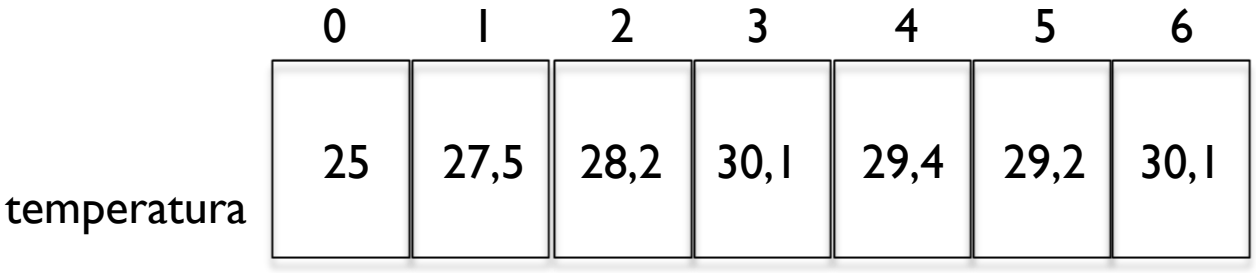
Vetores (*arrays*) – Parte 3
Vetores multidimensionais – Matrizes

Prof. Bruno Nogueira

Até agora...

- Vetores eram unidimensionais
  - Cada posição armazenava um único valor
  - Conseguimos pensar os valores armazenados em uma "linha"; uma única dimensão é necessária para representar a estrutura



Vetores multidimensionais

- Também conhecidos como matrizes

Savings Account Balances for Various Interest Rates Compounded Annually (Rounded to Whole Dollar Amounts)

Year	5.00%	5.50%	6.00%	6.50%	7.00%	7.50%
1	$1050	$1055	$1060	$1065	$1070	$1075
2	$1103	$1113	$1124	$1134	$1145	$1156
3	$1158	$1174	$1191	$1208	$1225	$1242
4	$1216	$1239	$1262	$1286	$1311	$1335
5	$1276	$1307	$1338	$1370	$1403	$1436
6	$1340	$1379	$1419	$1459	$1501	$1543
7	$1407	$1455	$1504	$1554	$1606	$1659
8	$1477	$1535	$1594	$1655	$1718	$1783
9	$1551	$1619	$1689	$1763	$1838	$1917
10	$1629	$1708	$1791	$1877	$1967	$2061

60 valores

Conseguiríamos representá-los com um único vetor unidimensional?

Vetores multidimensionais

- Também conhecidos como matrizes

Savings Account Balances for Various Interest Rates Compounded Annually (Rounded to Whole Dollar Amounts)

Year	5.00%	5.50%	6.00%	6.50%	7.00%	7.50%
1	$1050	$1055	$1060	$1065	$1070	$1075
2	$1103	$1113	$1124	$1134	$1145	$1156
3	$1158	$1174	$1191	$1208	$1225	$1242
4	$1216	$1239	$1262	$1286	$1311	$1335
5	$1276	$1307	$1338	$1370	$1403	$1436
6	$1340	$1379	$1419	$1459	$1501	$1543
7	$1407	$1455				$1659
8	$1477	$1535	$1594	$1655	$1718	$1783
9	$1551	$1619	$1689	$1763	$1838	$1917
10	$1629	$1708	$1791	$1877	$1967	$2061

Qual o índice desse valor?

60 valores

Conseguiríamos representá-los com um único vetor unidimensional?

Sim. Mas teríamos que saber exatamente em qual índice do vetor estaria cada um dos valores

Vetores multidimensionais

- Em vetores multidimensionais, usamos mais de um índice para indexar os elementos
  - Um índice para cada dimensão
  - Para matrizes bidimensionais, duas dimensões são suficientes

Savings Account Balances for Various Interest Rates Compounded Annually (Rounded to Whole Dollar Amounts)

Year	5.00%	5.50%	6.00%	6.50%	7.00%	7.50%
1	$1050	$1055	$1060	$1065	$1070	$1075
2	$1103	$1113	$1124	$1134	$1145	$1156
3	$1158	$1174	$1191	$1208	$1225	$1242
4	$1216	$1239	$1262	$1286	$1311	$1335
5	$1276	$1307	$1338	$1370	$1403	$1436
6	$1340	$1379	$1419	$1459	$1501	$1543
7	$1407	$1455	$1504	$1554	$1606	$1659
8	$1477	$1535	$1594	$1655	$1718	$1783
9	$1551	$1619	$1689	$1763	$1838	$1917
10	$1629	$1708	$1791	$1877	$1967	$2061

