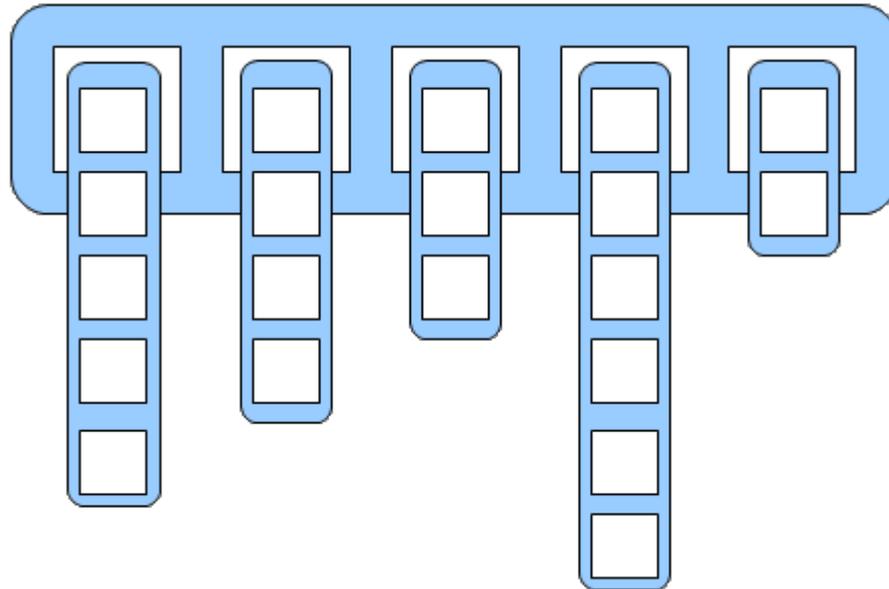


Matrizes

Matrizes (vetores bidimensionais)

Matrizes

Vetores de vetores

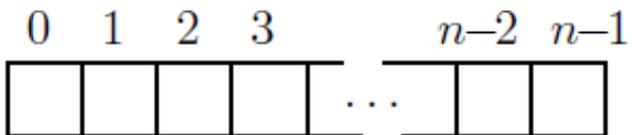




Matrizes

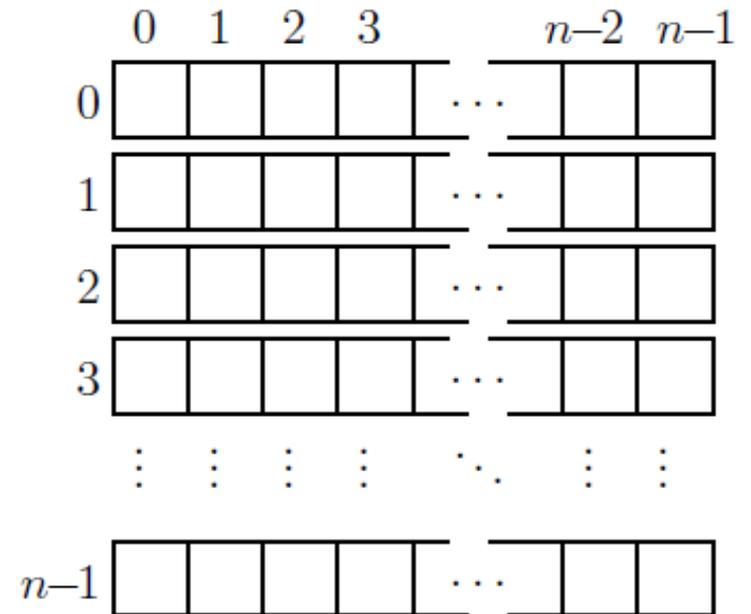
Primeiro índice

Vetor de vetores



Segundo índice

Vetores de inteiros



Matrizes

Sintaxe: regra geral

Indicadores de múltiplos valores

```
tipo nome[][] = new tipo[n][m];
```

Matriz

Quantos valores em cada dimensão

```
nome[i][j] = valor;
```

Posição no vetor

Matrizes

```
int matriz[][] = new int[n][m];
```

onde n e m são as dimensões

```
int matriz[][] = { {34, 56, 4, 78, 89},  
                  {-7, 36, 81, 32, -23},  
                  { 2, 45, 53, 62, 25},  
                  {12, 17, 28, 74, 33},  
                  {65, -34, 47, 8, 1} };  
for (int linha = 0; linha < matriz.length; linha++) {  
    for (int col = 0; col < matriz[linha].length; col++) {  
        System.out.printf("%5d ", matriz[linha][col]);  
    }  
    System.out.println();  
}
```

