Programação Orientada a Objetos I

# Exercício - **Auto-FT Veículos S.A.**

# 

# Participantes:

Larissa Benevides Vieira RA: 200805

Gabriel Domingues Ferreira RA: 216207

João Vitor Portuense Silva RA: 218965

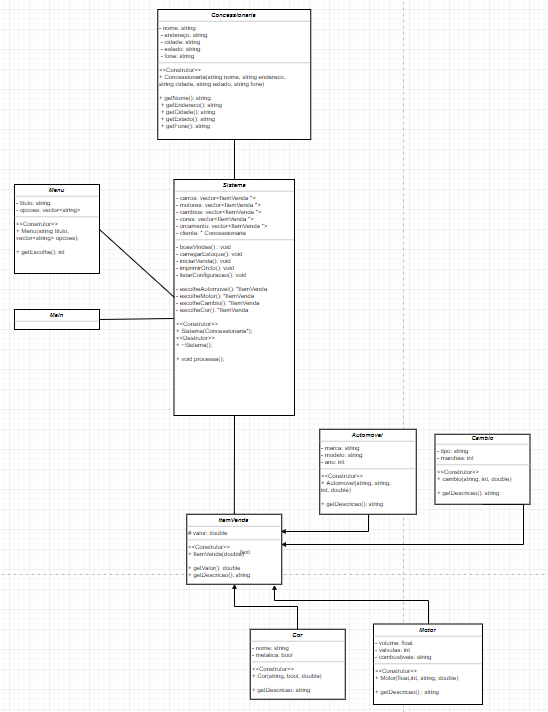
Arthur Santos Machado Rodrigues RA: 213315

O programa deste exercício emula a situação de uma concessionária de veículos e supõe um vendedor que atende um cliente interessado na compra de um carro. O sistema permite configurar o veículo de acordo com as escolhas do cliente e imprimir (em tela) um orçamento.Parte do código fonte está disponível no Ensino Aberto.

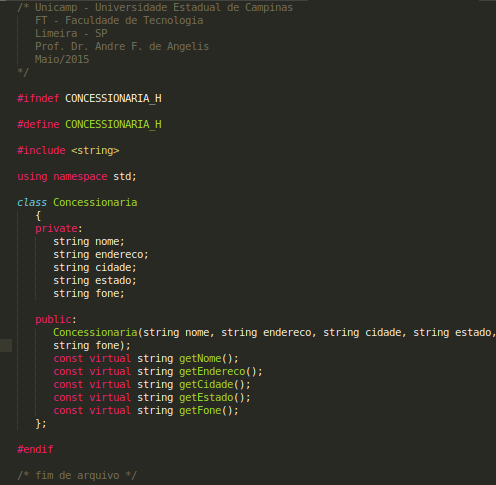
A seguir, as tarefas que devem ser feitas pelo grupo:

1.Baixe o código-fonte; ✅

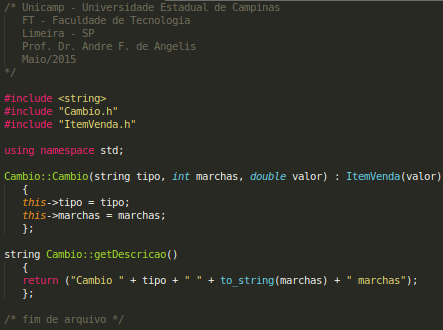
2. Desenhe o diagrama de classes do programa (gere um .pdf para entrega); ✅



3. Está ausente o arquivo "Concessionaria.h”; recrie-o com base em "Concessionaria.cpp”; ✅



4. Está ausente o arquivo "Cambio.cpp”; recrie-o com base em "Cambio.h” e nas implementações de **Motor** e **Cor**; ✅



5. Estude a classe **Menu** e seu uso para diversas situações no programa;✅

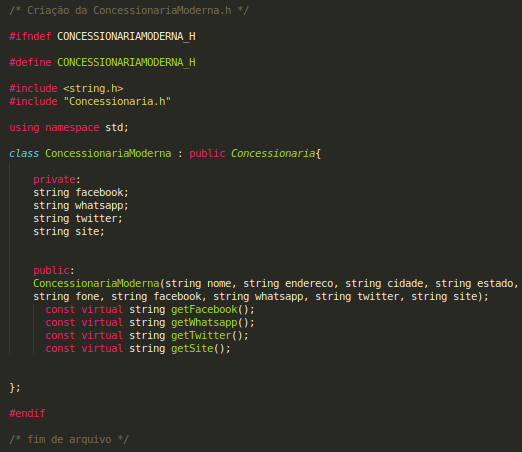
6. Compile e rode o sistema, que deve estar operando corretamente; a funcionalidade de imprimir orçamento ***não*** está disponível; ✅

