

# SME0330 - Introdução à Programação de Computadores

## Primeiro semestre de 2015

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

**Estagiário PAE:** Luzia de Menezes Romanetto (luzia@icmc.usp.br)

**Monitor:** Matheus Cabral Manoel (matheus.manoel@usp.br)

## Exercícios de laboratório 10

**Data:** 01/06/2015.

**Data máxima de entrega:** 05/06/2015, até às 23h59min. Trabalhos entregues fora do prazo não serão aceitos.

**Forma de entrega:** Os exercícios deverão ser entregues por e-mail para exercicios.sme0330@gmail.com e o título do e-mail deverá ser IPC2015\_Ex10. No início do arquivo deve haver um comentário com o nome e o número USP do aluno.

### Exercício 1

Escreva um programa em linguagem C que, dada uma *string* *s*, conte a frequência de todos os caracteres presentes em *s*.

Por exemplo, se a entrada do programa for a *string*

```
ccbbbbaaccaaz
```

a saída do seu programa deverá ser

```
c 4
b 4
a 4
z 1
```

Para facilitar o trabalho, você deve seguir o seguinte guia:

- Crie uma *struct* chamada **node** com um campo **element**, para guardar um caracter, e um campo **count**, para guardar a quantidade de vezes que o caractere apareceu na *string*.
- Inicialmente crie um vetor de ponteiros do tipo **\*node** (ou seja, um vetor de ponteiros para **node**). Tal vetor deve ter o tamanho necessário para guardar os diferentes caracteres presentes na *string*.
- Ao ler um caractere você deve checar se este caractere já apareceu anteriormente ou não.
  - Caso seja a primeira ocorrência deste caractere na *string*, você deve alocar uma estrutura **node** dinamicamente e apontar um dos ponteiros disponíveis do vetor para a memória alocada. Ao alocar tal estrutura, seus campos **element** e **count** devem ser inicializados com o caractere encontrado e 1, respectivamente.
  - Caso o caractere já tenha ocorrido na *string* anteriormente, adicione 1 ao contador correspondente àquele caractere.
- O objetivo é que ao final seja impresso na tela a tabela de caracteres com a contagem de ocorrência destes na *string* fornecida.

## Exercício 2

Duas palavras são anagramas se elas contiverem a mesma frequência de letras (por exemplo: anagrama, amargana  $\rightarrow$  {a 4, m 1, r 1, g 1, n 1}).

- a) Faça uma função, em linguagem C, que compare duas tabelas de frequência, retornando 0 se forem diferentes, e 1 se forem iguais.
- b) Faça um programa, em linguagem C, que recebe duas *strings* e decida se elas são anagramas.