

Laboratório de Programação

Array Bidimensional (matriz)

Prof. Dr. Antonio Marcos SELMINI
selmini@fiap.com.br



Exemplo de array bidimensional

	colunas			
linhas	2	10	5	0
	1	256	33	85
	9	66	65	63
	69	63	58	70

Array bidimensional (matriz)

As *matrizes* são utilizadas na computação para armazenar tabelas de elementos

Dentre as principais aplicações de matrizes podemos destacar:

- ❑ Programação de jogos computacionais: jogo da velha, batalha naval, jogos simulando bingos, dentre muitos outros;
- ❑ Representação computacional de imagens para processamento;
- ❑ Equacionamento e resolução de problemas de análise numérica aplicados à engenharia, dentre outros;

Array bidimensional (matriz)

Para declararmos um **array bidimensional** devemos especificar o número de linhas e o número de colunas presentes

Sintaxe **completa** para declaração de uma matriz:

```
tipo[][] nome_array;  
nome_array = new tipo[total_de_linhas][total_de_colunas];
```

tipo pode ser qualquer tipo válido em Java. O primeiro índice representa o número total de linhas do array e o segundo índice o número total de colunas

Array bidimensional (matriz)

Exemplo de declaração:

```
int[][] x = new int[5][5];  
double y[][] = new double[3][4];
```

Como os vetores, as matrizes podem ser declaradas e inicializadas:

```
int b[][] = {{1, 2}, {3, 4}};
```

Array bidimensional (matriz)

É possível também criar um *array multidimensional* em que cada linha tem um número diferente de colunas. Exemplo:

```
int b[][];  
b = new int[2][]; //cria apenas duas linhas  
b[0] = new int[5]; //cria 5 colunas para a linha 0  
b[1] = new int[3]; //cria 3 colunas para a linha 1
```


























Propriedades dos arrays bidimensionais

Da matemática temos algumas definições básicas quando trabalhamos com matrizes:

- ❑ Uma matriz é dita **quadrada** quando o número de linhas m é igual ao número de colunas n , ou seja, $m = n$;
- ❑ Por exemplo, uma matriz que apresenta 5 linhas e 5 colunas é uma matriz quadrada, pois o número de linhas é igual ao número de colunas;
- ❑ Neste caso dizemos que a matriz tem ordem 5. Quando uma matriz é dita quadrada outras definições aparecem.

Propriedades dos arrays bidimensionais

Veja na figura abaixo, que representa uma matriz quadrada de ordem 5:

	0	1	2	3	4
0					
1					
2					
3					
4					

Diagonal Secundária

Diagonal Principal

Exemplo 1

O programa abaixo declara e preenche um array bidimensional com valores aleatórios entre 0 e 20.

```
import java.util.Random;
public class Exemplo1 {
    public static void main(String[] args) {
        Random gerador = new Random();
        int[][] x = new int[2][2];
        int linha, coluna;

        for(linha = 0; linha < x.length; linha++) {
            for(coluna = 0; coluna < x[linha].length; coluna++) {
                x[linha][coluna] = gerador.nextInt(21);
            }
        }
    }
}
```

Exemplo 2

Neste exemplo, o array que foi preenchido no exemplo 1 é impresso no vídeo:

```
import java.util.Random;

public class Exemplo2 {
    public static void main(String[] args) {
        Random gerador = new Random();
        int[][] x = new int[2][2];
        int linha, coluna;

        for(linha = 0; linha < x.length; linha++) {
            for(coluna = 0; coluna < x[linha].length; coluna++) {
                System.out.print(x[linha][coluna]+"\\t");
            }
            System.out.println();
        }
    }
}
```

Exemplo 3

No primeiro exemplo, o array é preenchido com valores aleatórios. Podemos também preencher o array com valores informados via teclado. Veja o exemplo abaixo:

```
import java.util.Scanner;
public class Exemplo3 {
    public static void main(String[] args) {

        Scanner entrada = new Scanner(System.in);
        int[][] x = new int[2][2];
        int linha, coluna;

        for(linha = 0; linha < x.length; linha++)
            for(coluna = 0; coluna < x[linha].length; coluna++) {
                System.out.println("Digite um valor: ");
                x[linha][coluna] = entrada.nextInt();
            }
    }
}
```

Exercícios de programação

1. Escreva um programa em Java que preencha uma matriz quadrada de ordem 4 com valores inteiros aleatórios. Em seguida:
 - a) Mostre a soma dos elementos da diagonal principal.
 - b) Mostre a soma dos elementos da diagonal secundária.
2. Escreva um programa em Java que preencha uma matriz $D_{10 \times 10}$ com valores aleatórios entre 0 e 1000. Imprima o maior valor armazenado bem como a sua localização (linha e coluna). Observação: caso o maior valor se repetida dentro da matriz, todas as localizações deverão ser impressas.

