



Vetores

Professor Dr Francisco Isidro Massetto
francisco.massetto@ufabc.edu.br

Vetores

- Nem sempre os tipos básicos: (inteiro, real, caracter e lógico) são suficientes para implementar um algoritmo.
 - Por exemplo: Considere o problema para ler 5 notas de uma turma e imprimir média dessas notas.
-

Vetores

```
import java.util.*;
class Main{
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);
        double nota, soma=0, media=0;
            int i;
            for (i=0;i<5;i++){
                System.out.println("Informe a nota
valor: ");
                nota = in.nextDouble();
                soma = soma + nota;
            }
            media=soma/i;
            System.out.println("A MÉDIA É "+media);
        }
    }
```

Vetores – Estudo de Caso

- E se eu quiser imprimir a **terceira** nota, após o loop?
 - Neste algoritmo os valores das notas não são armazenados em lugar nenhum.
 - Para solucionar essa situação utilizaremos **VETORES!!!**
-

Vetores

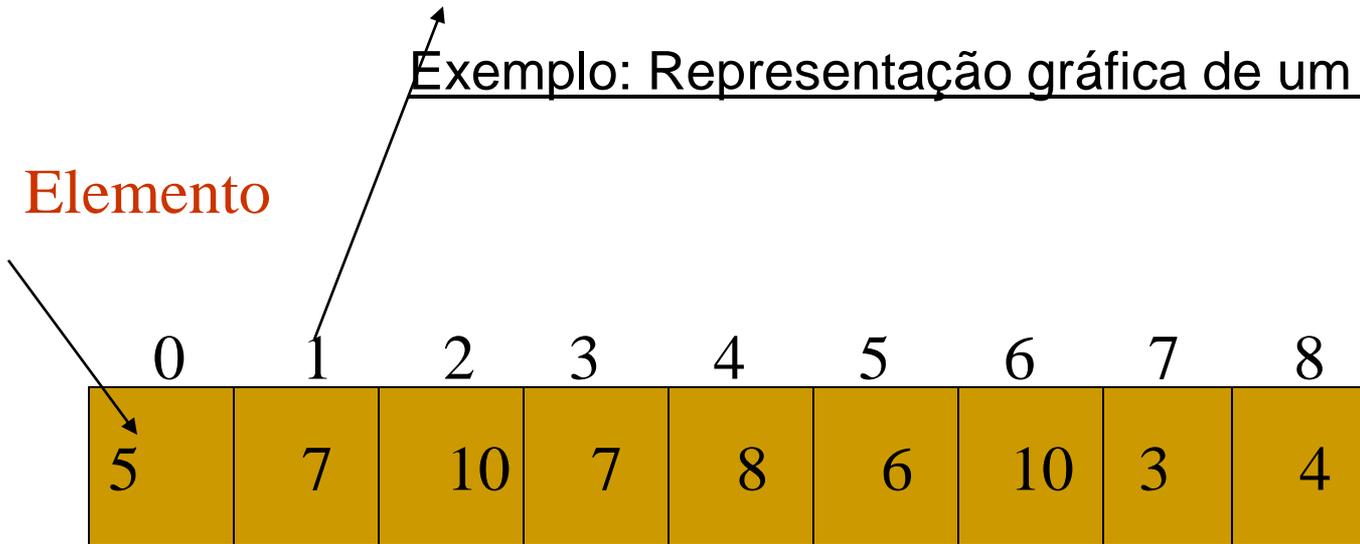
- Definição de vetores:
 - “Coleção de variáveis de mesmo tipo”.
 - “Estrutura de dados homogênea”.
 - Porque homogênea?
 - Agrupam várias informações
 - Dentro de uma **mesma** variável e
 - Com o **mesmo** tipo de dado
-

Vetores

Índice

Exemplo: Representação gráfica de um vetor

Elemento



Tamanho: 9 posições

Cada elemento ocupa uma posição diferente.

Genericamente chamamos as posições de **ÍNDICES**.

Vetores

- Ex: Agrupar as médias de 5 alunos

Alunos	Medias
1	4.5
2	6.5
3	8.0
4	3.5
5	6.0



**Mesmo tipo
de dado
tipo REAL**

- E onde ficam armazenados os dados?
 - ▶ De forma agrupada na **Memória Principal** !

