

## Prova Final

Considere os seguintes tipos predefinidos para a prova.. Responda a todas as questões em linguagem C.

<pre>typedef struct tipoNo {   int dado;   struct tipoNo *prox; } tipoNo;</pre>	<pre>typedef struct tipoLista {   tipoNo *prim; } tipoLista;</pre>	<pre>typedef struct {   char nome[30];   char endereco[20];   float altura; } tipoPessoa;</pre>
---	--	---

1 – Faça uma função que receba como parâmetro um arquivo já aberto para leitura e escrita (recebendo o descritor do arquivo), contendo registros de dados do tipoPessoa. A função deve ordenar o arquivo com o melhor algoritmo de ordenação dentre as opções: seleção, inserção, quicksort ou mergesort. Você tem as seguintes informações disponíveis sobre o problema: i) Os dados estão muito embaralhados e com nenhuma repetição de chave; ii) O conteúdo do arquivo cabe completamente na memória principal do seu computador, apesar de não haver memória suficiente para duas vezes o seu conteúdo; ii) há mais de 10 milhões de itens no arquivo. Implemente o algoritmo de ordenação dentro de sua função. Se você souber qual o algoritmo usado e não souber implementá-lo, indique o algoritmo correto para ganhar 1 ponto. A chave de ordenação deve ser o nome da pessoa. (2,5 pontos)

2- Faça uma função que retorne verdadeiro caso duas listas encadeadas passadas como parâmetro tenham interseção vazia (nenhum elemento, campo “dado”, em comum). A função deve retornar falso em caso contrário. Use o tipoLista definido acima. (2,5 pontos)

3 – Faça uma função que receba como parâmetro um arquivo e imprima na tela o número de vezes que dígitos entre ‘0’(valor 48) e ‘9’ (valor 57) ocorrem dentro do arquivo. A função deve imprimir na tela a frequência de cada um desses dígitos. Note que a função não tem valor de retorno. (2,5 pontos)

4 – Um programador fez um programa onde colocou dados de pessoas em um vetor (vetor com elementos do tipoPessoa). Agora ele quer ajuda para fazer uma função que grave os dados do vetor em um arquivo em formato texto, de maneira que qualquer pessoa que abra o arquivo em um editor de textos possa ver claramente os dados de cada pessoa registrada no vetor. Faça uma função que receba como parâmetros um vetor de pessoas e seu tamanho. A função deve gerar como resultado o arquivo solicitado, contendo os dados de pessoas (nome, endereço e altura) em linhas separadas. Deixe uma linha em branco separando cada registro. O nome do arquivo a ser criado é “saida.out”. Lembre-se de deixar o arquivo fechado ao final da função. (2,5 pontos)