

# Trabalho Final – Paradigmas de Programação

**Data Entrega: 26/11**

## Equipes de Trabalho

Para realização dos trabalhos, a turma deverá ser dividida em **equipes de até 4 (quatro) alunos\***.

\* Trabalhos entregues com mais de 4 alunos terão sua nota final dividida pelo número total de integrantes.

## Atenção

- O trabalho será avaliado em **10 pontos**.
- Copias de trabalho mesmo que parciais, receberão nota **ZERO** (Para todos os envolvidos).
- O sistema deve ser feito na linguagem de **programação C/C++ ou JAVA**.
- Caso seja utilizada outra linguagem de programação, o trabalho será desconsiderado e a nota do grupo será zerada.
- Deverá ser entregue o **código completo** do **sistema**.
- Códigos que não rodarem ou apresentarem erros de compilação receberão nota **ZERO**.

## Especificações

Neste trabalho os alunos deverão fazer um programa em **C/C++ ou JAVA** que será utilizado por um atendente de balcão de companhias aéreas. Assim, o programa deverá atender as solicitações dos passageiros quanto à venda de passagens, fila de espera, desistência, etc.

Devido à política de compartilhamento de aeronaves entre as empresas, os operadores da empresa em questão possuem a permissão de inserir no máximo 10 passageiros em cada um dos voos disponíveis. Caso este número seja ultrapassado, os passageiros devem ser automaticamente incluídos em uma fila de espera contendo no máximo 5 passageiros.

Os dados principais de um passageiro são (acrescente o que desejar):

1. CPF
2. Nome
3. Endereço
4. Telefone
5. Número da Passagem
6. Número da Poltrona
7. Número do Voo
8. Horário

Voos disponíveis:

- BH-RIO
- BH-SP
- BH-BRASÍLIA

O programa deverá ter um menu principal com as seguintes opções:

1. Mostrar a lista completa dos passageiros de um determinado voo
2. Pesquisar um passageiro na lista de determinado voo pelo CPF
3. Pesquisar um passageiro na lista de determinado voo pelo nome
4. Cadastrar um passageiro na lista de determinado voo
5. Excluir um passageiro da lista de um determinado voo
6. Mostrar a fila de espera dos passageiros de um determinado voo

## Observações:

1. Todos os dados devem ser salvos em arquivos texto para garantir a persistência dos dados.
2. Sempre mostrar o voo em que as operações serão feitas.
3. Na opção 1 do menu, mostrar CPF, Nome, Número da Passagem e Número da Poltrona de todos os passageiros da lista.
4. Nas opções 2 e 3 do menu, se o passageiro estiver na lista, mostrar o cadastro completo do passageiro. Caso negativo, mostrar mensagem "Passageiro não consta neste voo".
5. Na opção 4, caso a lista de passageiros esteja cheia, **incluir automaticamente** o novo passageiro na fila de espera, até que está também fique cheia. Caso já esteja cheia, mostrar a mensagem "Fila Cheia, a reserva não pode ser feita".
6. Na opção 5 do menu, se houver algum passageiro na fila de espera do voo, **retirar automaticamente** da fila de espera e incluir na lista de passageiros. Enviar um alerta do ocorrido (Uma mensagem de texto na tela).
7. A lista de passageiros deve ter no máximo 10 passageiros. Acima deste valor, os passageiros devem ser incluídos na fila de espera que deve ter no máximo 5 passageiros.

## Implementar o seguinte layout de tela:

1. A tela básica (sem o menu principal) sempre deve ser mantida.
2. O voo atual deve estar sempre visível.
3. Sugestões:
  1. Antes de mostrar o menu principal (na figura abaixo) crie um menu que pede o número do voo. A seguir então o menu de opções (confirme mostrado abaixo) deve ser chamado para aquele voo selecionado.
  2. Faça primeiro toda a manipulação de tela: menus, submenus, etc. Depois acrescente as listas e filas.

### EMPRESA AÉREA QUEDA LIVRE - VOO BH-SP

#### MENU DE OPÇÕES

- [1] Mostrar Lista de Passageiros
- [2] Pesquisar Passageiro por CPF
- [3] Pesquisar Passageiro por Nome
- [4] Cadastrar Passageiro
- [5] Remover Passageiro
- [6] Mostrar Lista de Espera

Digite sua opção:

## Observações finais e o que deve ser entregue

- O código completo em C/C++ ou JAVA que implemente o projeto acima descrito.
- Os arquivos .txt utilizados como “Banco de Dados” para armazenar as informações dos voos sendo que os dados devem conter dois voos totalmente preenchidos (com todos os dados dos passageiros) com suas respectivas filas de espera (com todos os dados dos passageiros) e um voo preenchido porém, sem fila de espera.
- O código deve rodar sem problemas portanto, testem os projetos em outros computadores para evitar penalizações na nota final.
- Todos dados deverão ser fictícios no entanto com aparência real.
- Se preferirem, podem postar o link do repositório do Github ao invés de postar todos os arquivos no Moodle.
- Não é necessário que todos os integrantes do grupo postem o trabalho no Moodle. Basta que apenas um integrante faça a postagem, não esquecendo de colocar o nome completo de todos os integrantes do grupo.
- Não serão aceitas inclusões de novos membros no grupo após a entrega final.

**ATENÇÃO:** A organização do código, modularização, legibilidade, dentre outros fatores também farão parte da avaliação do trabalho.