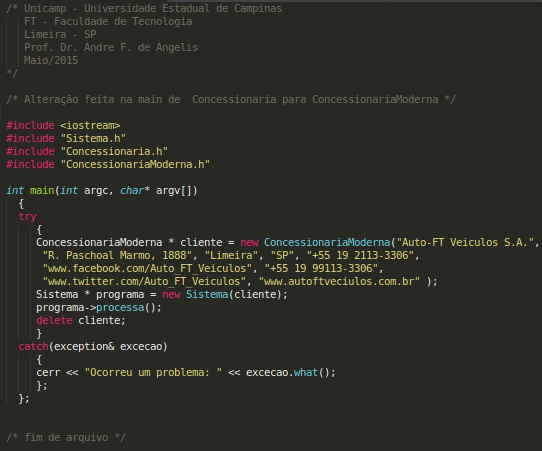
7. Escreva o método *imprimirOrcto()* da classe **Sistema** (existe, mas está vazio); o método deve usar somente os atributos *cliente* (para imprimir dados da concessionária no cabeçalho do orçamento) e o *vector orçamento* (para contabilizar os preços dos itens); veja o método *boasVindas()* para aproveitar ideias; ✅



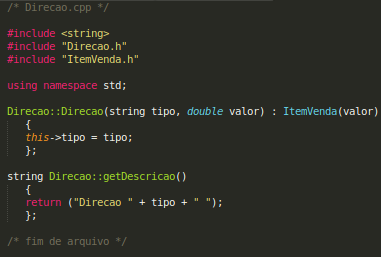
8. Crie a classe **ConcessionariaModerna**, derivada de **Concessionaria**, com novos atributos que facilitam contato dos clientes: *site, facebook, twitter e whatsapp*; use essa classe na função ***main*** e altere o programa para exibir também estes novos dados em ***todas*** as ocasiões em que dados da concessionária forem apresentados; ✅

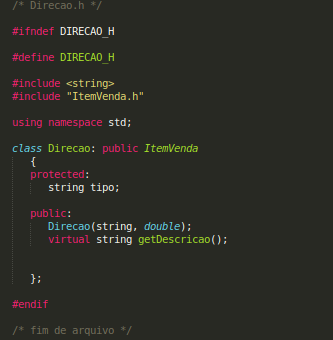


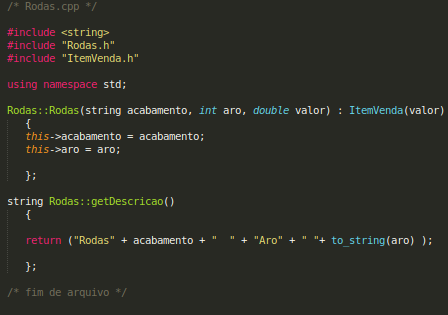


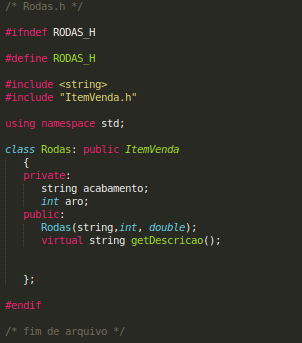


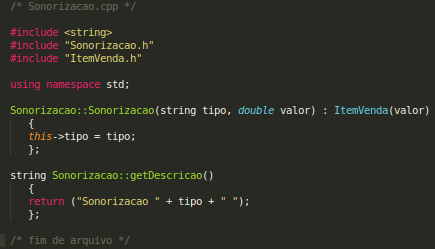
9. A partir de **ItemVenda**, derive: **Direcao**, **Rodas** e **Sonorizacao** para que novos acessórios possam ser colocados nas configurações dos veículos; crie *vários* objetos de cada uma destas três classes no método **Sistema**::*carregarEstoque()*; altere **Sistema**::*iniciarVenda()* para solicitar estes acessórios também; em **Sistema**, escreva os métodos *escolheDirecao(), escolheRoda()* e *escolheSonorizacao*(); ✅

