

Aulas 5 e 6

Considerando o código abaixo...

```
public class Lista {  
    public static void main(String[] args) {  
        int [][] num = {{1,1,1},{4,4,4},{9,9,9}};  
        boolean passou = false;  
        if (num[1][1]==4) {  
            passou=true;  
            System.out.println(passou);  
        } else {  
            System.out.println(passou);  
        }  
        if(passou) {  
            for (int i=0; i<=2; i++) {
```

```
                for (int j=0; j<=2; j++) {  
                    if (i==0 && j==0) {  
                        System.out.print("");  
                    }  
                    System.out.println(num[i][j]);  
                }  
            }  
            System.out.println("Tam num -> " +num.length);  
        } else {  
            System.out.println("\nErro");  
        }  
    }  
}
```

...tente debugar e responder:

a) Qual a saída em `System.out.println(num[i][j]);`?

É importante notar que esta saída está dentro de uma estrutura de repetição dupla, intercalando os índices da matriz. A cada iteração temos a saída de um valor, ou seja: 1 1 1 4 4 4 9 9 9 .

b) Que tipo de validação faz a variável `passou` e por que é importante no código?

É uma variável de tipo booleana. Ela é inicializada com `false` e recebe `true` se a posição 1 1 da matriz for igual a 4. Notamos que isso é verdade. Caso não fosse, o bloco `if(passou)` não seria executado e o programa retornaria a mensagem "Erro", ou seja, a matriz não seria impressa.

c) Qual tamanho será impresso em `num.length` ?

Tamanho 3.

d) Modifique o código para imprimir a saída "1,4,9, 1,4,9, 1,4,9"

A maneira mais simples é alterar a inicialização da matriz, passando diretamente estes valores, da seguinte forma:

(...)

```
int [][] num = {{1,4,9},{1,4,9},{1,4,9}};
```

(...)

f) Crie os métodos `criaMatriz()` e `mostraMatriz(int[][]n)` para deixar o `main()` mais "limpo" e devolver exatamente as mesmas saídas que o código acima faz.

```
public class Lista {  
    public static void criaMatriz() {  
        int [][]num = {{1,1,1},{4,4,4},{9,9,9}};  
  
        mostraMatriz(num);  
    }  
    public static void mostraMatriz(int [][]num) {  
        boolean passou = false;  
        if (num[1][1]==4) {  
            passou=true;  
            System.out.println(passou);
```

```

    } else {
        System.out.println(passou);
    }
    if(passou) {
        for (int i=0; i<=2; i++) {
            for (int j=0; j<=2; j++) {
                if (i==0 && j==0) {
                    System.out.print("");
                }
                System.out.println(num[i][j]);
            }
        }
        System.out.println("Tam num -> " + num.length);
    } else {
        System.out.println("\nErro");
    }
}

public static void main(String[] args) {
    criaMatriz();
}
}

```

g) No método criaMatriz() instancie o objeto Scanner(System.in) para permitir a entrada de dados via teclado.

```

import java.util.*;
public class Lista2 {
    public static void criaMatriz() {
        int[][] num = new int[3][3];
        Scanner sc = new Scanner(System.in);
        System.out.println("Digite 9 numeros.");
        for (int i=0; i<=2; i++) {
            for (int j=0; j<=2; j++) {
                System.out.print("Nro "+i+ " "+j+ " -> ");
                num[i][j] = sc.nextInt();
            }
        }
        mostraMatriz(num);
    }
    public static void mostraMatriz(int[][] num) {
        boolean passou = false;
        if (num[1][1]==4) {
            passou=true;
            System.out.println(passou);
        } else {
            System.out.println(passou);
        }
        if(passou) {
            for (int i=0; i<=2; i++) {

```

```

        for (int j=0; j<=2; j++) {
            if (i==0 && j==0) {
                System.out.print("");
            }
            System.out.println(num[i][j]);
        }
    }
    System.out.println("Tam num -> " + num.length);
} else {
    System.out.println("\nErro");
}
}
}
public static void main(String[] args) {
    criaMatriz();
}
}

```

h) Como você modificaria este código para permitir à variável num receber as notas de um aluno a cada trimestre e, ao final, devolver como resultado se ele passou ou não e qual seria a sua média anual?

// Detalhe deste código: modificamos a variável “num” para “notas” e modificamos o nome da classe para “CalculaMediaAnual”. Observe que o main tem apenas uma linha que é a chamada dos dois métodos da classe.

```

import java.util.*;
public class CalculaMediaAnual {
    public static double[][] criaMatriz() {
        double[][] notas = new double[3][3];
        Scanner sc = new Scanner(System.in);
        for (int i=0; i<=2; i++) {
            for (int j=0; j<=2; j++) {
                if (i==0 & j==0) {
                    System.out.println("Digite as notas do Primeiro trimestre");
                }
                if (i==1 & j==0) {
                    System.out.println("Digite as notas do Segundo trimestre");
                }
                if (i==2 & j==0) {
                    System.out.println("Digite as notas do Terceiro trimestre");
                }
                System.out.print("Nota "+i+ " "+j+ " -> ");
                notas[i][j] = sc.nextDouble();
            }
        }
        return notas;
    }
    public static void mostraMatriz(double[][] notas) {
        double media=0.0;
    }
}

```

```

    for (int i=0; i<=2; i++) {
        for (int j=0; j<=2; j++) {
            media += notas[i][j];
        }
    }
    if((media/9)>=7) {
        System.out.println("Aprovado com media anual "+(media/9));
    } else {
        System.out.println("Reprovado com media anual "+(media/9));
    }
}

public static void main(String[] args) {
    mostraMatriz(criaMatriz());
}
}

```

i) Qual a diferença entre uma matriz `notasTrimestrais[2][2]` e os vetores `primeiroTrimestre[2]`, `segundoTrimestre[2]` e `terceiroTrimestre[2]` no programa acima?

Para o caso de uma matriz `notasTrimestrais[2][2]`, podemos popular os espaços com apenas duas estruturas de repetição alinhadas com variáveis `i` e `j` para acesso aos índices. Se a escolha fosse um vetor para cada trimestre, teríamos que usar um for para cada vetor e isso resultaria em um código maior, entretanto, dadas estas diferenças, funcionaria da mesma forma. Ou seja, trata-se de uma escolha do desenvolvedor.

j) E se a regra do negócio não prevesse a quantidade certa de notas para cada trimestre, como seria a implementação do código?

De fato existe a limitação do tamanho da matriz ou vetor quando se desenvolve um código. Uma alternativa será o `ArrayList`, um recurso interessante, que permitiria adicionar as notas da seguinte forma: `notas.add(sc.nextDouble());` e a cada iteração a nota é adicionada na última posição da lista.

h) Diante do erro `java.lang.ArrayIndexOutOfBoundsException` como a linguagem Java permite evitar que o programa seja encerrado, ou seja, se “recuperar do erro”?

O erro citado interrompe bruptalmente a execução do programa. Uma situação destas poderia provocar a perda de trabalho e muito transtorno para o usuário. Assim, o Java permite se recuperar deste erro com o código `try... catch`. Vejamos:

```

try {
    int[][] notas = {{1, 1,1}, {4,4,4}, {9,9,9}}; // criamos a matriz
    System.out.println(notas[4][4]);           // tentativa de imprimir vai falhar
} catch (Exception e) {
    System.out.println(e);                     // ao invés de interromper, imprime o erro e segue
}

```

Esse recurso será melhor discutido nas próximas aulas.