Linha 6
Coluna 4

Vetores multidimensionais

- Por convenção, assume-se que o primeiro índice refere-se à linha e o segundo índice refere-se à coluna

```
NomeDoVetor [indice_da_linha][indice_da_coluna]
```

Savings Account Balances for Various Interest Rates Compounded Annually (Rounded to Whole Dollar Amounts)

Year	5.00%	5.50%	6.00%	6.50%	7.00%	7.50%
1	$1050	$1055	$1060	$1065	$1070	$1075
2	$1103	$1113	$1124	$1134	$1145	$1156
3	$1158	$1174	$1191	$1208	$1225	$1242
4	$1216	$1239	$1262	$1286	$1311	$1335
5	$1276	$1307	$1338	$1370	$1403	$1436
6	$1340	$1379	$1419	$1459	$1501	$1543
7	$1407	$1455	$1504	$1554	$1606	$1659
8	$1477	$1535	$1594	$1655	$1718	$1783
9	$1551	$1619	$1689	$1763	$1838	$1917
10	$1629	$1708	$1791	$1877	$1967	$2061

matriz[5][3]
Índices também começam de 0

Sintaxe

- Sintaxe é análoga à dos vetores unidimensionais
  - Declaração

```
tipo [][] nomeDoVetor = new tipo [numLinhas][numColunas];
```

  - Exemplos

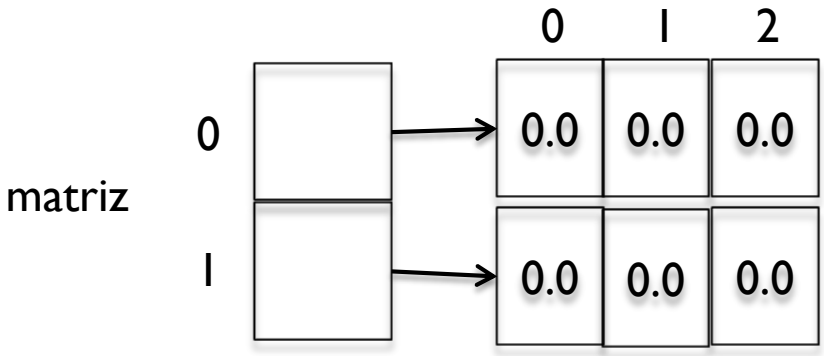
```
char [][] pagina = new char [100][80];
int [][] tabela = new int [10][6];
double [][][] figuraTresD = new double[10][20][30];
```

```java
import java.util.Scanner;
public class MediaProvasMatriz
{
    public static void main (String [] args)
    {
        Scanner teclado = new Scanner(System.in);
        double [][] notas;
        int numAlunos = 4, numProvas = 3;

        notas = new double [numAlunos][numProvas]; // Linhas - Alunos; Colunas - Provas

        for(int i = 0; i < notas.length; i++){
            for(int j = 0; j < notas[i].length; j++){
                System.out.println("Digite a nota " + j + " do aluno " + i + ":");
                notas[i][j] = teclado.nextDouble();
            }
        }

        for(int i = 0; i < notas.length; i++){
            System.out.print("Notas do aluno " + i + ":\t");
            for(int j = 0; j < notas[i].length; j++){
                System.out.print(notas[i][j] + "\t");
            }
            System.out.println();
        }
    }
}
```

Instanciando matrizes

- Quando declaramos vetores multidimensionais, é como se tivéssemos "vetores de vetores"

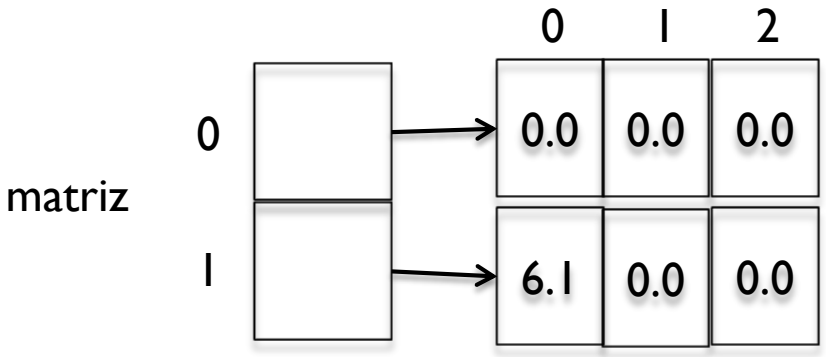
```
double [][] matriz= new double[2][3]; // 2 linhas; 3 colunas
```



