

Curso Java primeiros passos: Lógica de Programação

Capítulo: Matrizes

<http://educandoweb.com.br>

Prof. Dr. Nélcio Alves

Matrizes

<http://educandoweb.com.br>

Prof. Dr. Nélcio Alves

Matrizes

Uma matriz corresponde a uma coleção de dados de tamanho fixo, indexada, bidimensional e homogênea

- Indexada: os elementos são acessados por meio de índices
- bidimensional: duas dimensões
- Homogênea: todos dados são do mesmo tipo

	0	1	2
0			
1			
2			
3			
4			

A

Matrizes

Matrizes são também chamadas de **arranjos** bidimensionais

Em Java a primeira posição de uma matriz é a posição 0, 0 (linha 0, coluna 0)

Um arranjo deve ser alocado previamente, antes de ser utilizado. Uma vez alocado, sua quantidade de elementos é fixa

	0	1	2
0			
1			
2			
3			
4			

A

Como criar uma matriz?

declaração → `double[][] A;`

instanciação → `A = new double[3][4];`

Memória RAM

	0	1	2	3
0				
1				
2				

A

Como acessar os elementos de uma matriz?

`A[1][2] = 10;`

Memória RAM

	0	1	2	3
0				
1			10	
2				

A

Problema exemplo

Fazer um programa para ler dois números inteiros positivos M e N, depois ler uma matriz de M linhas e N colunas contendo números inteiros. Em seguida, mostrar na tela a matriz lida conforme exemplo.

Entrada	Saída
2 3	6 3 10
6 3 10	8 12 5
8 12 5	

Entrada	Saída
2 3	6 3 10
6 3 10	8 12 5
8 12 5	

	0	1	2
0	6	3	10
1	8	12	5

mat

Resumo da aula

- Matriz: coleção de dados
 - Tamanho fixo
 - Arranjo bidimensional
 - Indexada
 - Homogênea

- Declaração: `double[][] A;`
- Instanciação: `A = new double[3][4];`
- Acesso: `A[1][2] = 10;`

- Problema exemplo: ler e imprimir na tela uma matriz

	0	1	2	3
0				
1			10	
2				

A

```

import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int M = sc.nextInt();
        int N = sc.nextInt();

        int[][] mat = new int[M][N];

        for (int i=0; i<M; i++) {
            for (int j=0; j<N; j++) {
                mat[i][j] = sc.nextInt();
            }
        }

        for (int i=0; i<M; i++) {
            for (int j=0; j<N; j++) {
                System.out.print(mat[i][j] + " ");
            }
            System.out.println();
        }

        sc.close();
    }
}

```

Exercícios propostos PARTE 1 - testes de mesa com matrizes

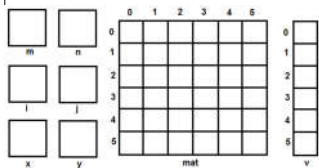
<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

```

n = 3;
for (i=0; i<n; i++) {
    for (j=0; j<n; j++) {
        mat[i][j] = 1 + i + j;
    }
}

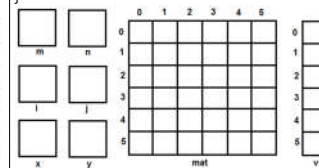
```



```

m = 2;
n = 4;
x = 0;
for (i=0; i<m; i++) {
    x = x + i;
    for (j=0; j<n; j++) {
        x = x + j;
        mat[i][j] = x;
    }
}

```



```

m = 5;
for (i=0; i<m; i++) {
    mat[3][i] = 10;
}
for (i=0; i<m; i++) {
    mat[i][4] = 10 * i;
    mat[i][i] = 50;
}

```

```

n = 3;
for (i=0; i<n; i++) {
    x = 0;
    for (j=0; j<n; j++) {
        mat[i][j] = i + j;
        x = x + mat[i][j];
    }
    v[i] = x;
}

```

```

m = 5;
for (i=0; i<m; i++) {
    v[i] = 10 - i;
}
for (i=0; i<m; i++) {
    mat[i][i] = v[i] / 10;
}

```

```

n = 4;
for (i=0; i<n; i++) {
    for (j=0; j<i; j++) {
        mat[i][j] = (i + 1) * 10;
    }
}