Exercícios de programação

3. Análise de Temperaturas Médias Anuais: Você foi contratado para realizar a análise das temperaturas médias anuais de uma cidade ao longo dos últimos 10 anos. Os dados são fornecidos em um formato de matriz (array bidimensional), onde cada linha representa um ano e cada coluna representa um mês (de janeiro a dezembro). **Instruções:**

- a) Crie uma matriz de 10x12 contendo as temperaturas médias mensais de cada ano (valores fictícios podem ser utilizados para testar o código).
- b) Calcule a média de cada linha (ano).
- c) Determine o ano com a maior média e o ano com a menor média.

Exercícios de programação

4. Análise de Vendas de Produtos: Uma loja online vende três categorias de produtos: Eletrônicos, Roupas, e Alimentos. As vendas desses produtos foram registradas ao longo dos últimos 6 meses. Você deve analisar essas vendas para obter insights sobre o desempenho de cada categoria. **Instruções:**

- a) Ler os valores fictícios das vendas.
- b) Calcular o total de vendas por categoria.
- c) Identificar o mês com o maior número de vendas totais.
- d) Calcular a média de vendas mensais para cada categoria.

Exercícios de programação

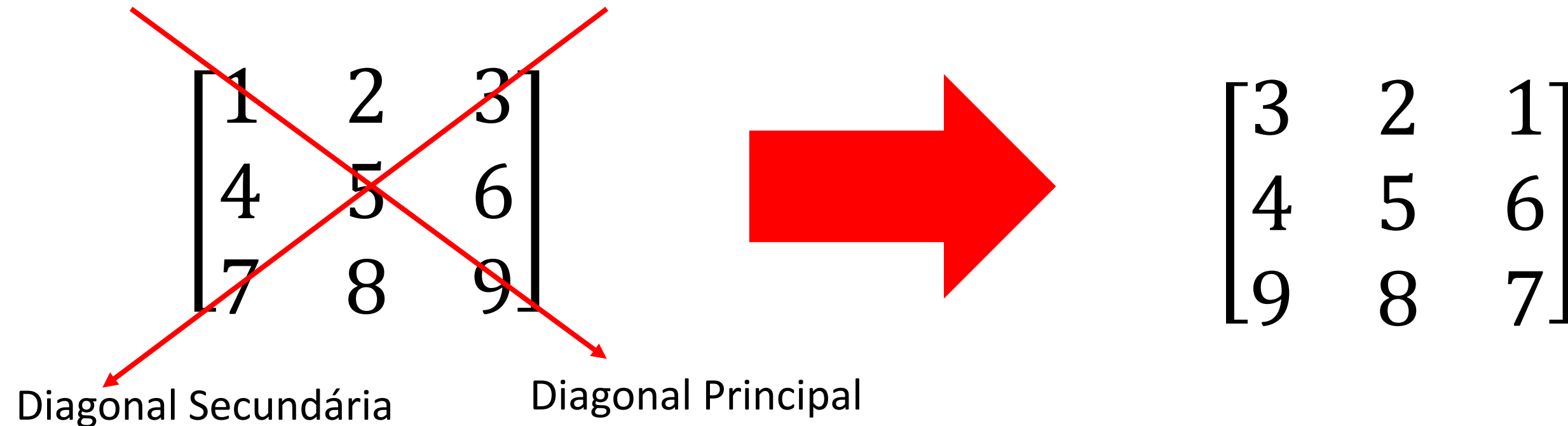
5. Faça um programa em Java que leia uma matriz $C_{2 \times 3}$. Gere e imprima a matriz transposta C^t . A matriz transposta é gerada trocando os elementos da linha pela coluna. Veja o exemplo a seguir:

$$C = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \quad \longrightarrow \quad C^t = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$$

6. Dadas as matrizes $A_{3 \times 2}$ e $B_{2 \times 3}$, gere $C = A * B$. Imprima no vídeo as matrizes A, B e C.

Exercícios de programação

7. Escreva um programa em Java que preencha um matriz quadrada de ordem 5 com valores aleatórios. Em seguida, troque os elementos da diagonal principal com os elementos da diagonal secundária. Como exemplo, considere a matriz quadrada de ordem 3.



