Aula 8 — Vetores

Processamento da Informação

Universidade Federal do ABC

Vetores (= *arrays* em Java)

Como armazenar 3 notas?

```
...
float nota1, nota2, nota3;
System.out.println("Digite a nota do aluno 1: ");
nota1 = scanner.nextFloat();
System.out.println("Digite a nota do aluno 2: ");
nota2 = scanner.nextFloat();
System.out.println("Digite a nota do aluno 3: ");
nota3 = scanner.nextFloat();
...
```

Vetores

Como armazenar 100 notas?

```
...
float nota1, nota2, nota3, /*...*/ nota100;
System.out.println("Digite a nota do aluno 1: ");
nota1 = scanner.nextFloat();
System.out.println("Digite a nota do aluno 2: ");
nota2 = scanner.nextFloat();

/*...*/

System.out.println("Digite a nota do aluno 100: ");
nota100 = scanner.nextFloat();
...
```

Vetores - Definição

- Coleção de variáveis do mesmo tipo referenciada por um nome comum (Herbert Schildt).
- Características:
  - Acesso por meio de um índice inteiro.
    - O uso de índices fora dos limites causa um erro durante a execução do programa.
  - Posições contíguas na memória.
  - Tamanho pré-definido.

Vetores - Declaração

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

Exemplo:

```
float[] notas = new float[100];
int[] primos = new int[30];
```

- O acesso aos elementos do vetor pode ser feito utilizando **[i]**, onde **i** é a posição desejada.
- O uso do valor de um vetor **v** no índice **i** pode ser feito **exatamente como uma variável**.
- ***ATENÇÃO*** o **primeiro elemento fica na posição 0**! Se o vetor tem n elementos, **o último fica na posição n - 1**

```
notas[13] = 9.2;
System.out.println(notas[14]);
notas[10] = notas[9] + 2;
```

Vetores - Utilização

Nosso exemplo inicial para guardar 100 notas poderia ser escrito assim:

```
...
float[] notas = new float[100];
for (int i = 0; i < 100; i++) {
    System.out.printf("Digite a nota do aluno %d: ", i);
    notas[i] = scanner.nextFloat();
}
...
```

- Note que a primeira linha impressa será *"Digite a nota do aluno 0:"*.

Vetores - Definindo o tamanho programaticamente

Suponha que não sabemos quantos alunos teremos. Podemos alterar o nosso código para perguntar ao usuário e só então criar o vetor.

```
...
int tamanho = scanner.nextInt();
float[] notas = new float[tamanho];
for (int i = 0; i < tamanho; i++) {
    System.out.printf("Digite a nota do aluno %d: ", i);
    notas[i] = scanner.nextFloat();
}
...
```

Obs.: O tamanho do vetor pode ser obtido da seguinte maneira:

```
int num_elems = notas.length;
```

Ex. 1 - Produto interno de dois vetores

- Leia n, inteiro, e dois vetores (float) de dimensão n e compute o seu produto interno com uma casa decimal.
- Lembrete: sejam $x = \langle x_1, x_2, \ldots x_n \rangle$ e $y = \langle y_1, y_2, \ldots y_n \rangle$ vetores de dimensão n. Então o produto interno $x \cdot y = x_1 y1 + x_2 y2 + \ldots + x_n y_n$

Entrada	Saída
2 7 42 11 3	203.0
3 2 3.5 7.9 1 2.8 4	43.4

Ex.2 - Elementos iguais

Leia dois vetores com 5 inteiros cada. Verifique quais elementos do segundo vetor estão contidos no primeiro. Se não houver elementos em comum, o programa deve informar.

Entrada	Saída
42 37 28 5 3 58 71 5 42 69	5 42
1 2 3 4 5 6 7 8 9 10	Nenhum elemento comum

Ex.2 - Produtos de pares

Crie uma função com a assinatura

```
public static boolean checa(int[] vet, int c)
```

Ela deve receber como entrada um vetor e um inteiro *c*. A função deve retornar **true** caso existam dois elementos distintos do vetor tal que a multiplicação destes seja c e **false** caso contrário.

Exemplo: Se vet = (2, 4, 5, -10, 7) e c = 35, então a função deve devolver true. Mas se c = -1, então a função deve devolver false.

Lista de exercícios (Entrega via URI)

- 1172
- 1173
- 1174
- 1175
- 1177
- 1179
- 1180
- 1973
- 2222
- 2310
- 1104