Universidade Tecnológica Federal do Paraná – Câmpus Apucarana
Computação 2 (CP63B) - DPGR3A – Passagem de Parâmetros
Prof. Dr. Rafael Gomes Mantovani

Instruções:

- Antes de codificar, esboce em um papel a sequência de passos necessários para criar o seu programa. Isso ajuda a programar a solução;
- Crie um arquivo .c para cada um dos exercícios. Por exemplo, na resolução do exercício 01, crie um arquivo chamado 'ex01.c'.
- em todos os exercícios faça uma função `main` para testar sua função.

Exercícios sobre Passagem de Parâmetros

Exercício 1. Escreva um programa que receba um número inteiro representando a quantidade total de segundos e, usando passagem de parâmetros por referência, converta a quantidade informada de segundos em horas, minutos e segundos. Imprima o resultado da conversão no formato HH:MM:SS. Utilize o seguinte protótipo da função:

```
void converteHora(int total_segundos, int *hora, int *min, int *seg);
```

Exercício 2. Crie um vetor V de inteiro com 10 elementos. Leia 9 valores inteiros e armazene em V (a última posição ficará com lixo). Em seguida, leia um outro valor inteiro A e o coloque na primeira posição. **Atenção**: Nenhum elemento do vetor deve ser perdido, portanto, você deve deslocar os 9 elementos já inseridos em direção ao final do vetor. Por fim, imprima o vetor V. Exemplo:

- V = [1, 2, 3, 4, 5, 6, 7, 8, 9, Lixo];
- Lendo A = 85
- O resultado será: V = [**85**, 1, 2, 3, 4, 5, 6, 7, 8, 9]

O programa deverá ter no mínimo duas funções além da função `main`. Sugestão: construa as funções `imprimirVetor` e `alterarVetor`.

Exercício 3. Leia um vetor A com 5 números reais, fornecidos pelo usuário. Crie uma função para imprimir o vetor A. Crie um vetor B, também de tamanho 5, que deverá conter cada um dos elementos de A dividido pelo maior valor contido em A. Para encontrar o maior elemento de A, faça uma função que procure e retorne esse maior valor. Por fim, o vetor B deve ser mostrado utilizando a função de impressão citada no exercício anterior (`imprimeVetor`). Exemplo:

- V = [5, 3, 6, 9, 2]
- Maior valor de A é 9, logo o vetor B deverá ser:
- B = [0.55, 0.33, 0.66, 1, 0.22]

Exercício 4. Faça um programa que lê N elementos e os coloca em um vetor de inteiros. A partir desses números faça:

a) Uma função que mostre a quantidade de números pares e quais são eles;

b) Uma função que mostre a quantidade de números ímpares e quais são eles;

c) Uma função que mostre a quantidade de números negativos e quais são eles;

d) uma função para mostrar o maior e o menor número do vetor.

Exercício 5. Quadrados mágicos 3 x 3 consistem em uma matriz que a soma das colunas e das duas diagonais principais é 15. Por exemplo:

$$\begin{bmatrix} 6 & 1 & 8 \\ 7 & 5 & 3 \\ 2 & 9 & 4 \end{bmatrix} \tag{1}$$

Faça:

a) uma função para colocar os elementos dentro da matriz;

b) uma função que verificar se a matriz é um quadrado mágico;

c) uma função que imprima a matriz.

Exercício 6. Faça um programa que receba 10 números inteiros. Em seguida o programa deverá permitir o usuário buscar números dentro do vetor, informando se o mesmo está ou não no vetor. O usuário poderá realizar quantas buscas quiser e finalizar ao solicitar a busca de um valor negativo.

Exercício 7. Crie um programa que gerencie um cadastro de materiais para obras disponíveis em um fornecedor. Cada material contém: nome (tamanho máximo 40 caracteres), quantidade (int) e preço (float) de no máximo 15 produtos. O programa deve possuir um menu para que o usuário do programa consiga manipular os cadastros. Nesse menu:

a) Ao digitar (I), deve-se realizar inserção de um material na primeira posição que estiver livre;

b) Ao digitar (R), deve-se remover o material pelo índice (a ser fornecido pelo usuário);

c) Ao digitar (L), deve-se listar todos os cadastros não vazios;

d) Ao digitar (P), deve-se solicitar o nome de um material e exibir a quantidade e o preço;

e) Ao digitar (V), deve-se listar todos os produtos que tenham a quantidade igual a 0;

f) Ao digitar (S), deve sair do programa.

Obs: Cada opção do menu deve ser implementada em uma função diferente. Após cada função o menu deve ser reexibido (Exceto opção S). Não se esqueça de criar um método para gerenciar qual posição do vetor está vazia ou não.

Exercício 8. Crie um Sistema de Gerenciamento de Bandas seguindo os seguintes passos:

a) Defina uma estrutura que irá representar bandas de música. Essa estrutura deve ter o nome da banda, que tipo de música ela toca, o número de integrantes e em que posição do ranking essa banda está dentre as suas 5 bandas favoritas;

b) Crie uma função para preencher as 5 estruturas de bandas criadas no exemplo passado. Após criar e preencher, exiba todas as informações das bandas/estruturas;

c) Crie uma função que peça ao usuário um número de 1 até 5. Em seguida, seu programa deve exibir informações da banda cuja posição no seu ranking é a que foi solicitada pelo usuário;

d) Crie uma função que peça ao usuário um tipo de música e exiba as bandas com esse tipo de música no seu ranking;

e) Crie uma função que peça o nome de uma banda ao usuário e diga se ela está entre suas bandas favoritas ou não;

f) Agora junte tudo e crie um menu com as opções de preencher as estruturas e todas as opções das questões passadas.