

## FACAMP – Linguagens de Programação

### Aula 2 – Exercícios

Todas as operações que envolvam carregar ou acessar valores em uma matriz devem ser realizadas com índices, dentro de laços de repetição.

Elabore um programa (classe Java, com método main) diferente para cada questão.

1. Crie um método que recebe 4 valores inteiros como parâmetros e retorna a média (double) dos valores.
2. Crie um método que recebe 3 valores inteiros como parâmetros e retorna o maior deles.
3. Crie um método que receba um valor inteiro e verifique se ele é par. O método deve retornar um valor booleano true, se for par, e false, caso contrário.
4. Crie um método para receber as três notas de um aluno, em formato double, calcular e retornar a média ponderada com os respectivos pesos: 0.3, 0.3 e 0.4.
5. Crie um método para calcular a área de um trapézio retorná-la em formato double. Os valores de entrada (base maior, base menor e altura) devem ser passados como parâmetros de entrada para o método, em formato double.

$$\text{Area\_Trapezio} = ((\text{BaseMaior} + \text{BaseMenor}) * \text{Altura}) / 2$$

6. Crie um método para calcular o fatorial de n (n!) que é a sequência de produtos sucessivos de 1 a N). O valor de n (inteiro) deve ser fornecido como parâmetro de entrada do método. O valor do fatorial deve ser retornado pelo método.
7. Crie um método que recebe por parâmetro o raio de uma esfera e calcula o seu volume.  
$$v = (4 * \pi * r^3) / 3$$
8. Crie um método que recebe um número inteiro como parâmetro e verifica se o número é primo (se é divisível apenas por 1 e por ele mesmo). O método deve retornar um booleano true, se for primo, e false, caso contrário.
9. Crie um método que recebe como parâmetros os valores necessários para o cálculo da fórmula de Bhaskara. O método deve calcular o discriminante d:  $d = b^2 - 4ac$

Se  $d > 0$ , então deve calcular e exibir as duas raízes 
$$x = \frac{-b \pm \sqrt{d}}{2a}$$

Se  $d = 0$ , então deve calcular e exibir a raiz dupla 
$$x = \frac{-b}{2a}$$

Se  $d < 0$ , o método deve exibir a mensagem “Equação não possui raízes reais”.

10. Crie um método que recebe como parâmetro um período de tempo em segundos e exiba na tela o período expresso em horas, minutos e segundos.
11. Crie um método que recebe a idade de uma pessoa em anos, meses e dias e retorna essa idade expressa em dias (por simplicidade, pode ignorar os anos bissextos).
12. Crie um método que receba um número inteiro e verifique ele é perfeito ou não. Um número é dito perfeito quando ele é igual à soma dos seus divisores, excetuando-se ele próprio (Ex.: 6 é perfeito, pois  $6 = 1 + 2 + 3$ , que são seus divisores). Crie um método deve retornar um valor booleano true, se for perfeito, e false, caso contrário.
13. Crie um método que receba uma matriz de inteiros como parâmetro e retorne a soma (inteiro) dos elementos da sua diagonal principal.

14. Refaça o exercício anterior, mas para a diagonal secundária.
15. Refaça novamente, mas agora para os elementos que estão acima a diagonal principal.
16. Refaça de novo, para os elementos abaixo da diagonal principal.
17. Crie um método que receba uma matriz de inteiros como parâmetro e retorne a média (double) dos elementos da sua diagonal principal.
18. Refaça o exercício anterior, mas para a diagonal secundária.
19. Refaça novamente, mas agora para os elementos que estão acima a diagonal principal.
20. Refaça de novo, para os elementos abaixo da diagonal principal.
21. Crie um método que receba uma matriz de inteiros como parâmetro e verifique se ela é simétrica. Crie um método deve retornar um booleano true, se for simétrica, ou false, caso contrário.
22. Crie um método que recebe três vetores de números inteiros, por parâmetro. Ao final do método, o segundo vetor deverá conter os elementos pares do primeiro, e o terceiro deverá conter os ímpares.
23. Crie um método que receba, como parâmetros, um número inteiro e uma matriz de inteiros e realize o produto escalar do número em cada elemento da matriz.
24. Crie um método que receba uma matriz de inteiros como parâmetro e retorne a sua transposta. (Cuidado! A matriz pode não ser quadrada!)
25. Crie um método que receba duas matrizes de inteiros e retorne uma matriz de inteiros contendo o resultado da multiplicação das duas. O método deve testar se a multiplicação é possível antes de realizá-la.
26. Crie um método que receba dois vetores de números inteiros. O método deve procurar a ocorrência da sequência de números do primeiro vetor entre os números do segundo vetor. Caso exista, deve retornar o índice do primeiro elemento da sequência encontrada no segundo vetor. Caso não exista, deve retornar -1.
27. Crie um método que receba como parâmetros uma matriz e um vetor de inteiros. O método deve inserir em cada posição do vetor a soma de cada coluna da matriz.
28. Refaça o exercício anterior, mas para a soma de cada linha da matriz.
29. Crie um método que ordene um vetor de números inteiros `ordenaVetor(int[] numeros)`
30. Crie um método que receba um vetor de inteiros e preencha com números aleatórios de 0 a 999, sem repetição. Caso o vetor tenha mais que 1000 posições, deve preenchê-lo com 0.
31. Crie um método `vetorParaMatriz(int[] v, int[][] m)` que carrega a matriz com os valores recebidos no vetor.
32. Faça o método `matrizParaVetor(int[][] m, int[] v)` que carrega o vetor com os valores recebidos na matriz.
33. Crie um método que ordene uma matriz `ordenaMatriz(int[][] m)` que, para isso, chame os métodos acima (`vetorParaMatriz`, `matrizParaVetor` e `ordenaVetor`). Para testar, carregue inicialmente a matriz com números aleatórios de 1 a 999.