Matrizes

M[linha][coluna] Acesso ao valor

M.Length Número de elementos do 1º índice
(linhas)

M[0].length Número de elementos do 2º índice
(colunas da linha zero)

Leitura de dados

```
Scanner key = new Scanner(System.in);
float M[][] = new float[5][5];
for (int i= 0; i< M.length; i++) {
    for (int j = 0; j< M[i].length; j++) {
        System.out.print(
            "Entre com o elemento (%d,%d): ",
            i+1, j+1);
        M[i][j] = key.nextFloat();
    }
    System.out.println();
}
```

Impressão

```
for (int i=0;i<5;i++) {  
    System.out.print(" | ");  
    for (int j=0;j<5;j++) {  
        System.out.printf("%9.2f ", M[i][j]);  
    }  
    System.out.println(" |");  
}
```

Métodos: Leitura

```
public static double[][] leiaMatriz(int n, int m) {  
    double M[][] = new double[n][m];  
    Scanner s = new Scanner(System.in);  
    for (int i= 0; i < M.length; i++) {  
        for (int j = 0; j < M[i].length; j++) {  
            System.out.printf(  
                "Entre com o elemento (%d,%d): ",  
                i+1, j+1);  
            M[i][j] = s.nextDouble();  
        }  
    }  
    return M;  
}
```

Métodos: escrita

```
public static void imprimeMatriz(double[][] X) {  
    for (int i= 0; i < X.length; i++) {  
        System.out.print(" | ");  
        for (int j = 0; j < X[i].length; j++) {  
            System.out.printf("%9.2f ", X[i][j]);  
        }  
        System.out.println(" |");  
    }  
}
```

Uso dos Métodos

```
double[][] A, B, C;  
System.out.println(" Entre com uma matriz 3x3");  
A=leiaMatriz(3,3);  
System.out.println(" Você entrou a matriz A=");  
imprimeMatriz(A);  
System.out.println(" Entre com uma matriz 3x3");  
B=leiaMatriz(3,3);  
System.out.println(" Você entrou a matriz B=");  
imprimeMatriz(B);  
  
System.out.println(" A * B=");  
C=multiplicaMatriz(A,B); // não implementado ainda  
imprimeMatriz(C);
```

Prática

1. Operação elemento a elemento

Fazer utilizando console

Crie três matrizes de números reais (double):

- Uma matriz **A** de 4x4 elementos cujos elementos são iguais ao número da linha (de 1 a 4) menos o número da coluna (de 1 a 4);
- Uma matriz **B** que é o resultado da transposição da matriz **A**;
- Uma matriz **C** que é o resultado da soma algébrica da matriz **A** e **B**.

Para **A**, **B** e **C**, imprima a matriz gerada no formato matemático.

Run:

```
A = | Aij ... |  
    | .      |  
    | .      |  
    | .      |
```

2. Determinante

Faça uma classe para operar matrizes. A classe deve ler uma matriz 3x3 do teclado, exibí-la corretamente e em seguida calcular e exibir seu determinante.

Saída: Matriz e determinante

Entrada: Matriz 3x3

```
run:
Entre com o elemento[1][1]: 5
Entre com o elemento[1][2]: 5
Entre com o elemento[1][3]: 5
Entre com o elemento[2][1]: 5
Entre com o elemento[2][2]: 5
...

5      5      5
5      5      5
5      5      5

Determinante: 0
BUILD SUCCESSFUL (total time: 2 minutes 5 seconds)
```

Processamento:

Cálculo do
determinante