Colégio ULBRA São Lucas
27ª Coordenadoria Regional de Educação – Deliberação nº 451/2016 CEEd/RS
ASSOCIAÇÃO EDUCACIONAL LUTERANA DO BRASIL

APOSTILA 4

Vetores[]

Créditos:

Autor: Klint.
Revisão Técnica: Prof. Jeferson Leon e Prof. Éder Oliveira de Rosso.
Curso de Java.
VC Ensinos.

SUMÁRIO

APRESENTAÇÃO DOS VETORES

Apresentação dos Vetores: O que são Vetores e Primeiro Exemplo

O que são vetores

O vetor é uma estrutura de dados que armazena uma coleção ordenada de elementos do mesmo tipo. Também é conhecido como array. Os elementos de um vetor podem ser acessados por meio de um índice que representa sua posição dentro do vetor. Em Java, os vetores têm um tamanho fixo após a sua criação e cada elemento é acessado usando sua posição no vetor, que começa do índice 0 até o tamanho do vetor menos um.

Primeiro Exemplo

Por exemplo, o seguinte trecho de código cria um vetor de inteiros em Java:

int[] vetor = new int[5];

Este código cria um vetor de inteiros chamado "vetor" com tamanho 5. Os elementos deste vetor podem ser acessados utilizando índices que variam de 0 a 4. Por exemplo, vetor[0] acessa o primeiro elemento, vetor[1] o segundo e assim por diante.

Inicialização direta

int[] vetor = {10, 20, 30, 40, 50};

- Esse método é simples e direto. Você especifica os valores entre chaves {} na declaração do vetor. É útil quando os valores são conhecidos antecipadamente e não precisam ser calculados ou processados.

Inicialização com valores específicos

int[] vetor = new int[]{1, 3, 5, 7, 9};

- Esta abordagem é semelhante à inicialização direta, mas permite que você declare o tamanho do vetor separadamente, se necessário. É útil quando você sabe os valores, mas deseja flexibilidade na declaração do tamanho do vetor, ou seja, é quando adicionamos apenas os primeiros valores, mas podemos adicionar um tamanho maior em seguida.

Declaração e Inicialização de Vetores

Inteiros

// Declaração e inicialização de um vetor de inteiros com tamanho 5

```
int[] vetor = new int[5];

// Atribuindo valores aos elementos do vetor
vetor[0] = 10;
vetor[1] = 20;
vetor[2] = 30;
vetor[3] = 40;
vetor[4] = 50;

// Acessando e imprimindo elementos do vetor
System.out.println("Elemento no índice 0: " + vetor[0]);  // Saída: Elemento no índice 0: 10
System.out.println("Elemento no índice 2: " + vetor[2]);  // Saída: Elemento no índice 2: 30
```

Strings

```
// Declaração e inicialização de um vetor de strings
String[] nomes = {"Aline", "Barbara", "Carlos", "Daniel"};

// Acessando e imprimindo elementos do vetor
System.out.println("Primeiro nome: " + nomes[0]);  // Saída: Primeiro nome: Aline
System.out.println("Último nome: " + nomes[nomes.length - 1]);  // Saída: Último nome: Daniel
```

Métodos Comuns

Existem alguns métodos para vetores que nos facilitam na hora de montar o código.

length

Retorna o tamanho do vetor (número de elementos).

```
int[] vetor = {1, 2, 3, 4, 5};
int tamanho = vetor.length; // tamanho será 5
```

clone

Cria uma cópia superficial do vetor.

```
int[] vetor = {1, 2, 3, 4, 5};
int[] copia = vetor.clone(); // copia contém os mesmos elementos que vetor
```

equals

Verifica se dois vetores são iguais (mesmos elementos na mesma ordem).

```
int[] vetor1 = {1, 2, 3};
int[] vetor2 = {1, 2, 3};
boolean iguais = Arrays.equals(vetor1, vetor2); // iguais será true
```

fill

Preenche todos os elementos do vetor com um valor específico.

```
int[] vetor = new int[5];
Arrays.fill(vetor, 10); // preenche todos os elementos de vetor com o valor 10
```

sort

Ordena os elementos do vetor em ordem crescente.

int[] vetor = {5, 3, 1, 4, 2};
Arrays.sort(vetor); // vetor será {1, 2, 3, 4, 5}

binarySearch

Realiza uma busca binária por um elemento em um vetor ordenado e retorna o índice do elemento se encontrado, ou um valor negativo se não encontrado.

int[] vetor = {1, 2, 3, 4, 5};
int indice = Arrays.binarySearch(vetor, 3); // retorna o índice 2 pois 3 está no índice 2 do vetor

toString

Converte um vetor em uma string representando seus elementos.

int[] vetor = {1, 2, 3, 4, 5};
String vetorString = Arrays.toString(vetor); // vetorString será "[1, 2, 3, 4, 5]"

- Esses são apenas alguns dos métodos mais comuns que podem ser usados com vetores em Java. Existem outros métodos disponíveis, dependendo das classes e bibliotecas que você está usando, mas esses são os mais utilizados na manipulação básica de vetores.

Desafios

1- O que é um vetor?

2- Seguindo o vetor (int[] vetor = {6, 3, 9, 15, 12, *};) resolva as seguintes questões:

a) Qual a posição do 3º elemento do vetor?
b) Qual elemento está na posição 5?

3- Crie e preencha um vetor, depois peça para o programa ordenar os elementos em ordem crescente. Assim que ordenado, imprima o vetor.

4- Crie dois vetores e peça para o usuário preencher eles, depois peça para o programa verificar se os dois vetores são iguais, se a resposta for verdadeira, imprima a seguinte frase na tela: "os dois vetores são iguais, nenhum elemento é diferente" e imprima cada um deles em forma de string, e imprimindo o seu maior número de forma destacada, se a resposta for falsa, imprima a seguinte frase na tela: "algum valor é diferente, talvez um elemento, ou todos são diferentes, verifique os vetores"

Relembrando

Um vetor pode ser conhecido como **Array.**

O código para criação de um vetor pode ser: **int[] vetor = new int[5];** ou **int[] vetor = new int[]{1, 3, 5, 7, 9};.**

Podemos usar um looping **for** para preencher um vetor, e podemos usar o looping **for** junto com a função **Scanner** para que o usuário digite os elementos do vetor.

Podemos usar métodos para facilitar e podemos utilizar mesmo que não estejam presentes na pergunta. E os métodos apresentados, são os mais usados e os mais básicos que podemos usar.

Temos os métodos **length**, **clone**, **equals**, **fill**, **sort**, **binarySearch**, **toString**, entre outros não apresentados.

Referências bibliográficas: BLOG. Rocketseat. Vetores em Java: o que você precisa saber. Disponível em: https://blog.rocketseat.com.br/vetores-em-java-o-que-voce-precisa-saber/. Acesso em: 2 dez. 2024.

DEVMEDIA. Vetores em Java. Disponível em: https://www.devmedia.com.br/vetores-em-java/21449. Acesso em: 4 dez. 2024.

BLOG. Rocketseat. Java: um guia sobre vetores. Disponível em: https://blog.rocketseat.com.br/java-um-guia-sobre-vetores/. Acesso em: 4 dez. 2024