```

```

n = 3;
for (i=0; i<n; i++) {
    for (j=0; j<n; j++) {
        mat[i][j] = 1 + i + j;
    }
}

```

```

m = 2;
n = 4;
x = 0;
for (i=0; i<m; i++) {
    x = x + i;
    for (j=0; j<n; j++) {
        x = x + j;
        mat[i][j] = x;
    }
}

```

```

m = 5;
for (i=0; i<m; i++) {
    mat[3][i] = 10;
}
for (i=0; i<m; i++) {
    mat[i][4] = 10 * i;
    mat[i][i] = 50;
}

```

```

n = 3;
for (i=0; i<n; i++) {
    x = 0;
    for (j=0; j<n; j++) {
        mat[i][j] = i + j;
        x = x + mat[i][j];
    }
    v[i] = x;
}

```

```

m = 5;
for (i=0; i<m; i++) {
    v[i] = 10 - i;
}
for (i=0; i<m; i++) {
    mat[i][i] = v[i] / 10;
}

```

```

n = 4;
for (i=0; i<n; i++) {
    for (j=0; j<i; j++) {
        mat[i][j] = (i + 1) * 10;
    }
}

```

Exercício resolvido 01

<http://educandoweb.com.br>
 Prof. Dr. Nélio Alves

Fazer um programa para ler um número inteiro N e uma matriz quadrada de ordem N contendo números inteiros. Em seguida, mostrar a diagonal principal e a quantidade de valores negativos da matriz.

Exemplo:

Entrada	Saída
3 5 -3 10 15 8 2 7 9 -4	DIAGONAL PRINCIPAL: 5 8 -4 QUANTIDADE DE NEGATIVOS = 2

Entrada	Saída
3 5 -3 10 15 8 2 7 9 -4	

	0	1	2
0	5	-3	10
1	15	8	2
2	7	9	-4

mat

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int N = sc.nextInt();
        int[][] mat = new int[N][N];

        for (int i=0; i<N; i++) {
            for (int j=0; j<N; j++) {
                mat[i][j] = sc.nextInt();
            }
        }

        System.out.println("DIAGONAL PRINCIPAL:");
        for (int i=0; i<N; i++) {
            System.out.print(mat[i][i] + " ");
        }
        System.out.println();

        int cont = 0;
        for (int i=0; i<N; i++) {
            for (int j=0; j<N; j++) {
                if (mat[i][j] < 0) {
                    cont = cont + 1;
                }
            }
        }
        System.out.println("QUANTIDADE DE NEGATIVOS = " + cont);

        sc.close();
    }
}
```

Exercício resolvido 02

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Fazer um programa para ler dois números inteiros M e N. Em seguida, ler uma matriz de M linhas e N colunas contendo números double. Gerar um vetor de modo que cada elemento do vetor seja a soma dos elementos da linha correspondente da matriz. Mostrar o vetor gerado.

Exemplo:

Entrada	Saída
2 3	25.0
7.0 8.0 10.0	10.0
2.0 3.0 5.0	

```
import java.util.Locale;
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Locale.setDefault(Locale.US);
        Scanner sc = new Scanner(System.in);

        int M = sc.nextInt();
        int N = sc.nextInt();

        double[][] numeros = new double[M][N];

        for (int i=0; i<M; i++) {
            for (int j=0; j<N; j++) {
                numeros[i][j] = sc.nextDouble();
            }
        }

        double[] vet = new double[M];

        for (int i=0; i<M; i++) {
            double soma = 0.0;
            for (int j=0; j<N; j++) {
                soma = soma + numeros[i][j];
            }
            vet[i] = soma;
        }

        for (int i=0; i<M; i++) {
            System.out.printf("%.1f\n", vet[i]);
        }

        sc.close();
    }
}
```

Exercícios propostos PARTE 2 - problemas com matrizes

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Problemas com matrizes

Favor fazer o download do PDF com os exercícios.
(link no início do capítulo "Matrizes")

Soluções:

<https://github.com/acenelio/curso-logica-de-programacao-java>

(procure os arquivos com nome estilo "matrizx.java")