Exercício Extra (não será feito na sala)

Tendo como base a matriz A, quais são os valores armazenados na matriz B após a execução de cada instrução abaixo?

$$A = \begin{bmatrix} 5 & -2 & 0 \\ 1 & 7 & 3 \\ 4 & 5 & -1 \end{bmatrix} \quad B = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}$$

- a) $B[0][0] = A[0][0] + A[2][2];$
- b) $B[0][1] = B[0][0] + A[0][1];$
- c) $B[0][2] = A[2][0] + 2 * A[2][2];$
- d) $B[1][0] = B[0][1];$
- e) $B[1][1] = B[0][0] + A[0][0] + A[1][1];$
- f) $B[1][2] = A[2][1]/2;$

Exercício Extra (não será feito na sala)

Considere uma matriz B , quadrada de ordem 3 inicialmente vazia. O programa abaixo irá preenchê-la. Após a execução do programa, desenhe a matriz com os elementos.

```
public class Exercicio {  
    public static void main(String[] args) {  
  
        int[][] b = new int[3][3];  
        int i, j;  
  
        for(j = 0; j < b.length; j++) {  
            for(i = 0; i < b.length; i++) {  
                b[i][j] = 2*i*j;  
            }  
        }  
    }  
}
```

Exercício Extra (não será feito na sala)

Considere que a matriz A seja processada pelo programa abaixo. Após o processamento, quais os novos valores da matriz?

```
public class Exercicio {  
    public static void main(String[] args) {  
        int[][] A = new int[3][3];  
        int i, j;  
        for(i = 0; i < A.length; i++) {  
            for(j = 0; j < A[i].length; j++) {  
                if(A[i][j] % 2 == 0) {  
                    A[i][j] = A[i][j] * 3;  
                }  
            }  
        }  
    }  
}
```

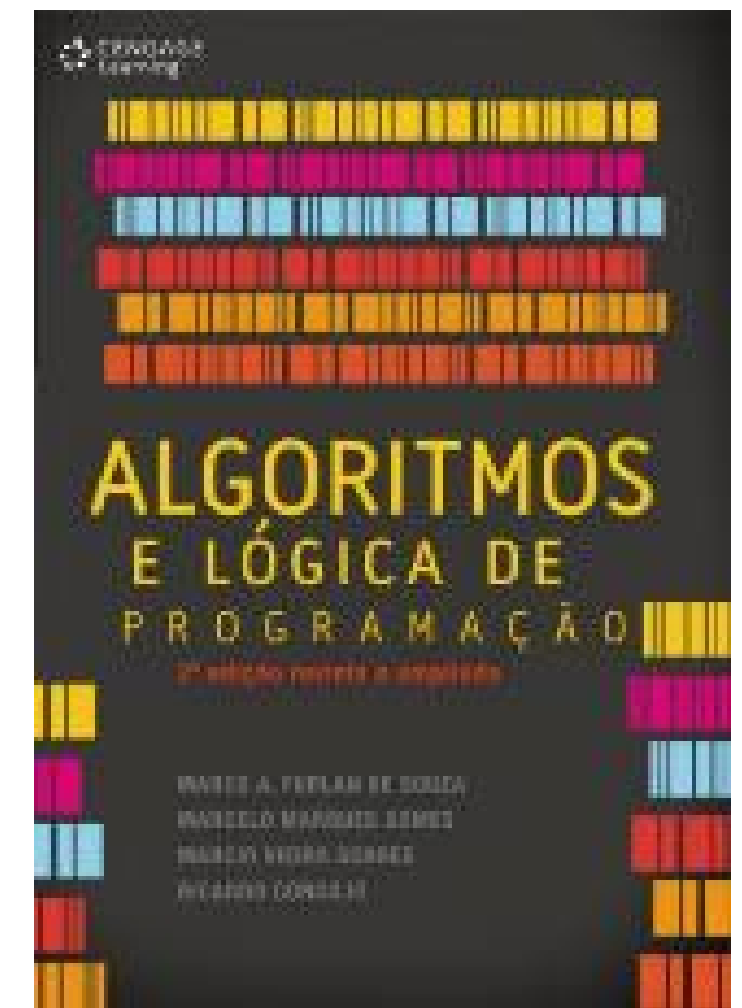
$$A = \begin{bmatrix} 3 & 1 & 5 \\ 2 & 4 & 6 \\ -2 & 0 & 7 \end{bmatrix}$$

Bibliografia



XAVIER, G. F. C. **Lógica de Programação**. 12ª ed. Editora Senac
São Paulo, 2011.

