

A thick, wavy yellow line runs vertically along the left side of the slide, starting from the top and extending to the bottom. It has a slightly irregular, hand-drawn appearance.

ARRAYS MULTIDIMENSIONAIS

MICHELLE NERY NASCIMENTO
PUC MINAS
ALGORITMOS E TÉCNICAS DE PROGRAMAÇÃO

DEFINIÇÃO

- São arrays de duas ou mais dimensões.
- Trabalharemos com arrays bidimensionais que exigem dois índices, um para linha e outro para coluna, para identificar um elemento em particular. Por convenção o primeiro identifica a linha e o segundo a coluna.

	Column 0	Column 1	Column 2	Column 3
Row 0	a[0][0]	a[0][1]	a[0][2]	a[0][3]
Row 1	a[1][0]	a[1][1]	a[1][2]	a[1][3]
Row 2	a[2][0]	a[2][1]	a[2][2]	a[2][3]

- Exemplo de array bidimensional com 3 linhas e 4 colunas.
- Todo elemento no array acima é identificado por um nome de elemento da forma a[l] [c], onde a é o nome do array, l e c são os índices (linha, coluna).

SINTAXE

tipo [][] nome_variável = new **tipo** [tam1][tam2];

- Exemplos:

1) **int** [][] mat = new **int**[2][2];

2) **int** [][] matriz =

{

 {1, 2, 3, 4}, // valores da linha 0

 {5, 6, 7, 8} // valores da linha 1

}; // o tamanho da matriz é 2 x 4 (2 linhas e 4 colunas)

- Obs.: vírgulas separam os blocos.

3) `int [][] mat;`

`mat = new int [2,2];`

`mat[0][0] = 1;`

`mat[0][1] = 2;`

`mat[1][0] = 3;`

`mat[1][1] = 4;`

ou lista inicializadora: `int [][] mat = { {1, 2} , {3, 4} };`

- Os valores são agrupados por linha em chaves.
- Assim, 1 e 2 inicializam `mat[0][0]` e `mat[0][1]` e 3, 4 `mat[1][0]` e `mat[1][1]`
- O compilador determina o número de linhas contando o número de listas subinicializadas (conj. de chaves) na lista inicializadora principal.

EXEMPLO

```
public static void main (string[ ] args)
{
    //declara o array de inteiros 4X3
    int [ ][ ] mat = new int [4][3];
    //preenche o array
    for (int i =0; i < 4; i++)
    {
        for (int j=0; j < 3; j++)
        {
            mat [i][j] = i + j;
        }
    }
    //imprime o conteudo do array
    for (int i =0; i < 4; i++)
    {
        for (int j=0; j < 3; j++)
        {
            System.out.println("Mat[" + i + "][" + j + "]: " + mat[i][j] );
        }
    }
}
```

```
Scanner sc = new Scanner (System.in);
```

```
public static void imprimeMatriz(int[ ][ ] mat)
```

```
{ int i, j;
```

```
for (i = 0; i < 3; i++)
```

```
{
```

```
for (j = 0; j < 3; j++)
```

```
{
```

```
System.out.println ("Mat[" + i + "][" + j + "]: " + mat[i][j] );
```

```
}
```

```
System.out.println ();
```

```
}
```

```
}
```

```
public static void Main()
```

```
{
```

```
int[ ][ ] mat = new int[3][3];
```

```
int i, j;
```

```
for (i=0; i<3; i++)
```

```
{
```

```
for (j=0; j<3; j++)
```

```
{
```

```
System.out.println ("\nDigite o elemento [" + i + "," + j + "] da matriz:");
```

```
mat[i][j] = sc.nextInt();
```

```
}
```

```
}
```

```
imprimeMatriz(mat);
```

```
}
```

Faça um programa que leia os elementos de uma matriz 3x3 e um método que imprima os seus elementos.

SINTAXE

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

Obs.:

- Os arrays multidimensionais são mantidos como arrays de arrays unidimensionais. Portanto, o array “mat” na declaração acima é na realidade composto de dois arrays unidimensionais separados – um que contém o valor na primeira lista de inicializadores aninhados {1,2} e outro que contém o valor na segunda lista de inicializadores aninhados {3,4}.
- Portanto, o próprio array “mat” é um array de dois elementos, cada um desses elementos é um array unidimensional de valores int.


```

public class ExemploArray
{
    public static void main(String[] args)
    {
        int[ ][ ] array1 = {{1, 2, 3}, {4, 5, 6}};
        int[ ][ ] array2 = {{1, 2}, {3}, {4, 5, 6}};

        System.out.println("Elementos no vetor 1 por linha:");
        imprimirArray(array1);
        System.out.print(" Elementos no vetor 2 por linha ");
        imprimirArray(array2);
    }

    public static void imprimirArray(int[ ][ ] array)
    {
        for (int row = 0; row < array.length; row++)
        {
            for (int column = 0; column < array[row].length; column++){
                System.out.print("Elementos:", array[row][column]);
            }
            System.out.println();
        }
    }
} // fim da classe ExemploArray

```

EXERCÍCIOS

- 1) Faça um programa que leia os elementos de uma matriz quadrada (mesmos números de linhas e colunas) e mostre os elementos da diagonal principal.
- 2) Faça um programa que leia os elementos de uma matriz e mostre a média dos elementos.
- 3) Faça um programa que carregue uma matriz 3×5 com números inteiros, calcule e mostre a quantidade de elementos entre 15 e 20.

AGRADECIMENTO

- À professora Soraia Lúcia pela cessão do material deste guia de aula