

Vetores

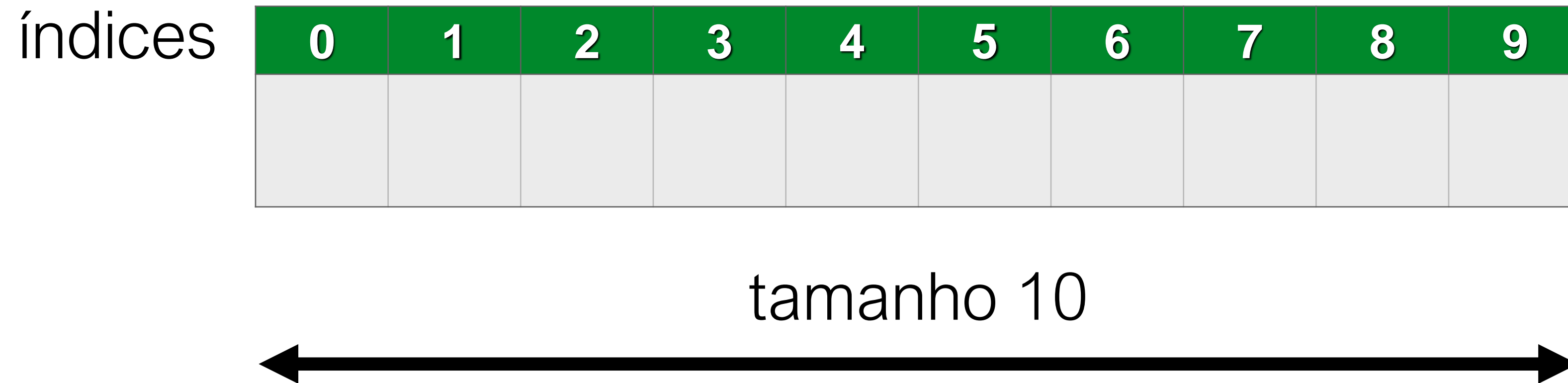
Programação Orientada a Objetos - Aula 11

Professor: Hamilton Machiti da Costa

Vetores

- Vetores são estruturas de dados homogêneas de acesso indexado.
Traduzindo:
 - é um conjunto de variáveis (estrutura de dados)
 - que são todas do mesmo tipo (homogêneas)
 - que são acessadas por sua posição (acesso indexado)
 - a primeira posição é 0
 - a última posição é (tamanho do vetor - 1)

Representação gráfica de um vetor



Declaração

```
//cria o vetor vetor de 10 posições contendo um inteiro em cada posição  
//a primeira é 0 e a última é 9
```

```
private int [] vetor = new int[10];
```

```
//cria o vetor livre de 5 posições contendo um true ou false em cada uma  
boolean livre[] = new boolean[5];
```

```
//cria um vetor de String de duas posições chamado nomes  
String[] nomes;  
nomes = new String[2];
```

Atribuição

- Para atribuir valores a um vetor você faz a atribuição na posição desejada:

```
vetor[0] = 1;  
vetor[9] = 10;  
livre[3] = false;
```

Atribuição

- Você também pode fazer a atribuição usando uma lista de literais na declaração:

```
int array[] = {10, 20, 30};
```

```
String[] nomes = {"João", "Maria"};
```

Leitura

- Para ler os valores, você acessa diretamente a posição desejada:

```
int x = vetor[7];  
  
if(livre[2]){  
    System.out.println(nomes[2]);  
}
```

“Andando” em um vetor: atribuição

- Uma combinação perfeita são os laços for e os vetores. Existe uma propriedade dos vetores, **length**, que indica o tamanho do vetor. Por exemplo, para preencher um vetor com valores digitados pelo usuário:

```
for(int i = 0; i < vetor.length; i++){  
    vetor[i] = Integer.parseInt(  
        JOptionPane.showInputDialog("Digite o valor"));  
}
```


“Andando” em um vetor: leitura

- Para ler é parecido:

```
for(int i = 0; i < vetor.length; i++){  
    System.out.println( (vetor[i]) );  
}
```

Busca sequencial

- Como encontrar “algo” em um vetor?
- Ande o vetor inteiro e pergunte, em cada posição, se o “algo” que você tem na mão, isto é, a chave de busca, é igual ao elemento que está naquela posição.
- Se for, retorne o índice da posição em que está o elemento.
- No final, se não encontrou depois de ter rodado o vetor inteiro, retorne -1.

Busca sequencial

```
public class Exemplo{
    int[] vetor;

    public Exemplo(int[] v){
        vetor = v;
    }
    public int busca(int chave){
        for(int i = 0; i < vetor.length; i++){
            if (vetor[i] == chave){
                return i;
            }
        }
        return -1;
    }
}
```

Vetores bidimensionais

- Um vetor de duas dimensões é um vetor que contém dois índices. O primeiro faz referência a um outro vetor. O segundo faz referência a um elemento deste outro vetor. Como se fosse uma matriz da matemática:

```
//instanciação
double[][] matriz = new double[10][5];

//atribuição a uma posição
matriz[3][2] = 10.0;

//atribuição a todas as posições
for(int i = 0; i < matriz.length; i++){
    for(int j = 0; j < matriz[i].length; j++){
        matriz[i][j] = Math.random();
    }
}

//leitura
double x = matriz[0][0];
```

Vetores multidimensionais

- Estendendo-se o conceito, você pode criar estruturas com n dimensões em Java. Para criar um cubo, por exemplo, que é uma matriz com um vetor em cada posição, faz-se:

```
String[][][] cubo = new String[10][10][10];
```

Tipos & Usos

- Assim como em qualquer variável em Java, um vetor pode ser de um tipo primitivo ou de um tipo objeto. E pode assumir o papel de variável local, variável de instância, parâmetro de método ou tipo de retorno de método.

```
public Aluno[] getTurma(Aluno[][][] campus, int ano, int curso){  
    Aluno[] alunos = campus[curso][ano];  
    return alunos;  
}
```

```
private Professor[] docentes;
```