

# Prova de Programação de Computadores

Prof.: Carlos Camarão

01 de Dezembro de 2009

1. Escreva um programa que leia um número inteiro positivo  $n$ , em seguida  $n$  valores inteiros positivos  $v_1, \dots, v_n$ , depois um valor inteiro  $m$ , e imprima a soma dos valores dentre  $v_1, \dots, v_n$  que são maiores que  $m$ .

Os valores devem ser lidos do dispositivo de entrada padrão e podem estar separados entre si por um ou mais espaços ou linhas.

Por exemplo, para a entrada:

4 1 2 3 4 2

a saída deve ser: 7 (pois 3 e 4 são os valores maiores que 2 dentre os 4 valores 1, 2, 3, 4).

2. Escreva um programa que leia um valor inteiro positivo  $n$ , em seguida  $n$  valores inteiros positivos  $v_1, v_2, \dots, v_n$ , depois um valor positivo  $m$ , e imprima o valor resultante do cálculo de:

$$((m - 1) + v_1) \times ((m - 2) + v_2) \times \dots \times ((m - n) + v_n)$$

Os valores devem ser lidos do dispositivo de entrada padrão e podem estar separados entre si por um ou mais espaços ou linhas.

Por exemplo, para a entrada:

4 1 2 3 4 10

a saída deve ser: 1000 (pois esse é o resultado de  $(9 + 1) \times (8 + 2) \times (7 + 3) \times (6 + 4)$ ).

3. Escreva um programa que leia um inteiro positivo  $n$ , depois  $n$  valores inteiros  $a_1, \dots, a_n$ , e imprima uma mensagem indicando se a sequência de valores  $a_1, \dots, a_n$  é crescente (ou seja, cada valor  $a_{i+1}$  é maior que  $a_i$ , para  $i = 1, \dots, n - 1$ ), decrescente (ou seja, cada valor  $a_{i+1}$  é menor que  $a_i$ , para  $i = 1, \dots, n - 1$ ), constante (todos os valores  $a_i$  são iguais entre si, para  $i = 1, \dots, n$ ), ou nenhuma dessas opções (nem crescente, nem decrescente, nem constante).

Por exemplo, para a entrada:

4 1 2 3 4

a saída deve ser: **"Crescente"** (pois a sequência dos 4 valores 1,2,3,4 é crescente).

Para a entrada:

4 1 2 3 3

a saída deve ser: **"Nem crescente, nem decrescente, nem constante"** (pois a sequência dos 4 valores 1,2,3,3 não é crescente, nem decrescente, nem constante).

Dica: Use três booleanos **crescente**, **decrescente**, **constante**, inicialize-os com **true** e atualize-os ao percorrer (a partir do segundo valor da sequência) a sequência de valores (armazenada em um arranjo) examinando valores adjacentes da sequência.