

AT2/N2 - Atividade prática individual 2

Crie um código que seja capaz de ler uma lista de artistas e suas respectivas músicas de um arquivo (*musicas.txt*) e armazenar os registros lidos em uma **lista circular duplamente encadeada** durante a execução. Cada linha do arquivo contém um registro, composto por um artista e o nome de uma música, separados por ponto e vírgula.

Inicialmente, o código deve ler o arquivo (que será enviado junto com esta descrição da atividade) e criar um “simulador” de *playlist* com as músicas do arquivo. Dessa forma, o código deve exibir a música atual da playlist e ser capaz de avançar ou retornar a alguma música. A solução também deve mostrar um menu interativo, pelo qual o usuário será capaz de executar as seguintes ações:

1. Exibição da *playlist* pela ordem de cadastro;
2. Exibição da *playlist* ordenada pelo nome das músicas;
3. Inserção de novas músicas;
4. Remoção de uma música;
5. Busca por uma música;
6. Avançar para próxima música;
7. Retornar a música anterior;

As funcionalidades de inserção e remoção devem aplicar as mudanças no conteúdo do arquivo também, ou seja, um artista inserido deve ser colocado tanto na lista alocada pelo código quanto no arquivo de entrada.

Arquivo

O arquivo de entrada é composto de registros formados pelo nome do artista e uma música, separados por “;”. Ele será divulgado pelo professor previamente. Exemplo de registros:

Artista X;Música 3
Artista Y;Música 42

Critérios de avaliação

- **Implementação das funcionalidades:** 1.4 (0.2 por funcionalidade);
 - **Correto funcionamento e usabilidade** das funcionalidades;
 - Uso adequado da estrutura de dados;
- **Estruturação e organização do código:** 0.5;
 - Organização/clareza do código (nomes significativos de variáveis, indentação, etc.);
 - Uso de funções;
 - Uso de arquivo .h;
- **Organização do GitHub (README, comentários e tamanho dos commits, etc.)** (0.1);

Soluções com erros de sintaxe (falta de “;”, nome de funções ou tipos escritos errados, etc.) ou de execução (variáveis indefinidas, escolha do identificador de tipo inadequado, etc.) serão penalizadas e receberão nota 0. **Portanto, testem o código antes de enviá-lo.**

A data de entrega da lista é dia 02/06, até às 23:59. O envio deve ser feito pelo AVA e deve conter um link para uma pasta do GitHub contendo o código que soluciona o problema abaixo.