Trabalho de Introdução aos Algoritmos

Lucas Antônio Lopes Neves

Luiz Carlos Coelho Conde

Thiago Luigi Gonçalves Lima

Relatório

Tema Escolhido

O tema escolhido foi *games*.

Como foi realizado o projeto

O grupo decidiu utilizar a plataforma G*itHub* para gerenciar o projeto. Com ela fomos capazes de ter um controle além de trabalhar em equipe vendo cada alteração feita no arquivo do trabalho.

O *link* para o acesso do nosso projeto no *GitHub* é esse [*aqui*](https://github.com/lucas54neves/games).

Uso do Registro

Como nosso grupo falaria sobre games e teríamos suas subcategorias (Nome, Ano de lançamento, Plataforma e Descrição) montamos um registro (*struct*) chamado *game.*

Função *main*

Na função principal (*main*) colocamos a seguinte linha *setlocale(LC\_ALL, “Portuguese”);* para que fosse possível utilizarmos palavras do idioma português na saída padrão *(cout)* para os menus e qualquer comunicação com o usuário.

Declaramos a variável *dadosGame* do tipo *game* e criamos um ponteiro pois um dos requisitos do trabalho era a utilização de variável dinâmica.

Criamos um laço *(while)* que é o nosso menu principal e ele funciona enquanto a opção escolhida pelo usuário não for *0* (que é a opção de fechar o programa). Para que o laço fosse inicializado, declaramos a variável opção com o valor 1.

Utilizamos o *system(“clear”)* para limpar a tela toda vez que se abre uma opção do menu.

No menu, damos ao usuário cinco opções e pedimos a ele que digite um número referente à opção desejada. De acordo com a opção selecionada pelo usuário, usamos um *switch case* para chamar as funções da opção desejada.

Função *lerArquivo*

Decidimos utilizar uma função apenas para a leitura dos dados. Após confirmar com a professora Juliana de que era permitido ler o arquivo em uma função, desde que os dados lidos não sejam trabalhados nessa função, decidimos que assim seria feito para uma melhor legibilidade do código.

Essa função pega o tamanho do arquivo, descobre a quantidade de jogos e lê jogo por jogo, colocando na variável bloco e salva a quantidade de jogos que o arquivo possui por parâmetro por referência.

Função print

Para uma melhor legibilidade do código, decidimos criar uma função específica para a impressão dos jogos cadastrados. Essa função *print* imprime o vetor *dadosGame[]* com o tamanho exato de jogos cadastrados.

Função insercaoDados

Na função para inserção de dados, usamos as funções da linguagem *cin.ignore()* e *cin.getline(variável, capacidade do vetor char)* para o problema ler frases ou nomes compostos sem que os espaços atrapalhem a leitura.

Função atualizacaoDados

Para atualizar os dados, criamos um função que usa um número de identificação para selecionar o jogo desejado para atualizar. Esse número foi chamado de *ID*. Além disso, criamos um controle de confirmação antes da atualização de fato dos dados.

Função buscaDados

Para a busca de um ou mais jogos cadastrados, utilizamos uma função que busca os jogos através do ano. É solicitado ao usuário o ano e o programa imprime os jogos encontrados ou a mensagem de *“Nenhum dado encontrado.”.*

Função ordenacaoDadosCadastrados

Como o método de ordenação Bubble Sort é o método de melhor compreensão entre os membros do grupo, decidimos utilizar este método para a ordenação crescente com o ano como referência.

Considerações Finais