

## **SSC0501 - Introdução à Ciência da Computação I (Teórica)**

**Professor responsável:** *Fernando Santos Osório*

**Semestre:** 2010/1

**Bolsista PAE:** Gustavo Pessin

**Monitor:** Matheus Lin (Seg. e Sexta 18h -19h)

**Horário:** Seg. 21h-22h40 e Terça 19h-20h40

**E-mail:** fosorio .at. icmc .dot. USP .dot. br

fosorio .at. gmail .dot. com

pessin .at. gmail .dot. com

matheus.lin .at. gmail .dot. com

**Web:** <http://www.icmc.usp.br/~fosorio/>

### ***LISTA DE EXERCÍCIOS - Nro. 10***

#### **[Sub-Rotinas e Passagem de Parâmetros]**

##### **[Passagem de Parâmetros por Referência]**

1. Faça um programa que leia 2 valores inteiros e chame uma sub-rotina que receba estas 2 variáveis e troque o seu conteúdo, ou seja, esta rotina é chamada passando duas variáveis A e B por exemplo, e após a execução da rotina A conterà o valor de B e B terá o valor de A.
2. Faça um programa que leia 2 valores double e chame uma sub-rotina que receba estas 2 variáveis e troque o seu conteúdo, ou seja, esta rotina é chamada passando duas variáveis A e B por exemplo, e após a execução da rotina A conterà o valor de B e B terá o valor de A.
3. Altere o programa do exercício 1 (troca inteiros) de forma a ler 4 valores em 4 variáveis: A, B, C, D. Use a rotina “troca\_valores” implementada no exercício anterior para trocar os valores de A com B, depois de C com D, mostre como ficaram os valores das variáveis, e para concluir destróque seus conteúdos mostrando novamente na tela como ficaram estas variáveis.

Valor1: <b>10</b> Valor2: <b>20</b> Valor3: <b>30</b> Valor4: <b>40</b> Trocando Valor1 com Valor2 e Valor3 com Valor4. Resultado: Valor1: <b>20</b> - Valor2: <b>10</b> - Valor3: <b>40</b> - Valor4: <b>30</b> Trocando Valor1 com Valor2 e Valor3 com Valor4. Resultado: Valor1: <b>10</b> - Valor2: <b>20</b> - Valor3: <b>30</b> - Valor4: <b>40</b>
--

4. Faça um programa que leia dois valores inteiros e chame uma sub-rotina que receba estes 2 valores de entrada e retorne o maior valor na primeira variável e o segundo maior valor na segunda variável. Escreva o conteúdo das 2 variáveis na tela.
5. Faça um programa que leia três valores inteiros e chame uma sub-rotina que receba estes 3 valores de entrada e retorne eles ordenados, ou seja, o menor valor na primeira variável, o segundo menor valor na variável do meio, e o maior valor na última variável. A rotina deve retornar o valor 1 se os três valores forem iguais e 0 se existirem valores diferentes.. Exibir os valores ordenados na tela.
6. Faça um programa baseado no exercício anterior, que ordene 3 valores fornecidos, onde deve ser usada a sub-rotina troca\_valores que foi implementada anteriormente.

7. Faça um programa que possua uma rotina para ler 2 notas (chamar uma rotina dedicada a ler 2 notas válidas e que devolver os 2 números lidos), uma outra rotina que das estas 2 notas vai calcular a média simples e a média ponderada onde a segunda nota tem peso 2 =>  $(n1 + n2 * 2) / 3$ , e uma última rotina que vai exibir na tela todos os dados que foram fornecidos e calculados: rotinas `le_notas`, `calc_media` e `exibe_dados`.
8. Faça um programa que leia 5 números. Este programa deve ter um menu que permita ao usuário escolher qual opção de cálculo que ele deseja realizar: média aritmética simples, média ponderada (ler os pesos associados a cada nota que serão informados pelo usuário), desvio padrão, maior valor e menor valor. A leitura dos 5 valores também deve ser uma das opções do menu. Exemplo:

```
>> Estatística <<
1 – Entrar com os dados (5 valores)
2 – Calcular a média aritmética simples dos dados
3 – Calcular a média ponderada dos dados (fornecer 5 pesos)
4 – Calcular o desvio padrão dos dados
5 – Achar o maior valor
6 – Achar o menor valor
7 – Sair do programa (Fim)
Entre com a sua opção: 1
Valor1: 5.3 Valor2: 8.2 Valor3: 7.3 Valor4: 3.7 Valor5: 7.1
Entre com a sua opção: 2
Média Aritmética Simples: 6.32
Entre com a sua opção: 7
FIM!
```