Modificando elementos

- Para modificar os valores de uma matriz, precisamos especificar ambos os índices

```
matriz[1][0] = 6.1;
```



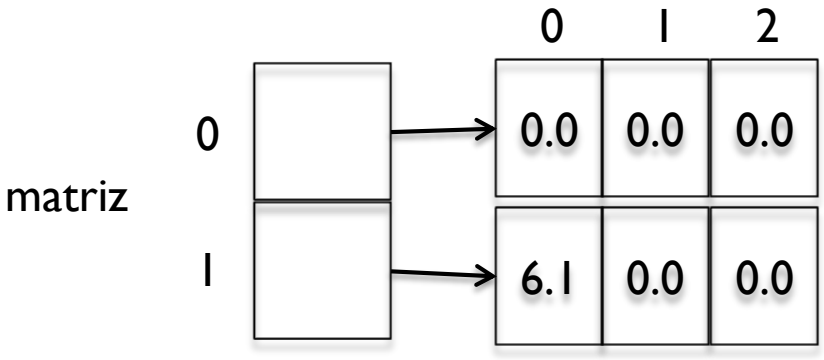
Tamanhos de matrizes

- Para saber o tamanho de uma matriz
  - Número de linhas

```
int numLinhas = matriz.length; // retorna 2
```

  - Número de colunas

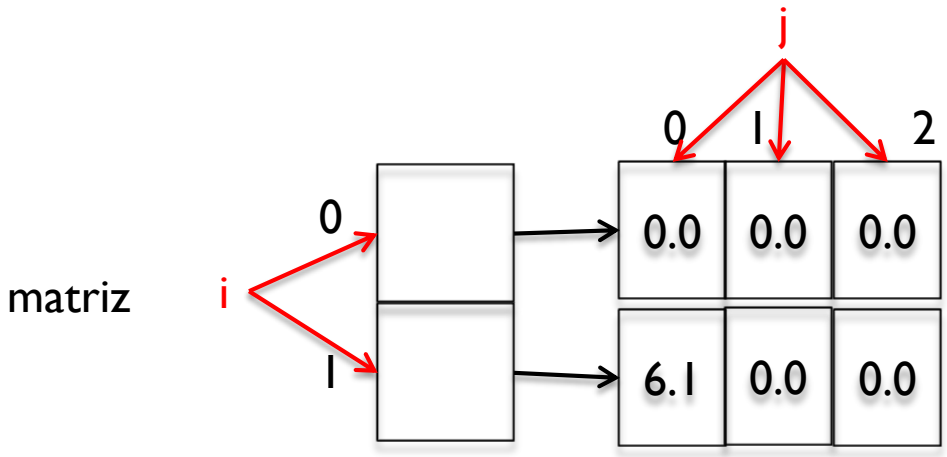
```
int numColunas = matriz[0].length; // retorna 3
```



Acessando elementos

- Também podemos utilizar estruturas de repetição for para percorrer os elementos de uma matriz
  - For aninhados – um para cada dimensão

```
for(int i = 0; i < matriz.length; i++) // percorre as linhas
    for(int j = 0; j < matriz[i].length; j++) // percorre as colunas
       System.out.println(matriz[i][j]);
```



Exercício

- O que será exibido na execução do código abaixo?

```
int[][] testArray = new int[3][4];
for (int row = 0; row < testArray.length; row++)
    for (int col = 0; col < testArray[row].length; col++)
       testArray[row][col] = col;
for (int row2 = 0; row2 < testArray.length; row2++)
{
    for (int col2 = 0; col2 < testArray[row2].length; col2++)
        System.out.print(testArray[row2][col2] + " ");
    System.out.println();
}
```

Exercício

- Modifique o programa MediaProvasMatriz para exibir ao final as médias por aluno e por prova.

Exercício

- Escreva um programa Java denominado ExibeTransposta que receba uma matriz de números inteiros e imprima a sua transposta na tela.

Exercício

- Faça um programa que receba do usuário duas matrizes de números de ponto flutuante de dupla precisão. As matrizes têm tamanho arbitrário, digitados pelo usuário. Faça um programa que gere uma terceira matriz, resultante da multiplicação das outras duas. Caso não seja possível fazer a multiplicação, mostre uma mensagem para o usuário avisando-o da impossibilidade.

Exercício

- Faça um programa em Java que receba uma matriz 3 x 3, armazenando caracteres, representando uma configuração de tabuleiro de jogo da velha ao final do mesmo. As posições são preenchidas com x, o ou b. Este último indica uma posição em branco. Verifique se houve vencedor neste jogo ou se o mesmo terminou empatado.