

Exercícios de Funções (Passagem de parâmetros por referência)

Essa lista de exercícios tem como objetivo principal desenvolver algoritmos a partir dos conteúdos abordados em sala de aula. Todos os exercícios também devem ser implementados em linguagem C. Nos programas que pedem para implementar apenas funções desenvolva também o programa principal (main) para testá-los. Não utilize variáveis globais.

1. Desenvolva um programa para testar as funções a seguir:

- Uma função que receba dois números a e b, em seguida, faça a troca destes dois números. Dica: a e b devem ser passados por referência.
- Uma função que receba dois números a e b, em seguida, decremente o primeiro e incremente o segundo. Dica: a e b devem ser passados por referência.
- Uma função que receba o raio, perímetro e área de um círculo. Em seguida, calcule e retorne nas variáveis perímetro e área o respectivo perímetro e área do círculo. Dica: perímetro e área devem ser passados por referência. Uma função que receba o lado, perímetro e área de um quadrado. Em seguida, calcule e retorne nas variáveis perímetro e área o respectivo perímetro e área do quadrado. Dica: perímetro e área devem ser passados por referência.
- Uma função que receba os valores a, b e c passados por valor, receba também dois valores x1 e x2 passados por referência. Em seguida, calcule e retorne as duas raízes da equação do segundo grau nas variáveis x1 e x2.

2. Desenvolva um programa que primeiro utilize uma função para gerar um vetor com tamanho definido pelo usuário e valores aleatórios dentro de um intervalo [min, max] determinado pelo usuário. Em seguida, uma outra função deve receber o vetor gerado, o tamanho do vetor e dois números maior e menor. Retorne nas variáveis maior e menor o menor e maior elemento do vetor. Dica1: menor e maior devem ser passados por referência. Dica2: use apenas uma estrutura de repetição para achar o menor e maior elemento.

3. Desenvolva um programa que utilize uma função para inverter os elementos de um vetor (com valores gerados usando a função do exercício anterior) sem utilizar um vetor auxiliar. Apenas o vetor fornecido à função deverá ser utilizado.

4. Desenvolva um programa para testar uma função com protótipo `void somabit(int b1, int b2, int *vaium, int *soma);`. A função recebe três bits (inteiros 0 ou 1) b1, b2 e *vaium. A função retorna um bit soma que armazena o resultado da soma dos três primeiros e o bit "vai-um" em *vaium.

5. Desenvolva um programa que leia a quantidade total de segundos e converta para Horas, Minutos e Segundos. Imprima o resultado da conversão no formato HH:MM:SS. Para isso, utilize a função com protótipo `void converteHora(int total segundos, int *hora, int *min, int *seg)`.
6. Desenvolva um programa que leia a dimensão **MxN** de uma matriz e os valores **min** e **max**. Em seguida, o programa principal utiliza uma função para gerar a matriz MxN com valores aleatórios dentro de um intervalo [**min**, **max**]. Uma outra função deve receber a matriz gerada e retornar o maior elemento da matriz, o menor elemento da matriz e o valor médio das entradas da matriz.
7. Faça um programa que calcule a pontuação de dois times após uma partida de futebol. Na função principal leia dois números inteiros que correspondem ao número de pontos dos dois times antes do jogo. Depois leia dois inteiros que representam o número de gols que cada time fez na partida. Você deverá fazer uma função separada que receba os pontos dos dois times por referência e os atualize de acordo com o resultado. Em caso de vitória são somados 3 pontos, em caso de empate é somado um ponto, e em caso de derrota, nenhum ponto é somado.

Fonte: USP - e-Disciplinas