



1º Período – Algoritmos e lógica de programação

Professor Gilberto Viana de Oliveira

PROJETO 2 – 20 pontos – Grupos de 3 a 5 pessoas

1. DESCRIÇÃO

Um empresário deseja contratar você para auxiliá-lo no desenvolvimento de um app para o controle dos usuários de seu sistema de músicas. A ideia do sistema é armazenar os dados músicas e bandas, onde algumas regras deverão ser respeitadas. Inicialmente, é necessário que o sistema possua 2 estruturas para seu funcionamento. Bandas e músicas.

Cada **banda** possui um nome, um id único, um gênero e um ano de criação.

Cada **música** possui um nome, uma nota e um compositor.



1.1 FUNCIONALIDADES OBRIGATÓRIAS

As funcionalidades descritas a seguir compõe a parte mais importante do sistema. Todas elas devem estar dispostas em um menu.

- 1.1.1 **Cadastrar Banda** – deve ser possível cadastrar uma banda, com todos os seus dados. Quando a banda for cadastrada, ela não deve ter nenhum valor em branco. Caso o usuário não preencha algum dado, o programa deve continuar pedindo até que todos os dados sejam preenchidos. Podem ser cadastrados no sistema até dez bandas diferentes.
- 1.1.2 **Cadastrar Músicas** – deve ser cadastrado uma nova música, a nota inicial dessa música deve ser zero, mas sempre que alguém avaliar a música, a sua nota deve ser a média de todas as pessoas que o avaliaram. Exemplo: A música “Evidencias” começa sem nota, se 1 pessoa o avaliar com nota 3, sua nova nota passa a ser 3. Se uma segunda pessoa o avaliar com nota 5. Sua nova nota atualizada deverá ser 4. Podem ser cadastrados no sistema 10 músicas diferentes.
- 1.1.3 **Imprimir todas as bandas cadastradas** – Deve imprimir apenas os dados de bandas que foram preenchidos, dados não inicializados como lixo de memória deverão ser descartados. Exemplo: Se apenas 3 bandas foram cadastradas, o sistema só deve imprimir os dados das 3 bandas.



- 1.1.4 **Imprimir a música com nota mais alta** - Deve imprimir os dados da música com a nota mais alta até o momento.
- 1.1.5 **Imprimir a música com nota mais baixa** - Deve imprimir os dados da música com a nota mais baixa até o momento.
- 1.1.6 **Imprimir bandas de um gênero** – Deve mostrar todos os nomes das bandas de um gênero.

1.1.7 Funcionalidade – AVALIAR UMA MÚSICA

O programa deve ter uma função “Avaliar música”. Quando essa função é escolhida, deve aparecer o nome de todas as músicas cadastradas para que o usuário escolha qual música quer avaliar. Em seguida, ele coloca uma nota que deve ser entre 0 e 5, o valor deve ser um número real e não pode estar fora desse intervalo. A nota média da música deve ser atualizada com base na nota anterior e na nota dada pelo usuário.

- 1.1.8 **Sair do programa** – Deve encerrar o programa de maneira satisfatória (sem erros).

2. ENTREGA DO TRABALHO

- O trabalho deverá ser entregue até o dia 04/07/2020 às 23:59 horas. Entrevistas acontecerão a na segunda feira a partir das 19 horas.
- O trabalho deve ser entregue via *google classroom*.
- O código deve ser enviado no formato de código fonte (arquivo.c).
ATENÇÃO: TRABALHOS ENVIADOS NO FORMATO .cpp RECEBERÃO NOTA ZERO.
- Todas as funcionalidades do trabalho devem ser desenvolvidas utilizando o conceito de modularização. Ou seja, todas as funcionalidades devem estar em funções.
- Arquivos com problemas de compilação receberão nota zero. O professor não receberá outro arquivo no momento da entrevista, o arquivo que será compilado será o arquivo enviado ao professor.
- Ao aluno que não enviar o código fonte, será atribuída nota zero.
- Não serão recebidos trabalhos atrasados. Caso o aluno não submeta o arquivo até a data limite, a nota zero será atribuída aos alunos.



- FLAGRANTES DE CÓPIAS DE TRABALHOS OCASIONARÃO EM NOTA ZERO PARA TODOS OS TRABALHOS SEMELHANTES.
- O trabalho poderá ser feito em trios ou quartetos.
- O trabalho deverá ser apresentado ao professor em datas e horários definidos. O não comparecimento de um dos integrantes do grupo resultará em nota zero para o aluno que faltar. O questionamento será feito de forma direcionada aos alunos do grupo entrevistado. A nota de cada aluno é individual, sendo esta dada por:
 - $notafinal = notaProjeto * notaEntrevista$
- Onde a *notaEntrevista* pode variar de 0 a 1.
- A indentação de código, comentários e qualidade da solução farão parte da avaliação do projeto.

Observações

O professor em hipótese alguma ajudará na construção do código.

O professor poderá tirar dúvidas sobre o enunciado do problema as funcionalidades pretendidas.