conjunto de números inteiros e armazena em uma matriz  $\text{Mat}_{3 \times 4}$ . O programa calcula a média aritmética dos elementos dessa matriz e escreve o resultado.





# Exemplo

```
#include <iostream.h>
main()
{
    int Matriz[3][4], i, j;
    float soma=0;
    cout << "Digite os valores da matriz:\n";
    for (i=0; i < 3; i++)
    {
        for (j=0; j < 4; j++)
        {
            cout << "\nLinha " << i << ", Coluna " << j << ": ";
            cin >> Matriz[i][j];
            soma = soma + Matriz[i][j];
        }
    }
    cout << "\nOs elementos digitados para a matriz foram:\n";
    for (i=0; i < 3; i++)
    {
        for (j=0; j < 4; j++)
        {
            cout << Matriz[i][j] << " ";
        }
        cout << "\n";
    }
    cout << "\nMedia dos elementos da matriz: " << soma/(3*4);
    system ("pause");
}
```







# Exercícios de Revisão



- 1) Fazer um programa em C para ler 10 números inteiros positivos e armazená-los num vetor. Em seguida, calcule e mostre a média aritmética apenas dos números ímpares.
- 2) Faça um programa em C que leia 50 números inteiros positivos e armazene-os num vetor **A**. A partir dos dados do vetor **A**, gere dois outros vetores:
  - Vetor **B**: só números pares;
  - Vetor **C**: somente os números ímpares.

Escreva os dois vetores resultantes.





# Exercícios de Revisão



- 3) Faça um programa que preencha uma matriz 3x5 com valores numéricos, calcule e mostre a quantidade de elementos entre 15 e 20.
- 4) Fazer um programa em C que leia uma matriz 4x3 e imprima o maior valor e o menor valor da matriz.

