



Universidade Federal de Ouro Preto - UFOP  
Instituto de Ciências Exatas e Biológicas - ICEB  
Departamento de Computação - DECOM  
Disciplina: Programação Orientada a Objetos  
Professor: Guillermo Cámara Chávez



## TRABALHO PRÁTICO I

### Instruções:

- O problema deve ser resolvido por meio de um programa em C++ e a STL;
- Utilize o site <http://www.cplusplus.com/> como referência;
- O código-fonte deve estar devidamente comentado;
- Não serão aceitos trabalhos que caracterizem cópia (mesma estrutura e algumas pequenas modificações) de outro ou de códigos da internet;
- Eventualmente, após a entrega dos trabalhos serão marcadas entrevistas com cada um dos alunos para apresentação dos mesmos para o professor.

### Entrega:

- A entrega do código-fonte será feita pelo *Moodle* até o dia **X**.
- Deve ser entregue um zip com:
  - Relatório (descrito ao fim deste texto);
  - Código fonte;
  - Descrição de como compilar e executar o projeto pela linha de comando.
  - Vídeo mostrando a compilação, execução e uso do código

### Avaliação:

- Funcionamento adequado do programa
  - Códigos que não compilarem e tiverem *warnings* diminuirão a nota;
  - Corretude (independente se gerado por IDE ou manualmente).
- Atendimento ao enunciado do trabalho;
- Comentários;
- Identação do código e boas práticas de programação;
- Boa organização do código fonte em geral;
- Bom uso da STL.

## Enunciado

Desenvolver um sistema que permita realizar o gerenciamento de serviços e vendas de peças em uma oficina mecânica. Como atores no nosso caso de uso, temos:

1. Os funcionários da oficina mecânica, podendo ser vendedores (que são também atendentes) ou mecânicos;
2. Os clientes;
3. Os veículos dos clientes;
4. Os itens de serviços e produtos de venda;

Todos os clientes e seus respectivos veículos deverão estar cadastrados no sistema da oficina para que possam realizar orçamento ou execução do serviço.

Ao chegar à oficina o cliente será atendido por um vendedor que irá providenciar o seu cadastro bem como de seu veículo, caso não exista, e em seguida gerar uma ordem de serviço para solicitação de orçamento ou manutenção.

Na abertura da ordem deverá possuir o motivo da manutenção, além de outras informações como quilometragem do veículo. O sistema deverá estar apto a gerar todo o histórico de manutenções do veículo.

O mecânico deverá visualizar as ordens de serviço abertas e escolher uma para realizar o serviço, caso seja orçamento o serviço não deverá ser executado antes de prévia autorização por parte do cliente, para tal o vendedor deverá visualizar as ordens de serviço aguardando aprovação do cliente, contatar o cliente e verificar se o mesmo aprova, caso seja aprovada ela deverá retornar a fila de ordens de serviços e poderá ser executada pelo mesmo mecânico ou por outro mecânico, caso o cliente não autorize deverá ser encerrada.

Caso a ordem de serviço seja aprovada, o mecânico irá detalhar os serviços realizados, como por exemplo, alinhamento, balanceamento, troca de peça, entre outros, e seus respectivos preços, além de informar também as peças utilizadas e seus preços. As trocas de peças envolvem o preço da mão de obra para troca da peça e também o preço da peça, um item não implica que o outro não será cobrado.

O sistema deverá ter um administrador para editar os dados dos vendedores e também dos mecânicos, este usuário poderá ter uma senha fixa.

A tela inicial do sistema deve ter as seguintes opções:

- Login (Para administrador, vendedor ou mecânico).
- Sair do sistema, não é necessário persistir as informações ao sair do sistema, ou seja, as informações devem existir enquanto o sistema estiver rodando.

Caso o usuário logado seja o administrador deverão ser exibidas as seguintes opções:

- Editar dados de Vendedores.
- Editar dados de Mecânicos.

Caso o usuário logado seja um Vendedor deverão ser exibidas as seguintes opções:

- Cadastrar cliente e veículo.
- Gerar ordem de serviço.
- Visualizar ordens de serviço de orçamento pendentes de aprovação do cliente e marcar como aprovadas.
- Visualizar ordens de serviço executadas e realizar o fechamento.

Caso o usuário logado seja um mecânico deverão ser exibidas as seguintes opções:

- Visualizar ordens de serviço abertas.
- Cadastrar serviços executados e peças utilizadas.

O trabalho consiste em implementar um programa em C++ seguindo as instruções listadas na parte anterior, as classes do programa devem ter alta coesão e baixo acoplamento e deve-se explorar os conceitos de abstração, herança e polimorfismo, para tal deverá ser feito uso de herança e classes abstratas.

Deverão ser entregues os seguintes itens compactados em um zip:

- Relatório
- Código fonte

O relatório deve ter:

- A descrição da arquitetura e implementação do trabalho em C++ através de diagramas da UML explicando detalhadamente como cada item foi construído,
- Quais foram as decisões de projeto, quais os recursos de linguagem foram utilizados, etc.
- O relatório deverá ter também as instruções de compilação e execução do programa utilizando terminal ou prompt de comando, com exemplos.

O Código-fonte do programa deve ter:

- Comentários descrevendo os detalhes de implementação.
- Deve-se seguir as boas práticas de escrita de código como:
  - Nomenclatura, como CamelCase.
  - Alinhamento.
  - Boa organização do código em geral para facilitar a leitura