



BIANCA LIMA
JÚLIA RIBEIRO
LARISSA ROCHA

PROJETO PRÁTICO

LAVRAS-MG

2023

SUMÁRIO

1. INTRODUÇÃO	1
2. ESTRUTURA DO PROJETO	2
3. PRINCIPAIS ERROS AO LONGO DO DESENVOLVIMENTO DO PROJETO	4
4. CONCLUSÃO	5

1. INTRODUÇÃO

Esse relatório tem como objetivo descrever e abordar dados sobre o projeto prático de Introdução aos Algoritmos. O projeto prático tem como seu principal objetivo implementar um sistema de cadastro que contém recursos para busca e alteração de dados, com armazenamento dos dados em arquivo binário. O tema escolhido foi “Ganhadores do Oscar” e conta com uma estrutura de seis variáveis, sendo quatro de tipo textual (nome do ganhador, modalidade, nome do filme e país do ganhador) e duas de tipo numérico (ano da premiação e duração em minutos do filme).

2. ESTRUTURA DO PROJETO

Em primeiro lugar, foi definida a estrutura “ganhadores” que armazena informações sobre os ganhadores do Oscar. Ela é utilizada como padrão em todos os procedimentos do código. Logo em seguida, variáveis são inicializadas, ocorre a importação do arquivo CSV para binário e são lidos os dados do arquivo binário para preencher o vetor de registros chamado “lista”. É oferecido ao usuário as opções de remover (digitar 1), inserir (digitar 2), buscar (digitar 3), imprimir arquivos (digitar 4), exportar arquivo binário para CSV (digitar 5) ou ordenar arquivo (digitar 6).

Se o usuário escolher a opção 1, é solicitado o índice que ele deseja remover e o procedimento “remover” é chamado. Esse procedimento recebe um vetor de ganhadores, seu tamanho (tam), e um índice (remove) que indica a parte do registro que deve ser removida, depois utiliza um looping para verificar se o índice atual é igual ao índice a ser removido, se a condição for verdadeira a duração do filme é negativada para indicar uma remoção marcada. Em seguida, a função “GravarBinario” é chamada para realizar inserção do vetor de registros modificado no arquivo binário e uma mensagem é exibida indicando que o índice foi marcado com negativo.

Caso o usuário tenha optado por “Inserir dados”, é verificado se existe no arquivo algum valor negativado na variável “lista[entrada].duracaoMin” para inserir o novo dado nesse índice do vetor. Caso contrário, o dado é inserido no fim do arquivo. Ambos utilizam os procedimentos “void dadosNovos” e “void GravarBinario” para, respectivamente, pedir entrada do usuário sobre os elementos a serem inseridos e escrever no arquivo “oscar.dat” o vetor de registros completo e com as alterações. Ao final, uma mensagem impressa na tela comunica ao usuário que a inserção foi feita com sucesso.

Se o usuário escolher a opção 3, ele deverá escolher se irá fazer a busca por ano de premiação (digitar 1) ou por nome do ganhador (digitar 2). Se o usuário escolher buscar por ano de premiação será chamada o procedimento “void buscarAno” que tem a função de buscar e imprimir os registros cujo ano de premiação seja igual ao ano fornecido pelo usuário. Caso não se encontre o elemento da busca, uma mensagem é impressa na tela comunicando o usuário. Caso a opção escolhida seja a busca pelo nome do ganhador, o código apresenta getline para ler toda a linha de entrada e armazena o nome do ganhador que foi fornecido pelo usuário na variável “buscarGanhador”. Depois, é iniciado um looping que percorre todos os registros na lista de ganhadores e usa strcmp para comparar o nome que foi fornecido com o nome do ganhador no registro atual. Se os nomes forem iguais, é impresso o registro inteiro correspondente ao índice encontrado na comparação. Caso não se encontre o elemento da busca, uma mensagem é impressa na tela comunicando o usuário.

Se o usuário escolher a opção 4 como primeira entrada é apresentada outra opção de input para imprimir o arquivo todo ou apenas um trecho. Caso opte por imprimir trecho, o programa abre espaço para o usuário digitar dois inteiros como início e fim que serão usados dentro de um looping para chamar o procedimento nos índices de vetores indicados pelo usuário. O procedimento utilizado para imprimir na tela é “void imprimir”.

Caso o usuário opte pela opção 5, é chamado o procedimento “void CSV” que exporta dados do arquivo binário para o formato CSV de nome “novo_oscar.csv”. Ao final, é impresso na tela uma mensagem que confirma a exportação.

Supondo que o usuário escolha a opção 6, o código redirecionará para a estrutura de ordenação crescente quick sort com base em dois campos diferentes, considerando os anos das premiações (opção interna 1) ou os nomes dos ganhadores (opção interna 2). Mas antes que o usuário possa escolher as opções internas, o algoritmo faz a cópia para a estrutura “ganhadores” de nome “lista2” daqueles que não estão marcados com chave negativa. Depois é copiado o conteúdo do vetor de registros “lista2” para o vetor “lista” e assim é possível excluir definitivamente os registros marcados com chave negativa pelo procedimento “void remover”. Explorando o processo de ordenação, foi utilizado uma função de particionamento e um procedimento com a estrutura do quick sort para cada campo que o usuário deseja buscar. Nas funções de particionamento são definidos os pivôs, no caso da variável do tipo string é necessário fazer o uso do vetor de char e da cópia por strcpy, após isso, são definidas as posições iniciais e os processos de comparação dos elementos se iniciam. Os elementos menores que os pivôs são movidos para os lados esquerdo dos vetores das funções, os elementos maiores para os lados direito, e a posição do pivô é retornada. Já as estruturas do quick sort são bem simples e ordenam de forma crescente os vetores dos lados esquerdo e direito separadamente, anteriormente particionados, utilizando recursão.

Caso a primeira entrada do usuário no programa seja um número diferente do estabelecido, é retornada a função main() e uma mensagem de aviso é exibida na tela.

3. PRINCIPAIS ERROS AO LONGO DO DESENVOLVIMENTO DO PROJETO

1- Erro de leitura do vetor de char, pois o número de caracteres real estava maior do que o descrito no código.

2- Ao executar o código após redimensionamento de vetor e inserção de dado no fim do arquivo, o código seguia imprimindo apenas até a posição antes do redimensionamento. Foi necessário incrementar a variável “tam” (inicializada com zero) toda vez que lia um vetor de registro do arquivo para saber exatamente o tamanho do vetor e assim imprimir utilizando looping for.

3- Erro na entrada de vetor de char com espaço, pois não utilizamos `cin.ignore()`.

4- Erro ao comparar string com char, foi preciso usar a função “.c_str()” da biblioteca `<cstring>` para transformar a string em char e assim fazer comparação por meio da função “`strcmp()`”.

4. CONCLUSÃO

Em resumo, este relatório teve o propósito de esclarecer o código elaborado pela equipe, abordando a lógica subjacente a cada função/procedimento apresentada. Durante a análise, identificamos e exploramos os pontos fundamentais do código, aproveitando os erros como oportunidades para aprender e aprimorar o projeto. Os desafios enfrentados ao longo do desenvolvimento não apenas nos permitiram corrigir falhas, mas também forneceram valiosos insights para compreender melhor o funcionamento do projeto como um todo. Por fim, a base de dados (que contém 50 itens cadastrados) se encontra em arquivo CSV e é exportada para binário assim que o programa se inicializa, além disso o código-fonte é capaz de performar todas as etapas exemplificadas na descrição do projeto prático da disciplina.