



Prof. Dr. Fernando Almeida
proffernando.almeida@fiap.com.br





DDD (Domain Driven Design)

Arrays Bidimensionais - Matrizes



O QUE VAMOS APRENDER HOJE?

1

Matrizes - Definição

2

Declaração de Matrizes

3

Inicialização de Matrizes

4

Representação de Matrizes em memória

5

Matrizes com métodos



O que são Matrizes?

Definição



“**Matrizes** (array bidimensional) são estruturas **indexadas** e **homogêneas**, também conhecidas como **vetor de vetores**, onde cada valor pode ser armazenado em uma posição (índice), denominado elemento do array.



• • • • •

+

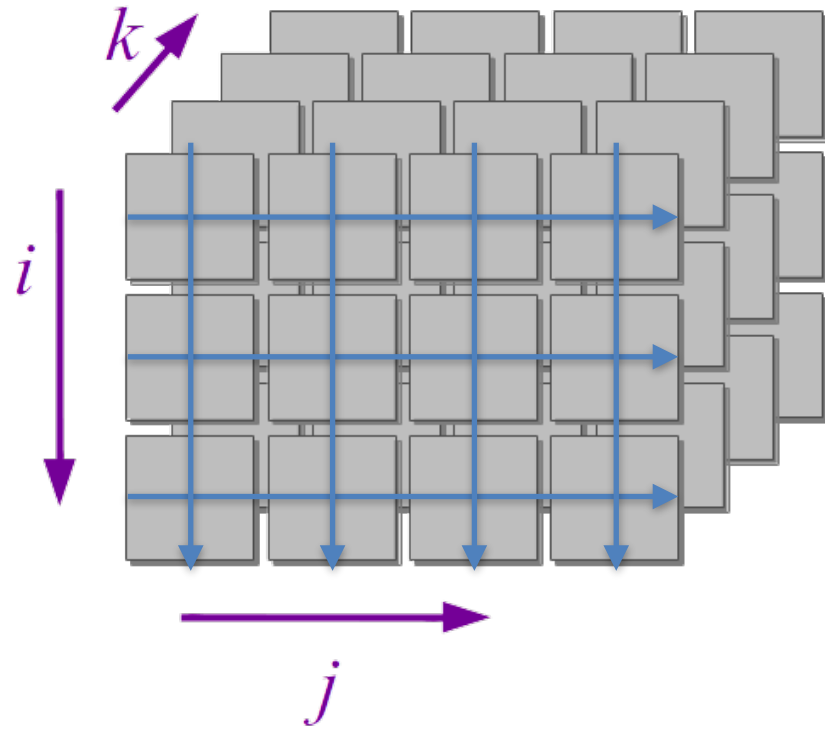
• Representação de um vetor bidimensional - Matriz

• + •

+ •

|

+



•

•

+

■

□

•

•

•

•

•

+

•

•

•

•

•

Matrizes

```
int a;  
int v[] = new int[5];  
int m[][] = new [linhas][colunas];
```

- Para criar um array bidimensional com 3 linhas e 5 colunas, faremos:
- `int[][] minhaMatriz = new int[3][5];`
- Representação em memória:



Declaração de um vetor bidimensional para inteiros

- Semelhante a declaração de variáveis
- Sintaticamente diferenciados pelo uso de **dois colchetes**

```
1 | int m[][] = new int[2][4]; // matriz com 2 linhas X 4 colunas
```

colunas
linhas

operador **new** (palavra reservada da linguagem) para alocar memória, neste exemplo, aloca memória para elementos do tipo **int**

Representação interna:

m[0][0]	m[0][1]	m[0][2]	m[0][3]
m[1][0]	m[1][1]	m[1][2]	m[1][3]

Referência a um elemento do vetor bidimensional

m[0][2] = 0;	// elemento da primeira linha, terceira coluna
	// ou ainda, terceiro elemento do primeiro vetor
m[i][j]	// elemento da i-ésima linha, j-ésima coluna

Declarando arrays bidimensionais

- Com expressões de criação de vetores:

```
1 | int m[][] = new int[3][3]; // matriz quadrada: 3 linhas X 3 colunas
```

- Atribuição de valores aos elementos do vetor

```
m[0][0] = 1;  
m[0][1] = 2;  
m[0][2] = 3;  
m[1][0] = 4;  
m[1][1] = 5;  
m[1][2] = 6;  
m[2][0] = 7;  
m[2][1] = 8;  
m[2][2] = 9;
```

