

3ª Lista de Algoritmos e Estruturas de Dados II

Prof. Rone Ilídio

1) Crie um programa que gera arquivos textos contendo vários números aleatórios, os quais serão utilizados para a ordenação a seguir. Ao executar, tal programa deve pedir a quantidade de números que devem ser gerados e criar um arquivo chamado numeros.txt.

2) Crie um programa para exibir os tempos de execução dos métodos de ordenação: Seleção, Inserção, Mergesort e Quicksort. Teste-os com vetores de inteiros de tamanhos 1000, 5000, 10000 e 100000. Os dados desses vetores devem ser lidos a partir arquivos textos. Não considere os tempos para leitura dos arquivos texto. Pesquise por formas de se calcular o tempo de execução de algoritmos. Essa pesquisa é parte da lista do exercício.

3) Crie um programa que leia um arquivo texto contendo nomes de pessoas, ordene esses nomes em ordem alfabética e escreva o resultado em outro arquivo texto. Os arquivos texto deverão estar no seguinte formato:

José Maria da Silva
Michael Jackson de Souza
Mick Jagger de Oliveira
Justin Bieber da Costa
Lady Gaga de Oliveira e Silva

4) Implemente o método *Mergesort* que ordene um vetor do tipo *produto* a partir do campo nome. Tal tipo é definido a seguir.

```
struct produto{  
    string nome;  
    double preco;  
};
```

Crie um programa que recebe dados do usuário, preencha o vetor e teste a ordenação.

5) Implemente um programa que leia um arquivo texto com o seguinte formato:

Mané Tapado da Silva;100
Manuel Fernandes;98
Maria das Graças;50

Ana Maria Bragis;100
João Paulo Premero;40
Pedro Paulo Pedroza;30

Tal arquivo contém os nomes e as notas dos alunos de uma turma. Ao final, tal programa deve exibir em ordem alfabética os alunos que passaram e os que foram reprovados. Considere reprovados os alunos com nota menor ou igual a 60. Utilize o Quicksort.

6) Uma prefeitura possui um arquivo texto (denominado empresas.txt) contendo os nomes das empresas que estão em dívida. Tal arquivo possui em cada uma de suas linhas o nome de uma empresa, o valor devido referente ao ano de 2011 e o valor devido referente ao ano de 2010 (tudo separado por espaço). A seguir um exemplo desse arquivo:

Maicrosofiti 100 200
Ibeeme 30 20
Horacol 0 200
Sançungui 30 20

Crie um programa para ajudar o setor de cobrança da referida prefeitura. Tal programa deve buscar os dados no arquivo texto, inseri-los em um vetor e oferecer o seguinte menu para o usuário:

Digite:
1-Busca dívida por nome da empresa
2-Exibe dívidas
3-Sair

Se o usuário escolher a opção 1, o programa deve pedir o nome da empresa, buscar e exibir a dívida dessa empresa. Se a empresa não for encontrada, uma mensagem deve aparecer para o usuário. O valor da dívida é igual à soma das dívidas de 2010 e 2011. Se o usuário escolher a opção 2, o programa exibir todos os dados das empresas ordenados pelo nome da empresa. A opção 3 termina o programa imediatamente.