

Lista de exercícios - Revisão de programação

Instruções para os exercícios:

- Esses exercícios são para fixação do conteúdo e não têm impacto no cálculo da menção dos alunos.
- Alunos que não estiveram presentes na aula online (via MS-Teams) deverão entregar os exercícios, via *pull-request*, no repositório da disciplina. Essa entrega será utilizada para que a presença na aula seja contabilizada.
- Bons estudos!

1. **Entrada e saída – Hello World:** Crie um programa que seja capaz de ler do teclado, via terminal (entrada textual), o nome completo e o ano de nascimento de uma pessoa e ao final imprima uma mensagem informando o nome e a idade da pessoa.

Exemplo:

nome: “Maria Celia”

ano de nascimento: “2000”

Saída: “Olá Maria Celia! Você possui 21 anos de idade.”

2. **Entrada e saída gráfica** Crie uma nova versão do programa criado para o exercício 1 em que a entrada e saída dos dados seja feita através de elementos gráficos da biblioteca `JOptionPane`. Lembre-se: valores lidos do teclado usando essa biblioteca são sempre **Strings**. No caso de valores numéricos, esses devem ser convertidos (*cast*) para o tipo de dado apropriado (`int`, `float`, `double`, etc ...).

3. **Estruturas de decisão – if-else, switch-case:** Crie um programa que seja capaz de calcular o perímetro e a área das figuras geométricas círculo, triângulo, quadrado e retângulo. Inicialmente o programa deve solicitar ao usuário para qual figura geométrica ele deseja informar os dados. Em seguida os dados da figura geométrica escolhida deverão ser informados pelo usuário pelo teclado. Exemplo: para círculo deve ser informado apenas o valor do raio, para triângulos devem ser informados os valores dos três lados, e de modo similar para as demais figuras geométricas. Os valores dos dados deverão ser testados para que valores inválidos (negativos ou zerados) não sejam informados. Sugestão: utilize switch-case para a seleção da figura geométrica.

Exemplo:

Figura geométrica: triângulo.

Lado 1: 3

Lado 2: -4

Lado 2: 0

Lado 2: 4

Lado 3: 5

Perímetro = 12

4. **Estruturas de repetição – for, while, do-while:** Crie três versões para um mesmo

programa de cálculo dos termos e da soma dos termos de uma progressão aritmética (PA) para os valores de n , a_0 e r informados pelo usuário. Cada versão do programa utilizará uma das estruturas de repetição informadas no enunciado dessa questão:

Exemplo:

1o. termo (a_0) = 2

Razão (r) = 3

Número de termos (n) = 5

Termos da PA = 2, 5, 8, 11, 14.

Soma dos termos da PA = 40.

5. **Vetores:** Altere o programa criado no exercício anterior de modo a utilizar vetores na solução do problema. Nesse caso o valor lido para n deverá ser utilizado para criação do vetor de inteiros a receber os valores da PA. Em um primeiro momento o programa deverá calcular cada valor da PA e aloca-lo em sua posição correspondente nesse vetor. Em seguida o programa deverá “varrer” o vetor para apresentar os valores e calcular a soma dos termos.
6. **Matrizes:** Crie um programa que leia as dimensões m e n , sendo m o número de linhas e n o número de colunas, de duas matrizes de inteiros . Assim que as dimensões das matrizes forem informadas pelo usuário, realize a leitura de cada um dos valores das matrizes. Imprima as matrizes.
7. **Métodos (funções e procedimentos):** Altere o programa do exercício anterior de modo a adicionar três novos métodos.
 - crie um método chamado `calcularDeterminante` que recebe como parâmetro uma matriz e retorne o valor da determinante da matriz (um valor inteiro).
 - crie um método chamado `somarMatrizes` que recebe duas matrizes como parâmetros e retorne uma terceira matriz, formada pela soma das matrizes. Esse método deve ser capaz de identificar se a soma das matrizes é possível de ser realizada.
 - Crie um método chamado `multiplicarMatrizes` que recebe duas matrizes como parâmetros e retorne uma terceira matriz, formada pela multiplicação das matrizes. Esse método deve ser capaz de identificar se a multiplicação das matrizes é possível de ser realizada.