Vetores

- Um vetor é representado por:
 - **Identificador** - nome do vetor
 - **Tamanho (dimensão)** - quantidade de elementos (dados)
 - **Tipo** - tipo de dado armazenado (integer, real, string)
- **Sintaxe:**

```
tipo identificador [ ] = new tipo[tamanho];
```

```
double exemplo[ ] = new double [55];
```

Vetores

- Como seria a declaração de um vetor para 80 notas?
 - **double notas [] = new double [80];**
-

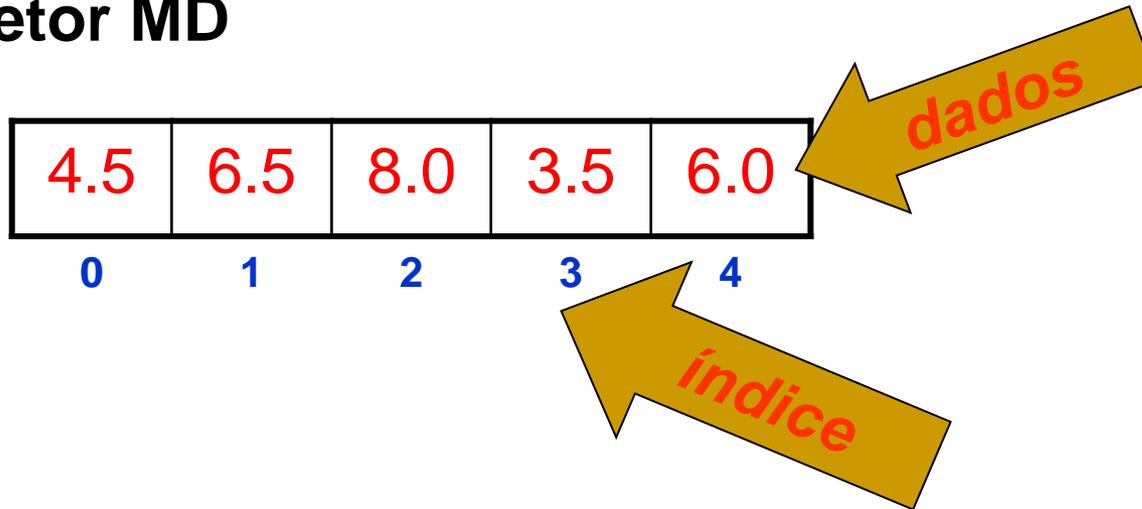
Manipulação de Vetores

- Os dados de um vetor são manipulados através de um índice
- O **índice** indica a **posição do dado** no vetor:
double notas[] = new double [5];
- Como identificar cada dado no vetor?
Através do **ÍNDICE!!**

MD guarda as notas dos 5 alunos do exemplo anterior

Manipulação de Vetores

Vetor MD



notas[0] = 4.5;
notas[1] = 6.5;
notas[2] = 8.0;
notas[3] = 3.5;
notas[4] = 6.0;

índice *dados*

Atribuição de valores em um vetor

Manipulação de Vetores

- Mas como caminhar entre as posições do vetor?
 - Utilizando estrutura de repetição!! (FOR)

```
for (i=0; i<5; i++)  
{  
    System.out.println("Valor" + i);  
    notas[i] = in.nextDouble();  
}
```

notas

4.5	6.5	8.0	3.5	6.0
-----	-----	-----	-----	-----

0 1 2 3 4

i
0
1
2
3
4
5

Exemplo – Voltando ao Estudo de Caso

```
double notas[] = new double [5];
int i;
double media, soma=0;
Scanner in = new Scanner(System.in);
for(i=0;i<5;i++){
    System.out.println("Informe a nota " + (i+1) + ": ");
    notas[i] = in.nextDouble();
    soma = soma + notas[i];
}
media = soma / 5;
System.out.println("A media do grupo eh: " + media);
System.out.println("A media do 3° aluno eh: " + notas[2]);
```

Declaração do vetor **media** e das demais variáveis do algoritmo!

Leitura dos dados no vetor e acumulação das médias lidas!

Impressão apenas da média do **terceiro** aluno!

Exemplo

■ Vetor de String

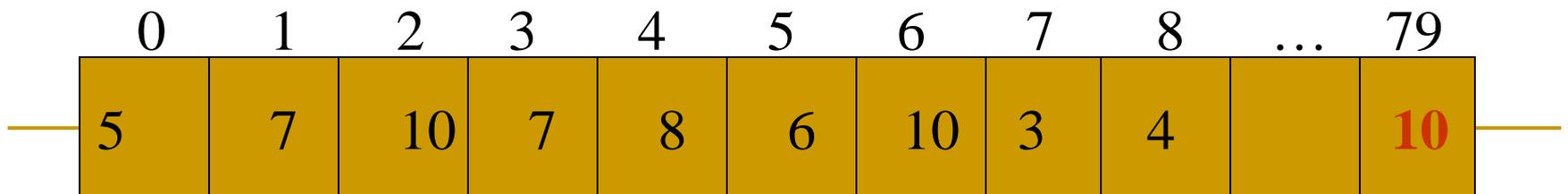
```
String nomes [ ] = new String [3] ;  
nomes[0] = in.nextLine();
```

■ Vetor de Inteiro

```
int vetor [ ] = new int [10] ;  
vetor[0] = in.nextInt();
```

Vetores

- Cada elemento de um vetor é tratado como se fosse uma variável simples. Para referência a um elemento do vetor utiliza-se o nome do vetor e o seu índice entre colchetes.
- Exemplo:
 - No vetor de 80 notas se eu quiser atribuir a nota 10 na posição 80 eu faço:
 - `notas[79] = 10`



Vetores

■ Exemplo 2

- Faça um programa que crie um vetor de string com 5 elementos e preencha cada elemento com uma cor de acordo como desenho abaixo. O programa deve pedir um número de 0 a 4 para o usuário, e imprimir a cor correspondente a esse número.