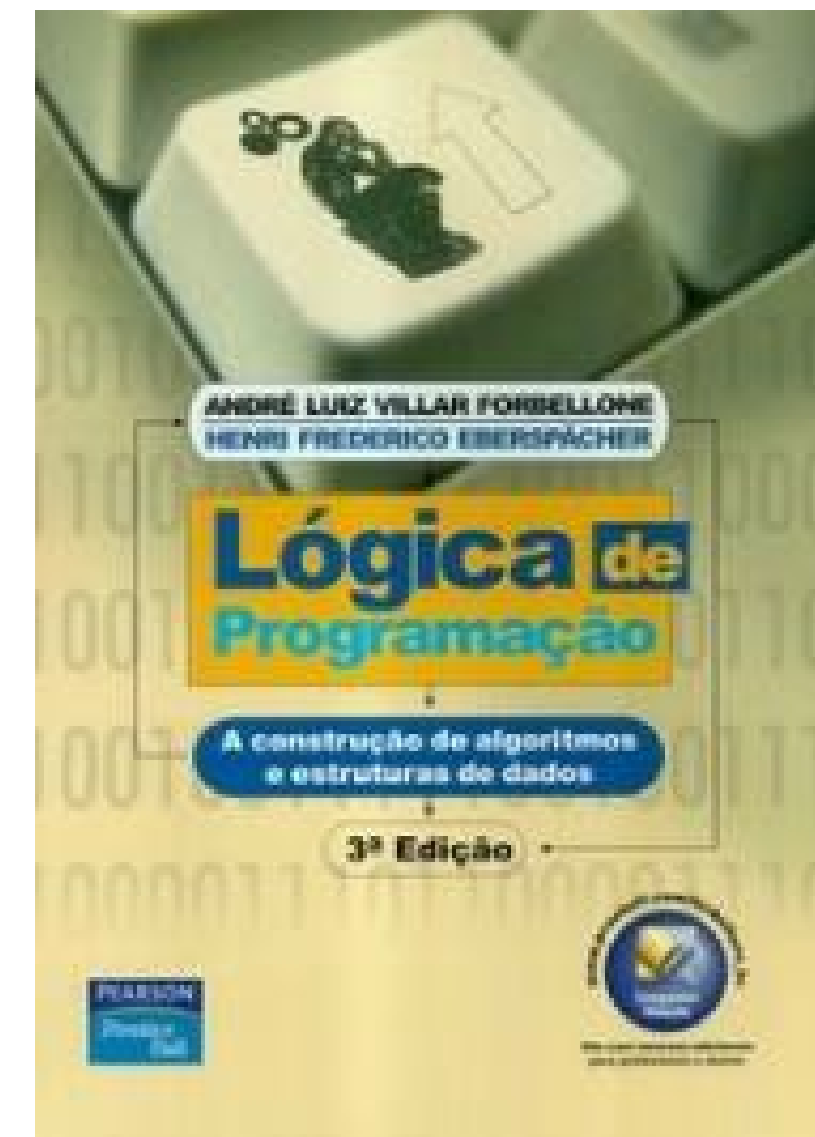
SOUZA, M. A. F.; SOARES, M. V.; GOMES, M. M.; CONCILIO,
R. **Algoritmos e Lógica de Programação**. 1ª e 2ª ed. São
Paulo: Cengage Learning, 2011.



Bibliografia



PUGA, S., RISSETTI, G. Lógica de programação e estrutura de dados com aplicações em Java. 2ª ed. São Paulo: Editora Pearson, 2013.



FORBELLONE, A. L., EBERSPACHER, H. F. Lógica de programação - A construção de algoritmos e estruturas de dados. 3ª ed. São Paulo: Editora Pearson, 2005.

Bibliografia



GUIMARÃES, A. de M. Algoritmos e estruturas de dados. Rio de Janeiro: Editora LTC, 1994.

ASCENCIO, A. F. G.; CAMPOS, E. A. V. Fundamentos da Programação de Computadores – Algoritmos em Pascal, C++ e Java. 3ª ed. Editora Pearson, 2013.