Declarando e inicializando (diretamente) o vetor

```
1 | int m[3][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} };
```

equivale as seguintes atribuições:

```
1 | // 1ª linha:
2 | m[0][0] = 1;
3 | m[0][1] = 2;
4 | m[0][2] = 3;
5 |
6 | // 2ª linha:
7 | m[1][0] = 4;
8 | m[1][1] = 5;
9 | m[1][2] = 6;
10 |
11 | // 3ª linha:
12 | m[2][0] = 7;
13 | m[2][1] = 8;
14 | m[2][2] = 9;
```

Criando uma matriz com diferentes tamanhos

```
1 | int m[][] = new int[2][]; // cria 2 linhas
2 |
3 | m[0] = new int[5]; // cria 5 colunas para a linha 0
4 | m[1] = new int[3]; // cria 3 colunas para a linha 1
```

- Declaração e inicialização direta de linhas com diferentes tamanhos

```
1 | int m[][] = { {1, 2}, {4, 5, 6, 7, 8}, {9, 10, 11} };
```

Propriedade Length

- Utilizado para conhecer o tamanho da matriz (elementos em cada linha)
- **m.length** - retorna o número de linhas
- **m[i].length** - retorna o número de colunas da i-ésima linha

```
1 package aula15_Matrizes;
2
3 public class Exemplo1 {
4     public static void main(String[] args) {
5
6         int m[][] = {{1,2}, {3}, {4,5,6}};
7
8         //i representa o número de linhas
9         for(int i=0; i<m.length; i++) {
10             System.out.printf("%da. linha: \n", i+1);
11             //j representa o número de colunas
12             for(int j=0; j<m[i].length; j++) {
13                 System.out.printf("matriz[%d][%d]: %d \n", i,j, m[i][j]);
14             }
15         }
16     }
17 }
```

saída:

```
1a. linha:
matriz[0][0]: 1
matriz[0][1]: 2
2a. linha:
matriz[1][0]: 3
3a. linha:
matriz[2][0]: 4
matriz[2][1]: 5
matriz[2][2]: 6
```



Exemplo de Aplicação Java

Arrays bidimensionais



Exemplo 1

```
1 import java.util.Scanner;
2
3 public class Matriz1 {
4     public static void main(String[] args) {
5         Scanner ler = new Scanner(System.in);
6         int i, j, m[][] = new int[2][4];
7
8         for (i=0; i<2; i++) {
9             System.out.printf("Informe os elementos %da. linha:\n", (i+1));
10             for (j=0; j<4; j++) {
11                 System.out.printf("m[%d][%d] = ", i, j);
12                 m[i][j] = ler.nextInt();
13             }
14             System.out.printf("\n");
15         }
16
17         System.out.printf("\n");
18         for (i=0; i<2; i++) {
19             System.out.printf("%da. linha: ", (i+1));
20             for (j=0; j<4; j++) {
21                 System.out.printf("%d ", m[i][j]);
22             }
23             System.out.printf("\n");
24         }
25     }
26 }
```



Exercícios

(sem métodos)



Exercícios

1. Escreva um programa que crie uma matriz de tamanho 3x3 e imprima os elementos pares da matriz. Adicione um contador para armazenar o número de elementos pares presentes na matriz.
2. Escreva um programa para somar todos os elementos de uma matriz nem e imprima a média dos elementos.
3. Um sistema meteorológico deseja armazenar as médias de temperaturas de cada dia em um determinado mês. Escreva um programa em Java que seja capaz de armazenar de criar esse controle e imprimir qual foi a temperatura mais alta, a mais baixa e a média das temperaturas desse mês.



Matrizes com Métodos

Exemplo



Exemplos

```
public int[][] criaMatriz(int[] v){
    System.out.println("*- criaMatriz -*");
    int[][] matriz = new int[v[0]][v[1]];
    for(int i=0; i<matriz.length;i++){
        for(int j=0; j<matriz[i].length; j++){
            System.out.printf("matriz[%d][%d]: \n", i,j);
            matriz[i][j] = input.nextInt();
        }
    }
    return matriz;
}
```

Exemplo 1: retorno de matriz

```
public int verificaOcorrências(int[][] m, int n){
    System.out.println("*-Verifica Ocorrências-*");
    int cont=0;
    for(int i = 0; i < m.length; i++) {
        for (int j = 0; j < m[i].length; j++) {
            if(m[i][j] == n){
                cont++;
            }
        }
    }
    return cont;
}
```

Exemplo 2: Passagem por parâmetro



Vamos testar no Eclipse?

Exemplo Prático





OBRIGADO

FIAP

Copyright © 2021 | Prof. Dr. Fernando Luiz de Almeida

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.