0	1	2	3	4
Azul	Rosa	Verde	Roxo	Preto

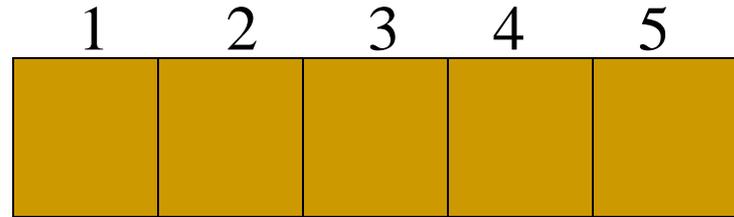
Vetores

0	1	2	3	4
Azul	Rosa	Verde	Roxo	Preto

Exemplo 2

```
String Cor[] = new String[5];
Scanner in = new Scanner(System.in);
Cor[0] = "Azul";
Cor[1] = "Rosa";
Cor[2] = "Verde";
Cor[3] = "Roxo";
Cor[4] = "Preto";
System.out.println("Digite um número de 0 a 4");
int num = in.nextInt();
System.out.println("A cor correspondente é
"+Cor[num]);
```

Vetores



■ Exemplo 2

- Refaça o exemplo 2 de maneira que o vetor seja preenchido utilizando os valores digitados pelo usuário.

Vetores

Exemplo 2

```
public static void main(String[] args) {  
  
    // TODO code application logic here  
    String Cor[] = new String[5];  
    Scanner in = new Scanner(System.in);  
    Scanner in2 = new Scanner(System.in);  
    for(int i=0;i<5;i++){  
        System.out.println("Digite uma cor");  
        Cor[i] = in.nextLine();  
    }  
    System.out.println("Digite um número de 0 a 4");  
    int num = in.nextInt();  
    System.out.println("A cor correspondente é "+Cor[num]);  
  
}
```

Vetores

- Observações importantes!
 - Imagine fazer o Estudo de Caso para 80 notas! Sem utilizar vetores. Se não fosse a utilização de vetores o código ficaria muito maior e difícil de manipular.
- ERROS comuns:
 - `int A = new int[5];`
 - `int B = new int[5];`
 - **Jamais** fazer: `A = B`. A atribuição de dois vetores tem que ser feito elemento por elemento. Ex: `A[1] := B[2]`
 - **Jamais** fazer: `A = in.nextInt();` Tem que ler elemento por elemento. Ex: `A[0] = in.nextInt();`
 - **Jamais** fazer: `System.out.println(A);` Tem que escrever elemento por elemento. Ex: `System.out.println(A[1])`

Exercícios

1. Leia 10 valores inteiros e armazene-os em um Vetor. Em seguida, imprima todos os elementos.
2. Leia 22 valores reais e armazene-os em um Vetor. Em seguida, imprima todos os elementos.
3. Leia 15 nomes de pessoas e armazene-os. Em seguida, mostre todos os nomes em sequência contrária.
4. Leia 8 elementos em um vetor inteiro A. Construa um outro vetor B, de mesma dimensão de A, com seus elementos sendo a multiplicação dos elementos de A por 3. Mostre os elementos de B.

Exercícios

5. Leia dois vetores A e B inteiros, de mesma dimensão. Construa o vetor C, de mesma dimensão, cujo elementos de C é a subtração dos elementos de A por B. Mostre o vetor C.
6. Leia dois vetores reais A (de tamanho 4) e B (de tamanho 7). Construa um 3º vetor que seja a **junção** de A e B (colocar os elementos de A seguidos dos elementos de B).
7. Leia um vetor e ache o maior elemento.
8. Leia um vetor e ache o menor elemento.

Exercícios

9. Leia um vetor de 15 elementos inteiros e procure por um elemento desejado, verificando ou não sua existência.
10. Leia 20 elementos em um vetor A e construa o vetor B com elementos de A, porém invertidos, ou seja, o primeiro elemento de A passa a ser o último elemento de B, e assim por diante. Mostre os 2 vetores.
11. Leia um vetor de 9 elementos. Imprima esse vetor na ordem da entrada e ao inverso.
12. Leia um vetor de 9 elementos. Imprima os elementos do vetor identificando como pares ou ímpares.

Exercícios

■ Desafio 3

- Construir um algoritmo que leia em um vetor os modelos de cinco carros (exemplo: FUSCA, GOL, VECTRA, etc). Leia outro vetor com o consumo desses carros, isto é, quantos quilômetros cada um desses carros faz com 1 litro de combustível. Calcule e mostre:
 - O modelo do carro mais econômico.
 - Quantos litros de combustível cada um dos carros cadastrados consome para percorrer uma distância de 1000 quilômetros.