



03.20. Desafio – estruturas de repetição

(Módulo 03)

1) Desafio da estrutura for

Escreva um programa que solicita 10 números para o usuário e imprime na saída a soma dos números digitados.

Permita que o usuário digite apenas números pares. Caso um número ímpar seja informado, ignore e solicite um novo número ao usuário.

2) Desafio da estrutura for (intermediário)

Escreva um programa que solicita um número inteiro positivo para o usuário e imprime na saída uma mensagem dizendo se o número digitado é um número primo ou não.

Números primos são números naturais que possuem dois divisores: 1 e ele mesmo.

Por exemplo, o número 1 não é um número primo, porque ele tem apenas o número 1 (ele mesmo) como divisor. O número 2 é primo, porque ele é divisível por 1 e ele mesmo. O número 4 não é primo, porque ele é divisível por 1, 2 e 4.

Para saber se um número é divisível por outro, basta usar o operador de módulo (resto) e avaliar se é igual a 0.

3) Desafio da estrutura while

Escreva um programa que solicita números inteiros ao usuário de forma contínua e soma todos os números informados.

O programa deve parar de solicitar mais números e imprimir a soma deles na saída apenas quando o valor total somado for igual ou superior a 100.

4) Desafio da estrutura while (intermediário)

Escreva um programa que solicita um número inteiro para o usuário e imprime na saída o número com os dígitos invertidos.

Por exemplo, se o número digitado for 98765, a saída deve ser 56789.

Este é um exercício intermediário, especialmente porque precisará usar suas habilidades em lógica. É possível fazer isso apenas com um loop e cálculos matemáticos.

5) Desafio da estrutura do-while

Escreva um programa que solicita números inteiros para o usuário de forma "infinita", até que o usuário escolha não digitar mais nenhum número (pergunte ao usuário após informar cada número).

Somme todos os números ímpares e todos os números pares e imprima na saída o resultado.