

INTRODUÇÃO À CIÊNCIA DA COMPUTAÇÃO I
SCC0221

LISTA DE EXERCÍCIOS II - VETORES E MATRIZES

Em todos os exercícios, considere que o usuário pode digitar valores não correspondentes ao que é pedido (por exemplo, dar de entrada um número negativo enquanto o que foi pedido foi um número natural). Recomenda-se que essas exceções sejam tratadas utilizando estruturas condicionais (por exemplo, finalizando a execução do programa previamente através de `exit(0)` ou `return 0`).

1. Escreva um algoritmo em C que receba o nome e altura de 10 pessoas, em sequência. No final do programa, imprima na tela o nome e altura da maior dentre as dez pessoas (lembre-se que é inconsistente declarar 10 variáveis diferentes para armazenamento, utilize vetores no lugar).
2. Escreva um programa em C que receba 20 números de entrada, em sequência. Em seguida, exiba na tela com o respectivo rótulo,
 - (i) Quantos e quais são pares.
 - (ii) Quantos e quais são múltiplos de 3.
 - (iii) Quantos e quais são primos.
3. Implemente um programa em C que receba o tamanho e o conteúdo de duas sequências de mesmo tamanho de números inteiros, A e B , separadas por linhas. No final da execução, exiba na tela os valores da sequência C tal que $C[i] = A[i] \cdot B[i]$.
4. Faça um algoritmo em C que receba um inteiro positivo n e, em seguida, uma lista de n números separados por espaço e exiba na tela, após todas as entradas, quais números foram dados de entrada. **Ao exibir os valores na tela, possíveis repetições de números na lista não devem ser consideradas.**
5. Construa um programa em C que calcule a **média harmônica** de n números inteiros dados de entrada. A média harmônica dos números k_1, k_2, \dots, k_n pode ser calculada a partir da relação

$$M_{\text{harm}} = \frac{n}{\frac{1}{k_1} + \frac{1}{k_2} + \frac{1}{k_3} + \dots + \frac{1}{k_n}}.$$

Dica: Em uma média harmônica, nenhum dos números pode ser zero.

6. Escreva um algoritmo em C que realize as seguintes operações, em sequência:
 - (i) Declare um vetor de **caracteres**, com 50 espaços, e uma variável contadora, inicializada em zero.
 - (ii) Introduza um loop que, a cada iteração, receba de entrada um caractere do usuário e armazene, dentro do vetor, na posição representada pelo contador, até receber um '@' (não incremente a variável contadora quando ele aparecer). Ao final de cada iteração, incremente o contador.

- (iii) Saindo do loop, insira no vetor, na posição do contador, o caractere '\0'.
- (iv) Experimente exibir o vetor na tela, com um `printf`, utilizando a máscara '%s'.

(*Reflexão*) Podemos dar a esse vetor outro nome, que tal **cadeia de caracteres**? Ou *string*, para os mais íntimos. :)

7. Em computação, um dos métodos mais simples de ordenação é a *ordenação por flutuação*, mais conhecida como *Bubble Sort*. Esse algoritmo pode ser utilizado para a ordenação de pequenos vetores de números inteiros, e possui esse nome pois seu método de ordenação consiste em, a cada iteração no vetor, levar o maior número ainda não ordenado para o final do vetor. Seus cuidadosos passos podem ser pontuados em poucos tópicos, sendo eles:
- (i) Comece iterando o vetor a partir de sua posição inicial.
 - (ii) Compare o número da atual posição do vetor com seu sucessor.
 - (iii) Se o número da posição atual do vetor for maior do que seu sucessor, troque-os de posição entre si. Caso contrário, não faça nada e vá para a próxima posição do vetor.
 - (iv) Repita esse processo iterando sobre o vetor várias vezes até que, após percorrer todo o vetor, nenhuma troca seja feita.

Implemente um algoritmo do tipo *Bubble Sort* utilizando linguagem C.

8. Construa um código em C que exiba na tela a matriz transposta (isto é, a matriz cujas linhas são trocadas pelas colunas) de uma matriz dada de entrada. As dimensões da matriz também são fornecidas antes de seus valores.
9. Implemente um programa em C que realize a soma de duas matrizes quadradas de ordem k , com k dado de entrada. Os valores de cada matriz também devem ser fornecidos de entrada.
10. Implemente um algoritmo em C que calcule e exiba na tela o determinante de uma matriz quadrada de ordem 2 ou 3. A dimensão da matriz e o valor de seus campos devem ser dado de entrada.
11. Em um torneio de computação competitiva, cada um dos 10 times precisa passar por uma sequência de 8 provas para serem qualificados para a próxima etapa. Cada prova possui uma nota de 0 a 5 e a nota final do time é dada pela média aritmética das 8 provas. Crie um programa em C que receba as notas individuais das provas de cada time (cada time é separado por uma quebra de linha) e determine o time vencedor.

Sugestão: Utilize uma matriz para armazenar o conjunto de notas.

Bom divertimento! :)