Obs: Desvio Padrão = Raiz quadrada da divisão do somatório do quadrado das diferenças entre valor e a média de todos os valores, dividido pelo número total de dados. Ver Wikipedia.org

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2},$$

9. Faça um programa que leia um vetor de 10 números inteiros e ordene este vetor em ordem crescente. Para ordenar o vetor, utilize uma rotina que receba dois números e retorne o menor e o maior. Esta rotina pode ser declarada da seguinte forma: `compara(N1, N2, Menor, Maior)` ou `comp_e_ord(V1, V2)`. Usando esta rotina, compare de dois a dois os números do vetor, até que o maior de todos tenha sido armazenado no final do vetor. Repita esta operação até que o segundo maior tenha sido armazenado na penúltima casa do vetor, depois o terceiro na ante-penúltima, e assim por diante até que todos os dados do vetor estejam ordenados. Este método é conhecido como o método da ordenação da bolha, ou, “*Bubble Sort*”.
10. **DESAFIO:** Faça um jogo no qual o computador desafia dois usuários para ver quem tem mais memória. O jogo deve gerar um número de 0 à 9 e apresentar ao jogador 1 durante 5 segundos. Depois deve repetir este procedimento para o jogador 2. Em seguida, ele deve pedir ao jogador 1 para informar o número escolhido, e após pedir ao jogador 2 para fazer a mesma coisa. Na segunda rodada, o computador deve gerar mais um número entre 0 e 9, que será “concatenado” à direita do primeiro número. Então repetiremos o procedimento de exibição e teste de memorização para os dois usuário. O programa deve continuar adicionando números (casas adicionadas ao final do número) até que um dos dois jogadores não consiga mais memorizar perfeitamente o número. Para concluir, o programa deve informar qual dos dois jogadores ganhou, quantas casas foram memorizadas ao total, e em que casa foi que o jogador perdedor errou (primeira casa errada começando da esquerda para a direita). Dicas: procure desenvolver o programa em módulos; armazene a sequência de números dentro de uma string. => Seu programa ficou bem modular? Seria fácil de adaptá-lo para 3 ou mais jogadores?

## TESTE DE MESA:

11. Procure *prever o comportamento do programa* abaixo e depois teste o programa na prática. Indique qual é o valor das variáveis A, B e X, Y que é escrito na tela no final da execução do programa.

A: \_\_\_\_      B: \_\_\_\_  
X: \_\_\_\_      Y: \_\_\_\_

---

---

Programa:

```
#include <stdio.h>
#include <conio.h>

double x, y;

troca_ab(a,b)
int a,b;
{
    int temp;

    temp=a;
    a=b;
    b=temp;
}

main()
{
    int a,b;

    a=10;
    b=20;

    troca_ab(b,a);

    if ((a > 5) && (b < 15))
    {
        x = a / 3 + b / 2;
        y = a;
    }
    else
    {
        x = a;
        y = (a > b);
    }

    printf("A=%d - B=%d\n",a,b);
    printf("X=%.2lf - Y=%.2lf\n",x,y);

    getch();
}
```

12. Procure *prever o comportamento do programa* acima, considerando que a rotina `troca_ab` foi alterada conforme indicado abaixo. Depois teste o programa na prática. Indique qual é o valor das variáveis A, B e X, Y que é escrito na tela no final da execução do programa.

A: \_\_\_\_      B: \_\_\_\_  
X: \_\_\_\_      Y: \_\_\_\_

```
troca_ab(a,b)
int *a,*b;
{
    int temp;
    temp=*a;
    *a=*b;
    *b=temp;
}

...

    troca_ab(&a,&b);

...
```