

ESTRUTURA DE DADOS E ALGORITMOS

PROVA 1 – MAIO DE 2012

PROF. RENATO CAMPOS MAURO

1) Considere a seguinte estrutura para implementar uma lista simplesmente encadeada com nó cabeça:

```
typedef struct _Lista Lista;
typedef struct _NoLista NoLista;

struct _Lista
{
    NoLista *primeiro;
};

struct _NoLista
{
    int matricula;
    char nome[40];
    float salario;
    NoLista *proximo;
};
```

Baseado nestas estrutura escreva cada uma das funções abaixo:

a) NoLista* PesquisaPorMatricula(Lista *l, int matricula)

Esta função deverá retornar o ponteiro para NoLista que contem um registro dado a matrícula, ou NULL, caso o elemento não se encontre na lista. Não é para usar printf. É para retornar o Nó.

b) NoLista* PesquisaPorNome(Lista *l, char* nome)

Esta função deverá retornar o ponteiro para NoLista que contem um registro dado o nome, ou NULL, caso o elemento não se encontre na lista. Não é para usar printf. É para retornar o Nó. Use strcmp para comparar as strings.

c) void ImprimeNomeMaiorSalario(Lista *l)

Esta função deverá imprimir o nome da pessoa que tiver o maior salário da lista. Se a lista estiver vazia, imprimir a mensagem LISTA VAZIA. Não é para retornar nada nesta função. Usar printf para imprimir apenas o nome do registro correspondente.

d) void OrdenaSalario(Lista *l)

Reorganizar a lista, aplicando o método selection sort para ordenar os elementos da lista. Não retornar nem imprimir nada.

2) Dado a sequência 10 8 4 7 6 1 2 9 5, como será a reorganização destes elementos usando o método de particionamento dado em aula (escolhendo o último como pivô)? Logo após o 1º particionamento, quais serão os subvetores passados como parâmetro para a primeira chamada recursiva subsequente?

3) Dado uma cadeia de caracteres (string), escreva esta string de trás para frente usando recursividade. Considere o protótipo dado abaixo. A string é terminada com o caracter zero 0.

```
void imprimeContrario(char *str)
{
}
```