

Universidade Federal de Lavras

Disciplina: Estrutura de Dados – GAC108

Professor: Renato Ramos da Silva

Alunos:

Alex Cyrillo de Sousa Borges

Gabriel Furtado Teixeira

Mardem Arantes de Castro

Relatório do projeto prático - Etapa 2

I. Introdução

O trabalho nos permitiu adquirir maior conhecimento a respeito de manipulação de arquivos tipados e na ordenação dos mesmos, com foco maior nessa. Composto pelo arquivo principal de ordenação, ele lê os trechos do arquivo binário e a partir daí ele ordena de forma decrescente usando o nome e o id, respectivamente.

II. Objetivo

Esta segunda etapa do projeto prático tem como objetivo principal a implementação de um sistema que realize a ordenação do arquivo binário criado na primeira etapa, a partir do conversor de csv, de forma decrescente utilizando o nome cadastrado e, caso exista mais de um nome igual, utilizando o ID do usuário para realizar essa ordenação.

III. Programa para a ordenação do binário

O programa é utilizado para realizar a ordenação do arquivo binário recebido, neste caso "san_francisco_payroll_dataset.bin". Ele pode ser dividido, basicamente nas funções de Quicksort, compostas por trocar, particao e quicksort, e a função main, que possui algumas funções auxiliares para melhor visualização do funcionamento do programa, sendo a principal um menu, no qual é possível escolher as opções de ordenar o arquivo binário, imprimir e por fim encerrar a execução do programa.

O objetivo principal, que é a ordenação, ocorre quando se seleciona a opção ordenar no menu. Após escolhida essa opção a função recursiva principal do quicksort é chamada. Esse método de ordenação foi escolhido devido à sua fácil

implementação e sua eficiência quando lidando com grande número de dados, ao contrário dos métodos bubble sort e selection sort. Seu funcionamento depende da escolha de um pivô, podendo ser considerado "ideal", que divide o arquivo em duas partes iguais, ou próximo a isso. Para a escolha do pivô, escolhemos sempre o último elemento do arquivo, uma tática padrão utilizada em algoritmos quicksort. A partir disso, separamos o arquivo em duas partes, antes e depois do pivô, onde buscamos que a parte anterior apresente elementos sempre menores que esse e a parte posterior, elementos maiores que ele. Isso ocorre dentro da função particiona que chama a função troca para realizar a alteração de posição. Dessa maneira, o algoritmo funciona de forma recursiva, chamando a si mesmo, até que o arquivo esteja ordenado.

IV. Conclusão

Esta etapa do projeto prático, que atua como estudo autônomo, foi de grande importância para que continuemos nos familiarizando com o estudo sozinho, sem o acompanhamento de professores, e ainda melhorou nossos conhecimentos em métodos de ordenação. Apresentou-se algumas dificuldades na questão da implementação, pois a ordenação apresentava diversos erros que foram corrigidos após estudos e paciência no entendimento do mesmo. Porém apesar disso, foi uma experiência que permitiu mais um contato com o trabalho em grupo e divisão de funções. O prazo em questão continuava bem apertado, já que havia a necessidade de estudo constante todas as semanas para as demais disciplinas e os assuntos recorrentes que apareciam na disciplina Estrutura de Dados, que se tornam cada vez mais complexos e necessitam de mais dedicação para serem compreendidos. Dessa forma, o grupo entende que o trabalho é imprescindível na formação profissional dos participantes e para adquirir experiência em ambiente de trabalho.