

## **SME0230 - Introdução à Programação de Computadores** **Primeiro semestre de 2010**

**Professora:** Marina Andretta (*andretta@icmc.usp.br*)

**Estagiário PAE:** Mayron César de Oliveira Moreira (*mayron@icmc.usp.br*)

**Monitor:** Luiz Carlos Lucca (*lluca@icmc.usp.br*)

**Forma de entrega:** O exercício desta aula prática deverá ser entregue por e-mail para *andretta@icmc.usp.br*, com cópia para *mayron@icmc.usp.br* e *lluca@icmc.usp.br*.

O nome do arquivo deverá ser **IPC\_ap10\_númeroUsp\_1.c**. Exemplo:  
**IPC\_ap10\_6711776\_1.c**.

O título da mensagem será: **IPC\_ap10\_númeroUsp**. Exemplo:  
**IPC\_ap10\_6711776**.

**Atenção:** Mandar **apenas 1 email** com o código em **anexo**.

### **Observações importantes:**

1. Trabalhos entregues após a 23h59 do dia 07/06 terão 10% a menos na nota;
2. O aluno que enviar o trabalho fora do padrão ou que não enviar para os e-mails especificados no enunciado terá 20% a menos da nota referente ao exercício desta aula. Caso na próxima aula o aluno entregue o trabalho novamente fora do padrão ou não envie para os e-mails especificados no enunciado, o mesmo receberá zero como nota do referente exercício da aula prática.

## **Aula Prática 10 – Funções e estruturas**

### **Exercício: Vamos otimizar o serviço da Secretária?**

Este exercício consiste em melhorar a qualidade de vida da secretária da biblioteca do ICMC. Como existem muitos alunos, é muito difícil encontrar uma ficha de um aluno no arquivo, pois as fichas encontram-se embaralhadas. A idéia é criar um programa que dado um identificador de aluno, o programa retorne a identificação da ficha no arquivo.

Crie uma estrutura de Dados que contenha uma **chave** de busca e quatro campos pessoais (**nome, endereço, telefone, idDaFicha**). Nome e endereço são strings, de no máximo 15 caracteres; telefone, idDaFicha e chave são inteiros.

**Seu programa deve conter um vetor de registros do tipo acima.**

Por Exemplo:

- Alunos:
  - 0
  - Luiz Carlos Lucca
  - Rua dos Pombos
  - 2345678
  - 00
  
- 1
- Mayron Moreira
- Rua do Lago
- 2345667
- 01

## Primeira Parte:

- Faça uma função que dado um vetor de registros, os ordene de maneira crescente, de forma que o primeiro registro do vetor seja o menor e o último o maior. Para isso, **utilize a chave de cada registro**; No caso do exemplo, o aluno Luiz seria o primeiro do vetor (**chave 0**) e o aluno Mayron seria o segundo (**chave 1**).
  - Dica para este algoritmo (InsertionSort):
    - Comece pelo segundo elemento do vetor; Compare com o primeiro; se menor, troque de posição.
      - A partir do terceiro:
        - Compare o elemento com o anterior, se menor, troque, até que não seja mais possível;

Exemplo:

Compara o 7 com o 7

7 1 7 1 9 1 4 1 2 1 8

Compara o 9 com o 7, 7

7 1 7 1 9 1 4 1 2 1 8

Compara o 4 com o 9, 7, 7

7 1 7 1 9 1 4 1 2 1 8

Compara o 2 com o 9, 7, 7, 4

4 1 7 1 7 1 9 1 2 1 8

Compara o 8 com o 9, 7, 7, 4, 2

2 1 4 1 7 1 7 1 9 1 8

**A ordem final é:**

2 1 4 1 7 1 7 1 8 1 9

Link: <http://wikipedia.artudi.org/Insertion%20Sort.php>

## Segunda Parte:

- Com o vetor já ordenado, faça uma função que procure uma chave nesse vetor, utilizando a seguinte lógica: (este algoritmo chama-se busca binária)
  - Pegue a posição do primeiro e do último elemento
  - Pegue o meio do vetor
  - Compare a chave que estamos procurando com a posição do passo acima. Veja se o elemento procurado é maior ou menor.
    - Se menor, faça o início receber esta posição, senão faça o fim receber esta posição.
  - Repita até que o início não ultrapasse o fim, ou vice e versa.

- Exemplo (considerando apenas as chaves) : (procurando o 8)

**0      2      4      6      8      10**

Pós Início: 0

Pós fim: 6

Posição do meio: 3;  $(0+6)/2$

Elemento do meio: 6; 6 é menor que 8 => pós início =3

0 2 4 6 8 10

Pós Início: 3

Pós fim: 6

Posição do meio: 4;  $(3+6)/2$

Elemento do meio: 8; Achou.

Tome como entrada para o seu programa as seguintes chaves (as outras informações ficam por conta da sua imaginação):

2 9 6 5 1 0 18 3 25 12 8 99 65 58 43 37 20.

Utilize o algoritmo de busca para localizar as chaves:

- 19
- 2
- 20
- 100
- 58
- 67

A saída deve ser da seguinte forma: **Número, presente no vetor**; ou, **Número, não esta presente no vetor**; seguido **da localização da ficha** na prateleira

Ex:

- 19: Não está presente no vetor;
- 2: Está presente no vetor; Localização da ficha: 00889.