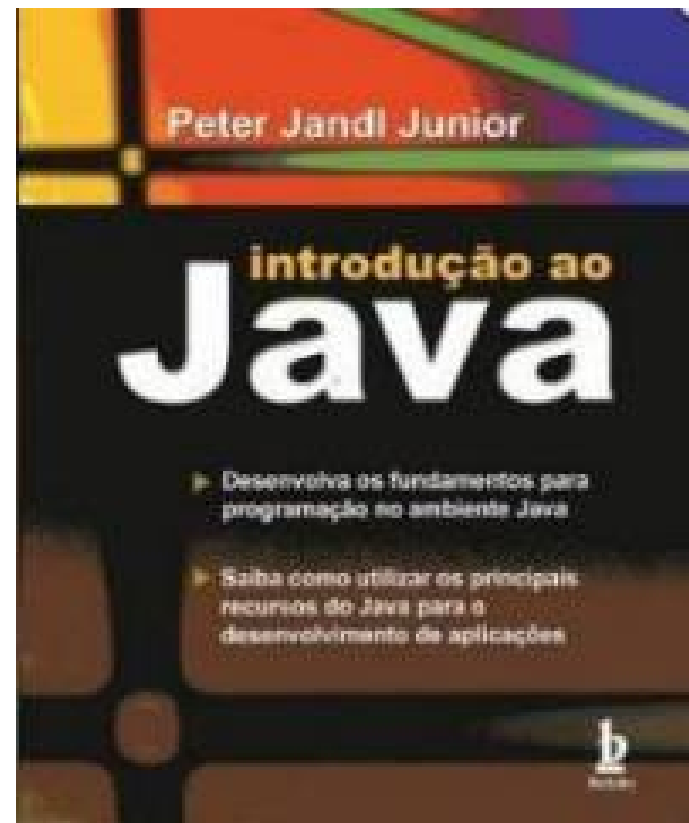


Bibliografia

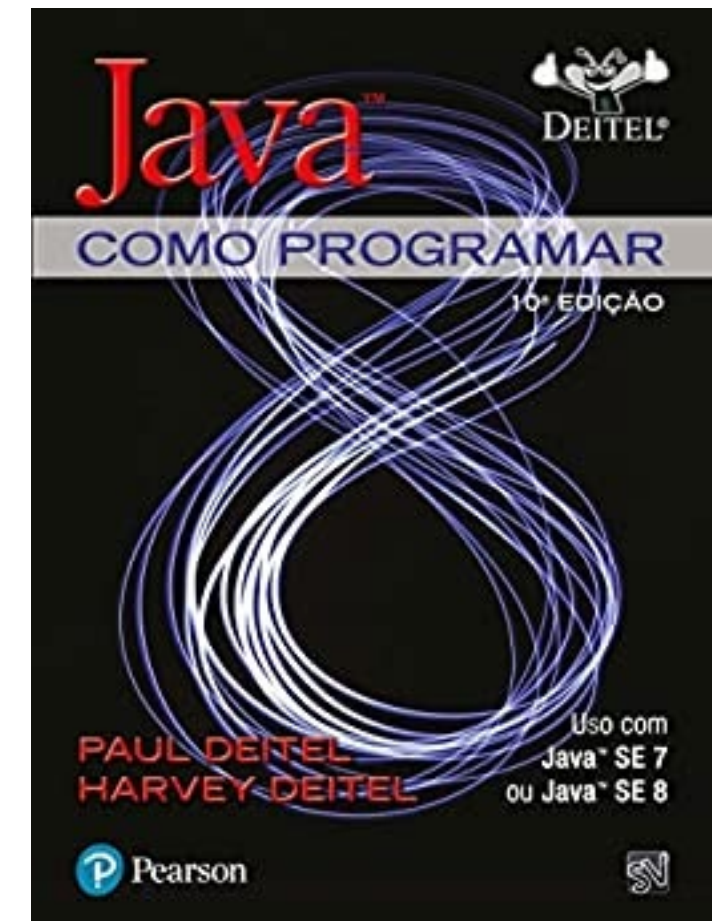


ARNOLD, K., GOSLING, J., HOLMES, D. **Java programming language**. 4th Edition, Editora Addison-Wesley, 2005.

Bibliografia



JANDL JUNIOR, P. Introdução ao Java. São Paulo: Editora Berkeley, 2002.



DEITEL, H. M., DEITEL, P. J. JAVA como programar. 10ª edição. São Paulo: Prentice-Hall, 2010.