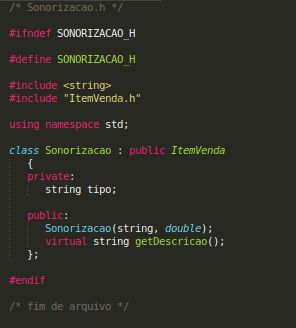


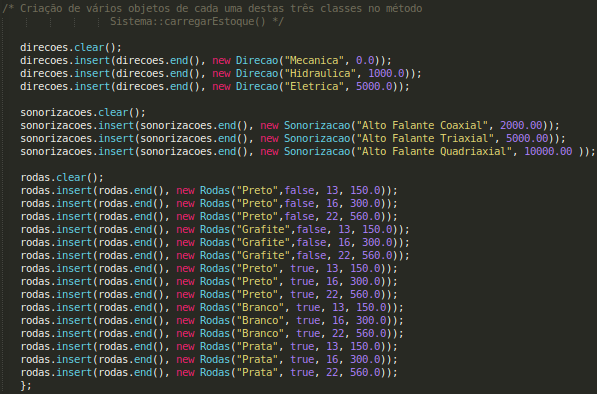


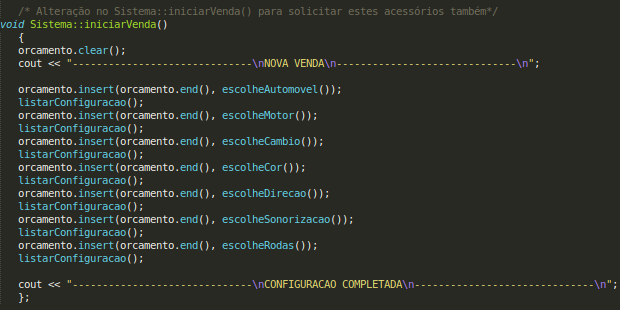


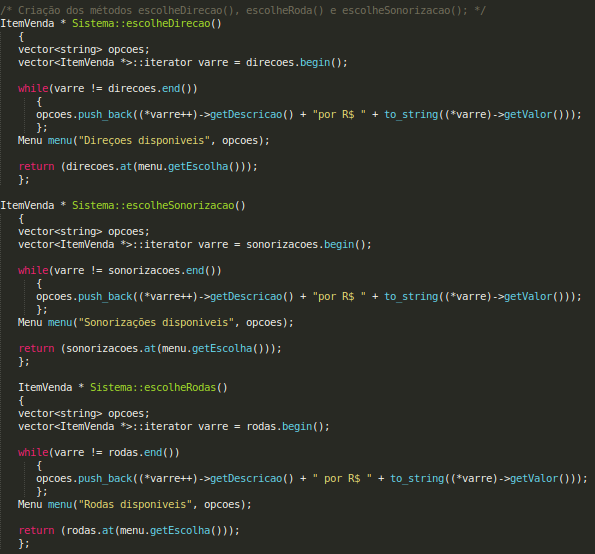






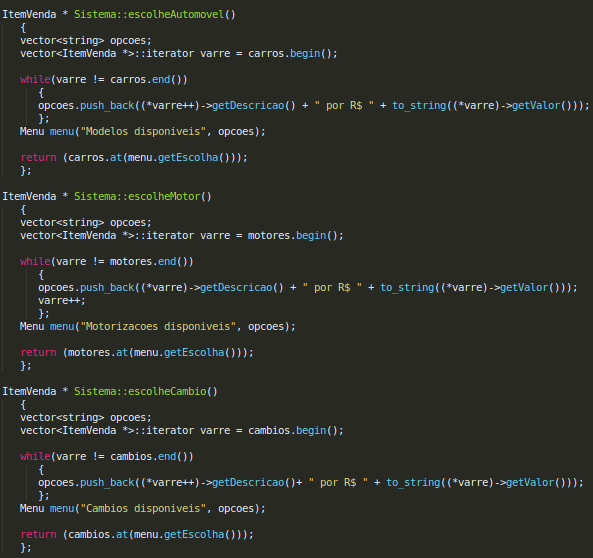
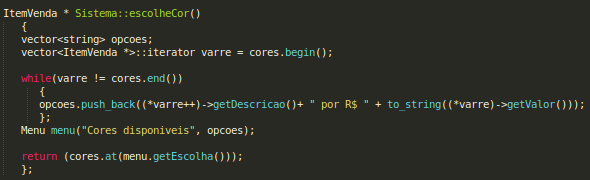


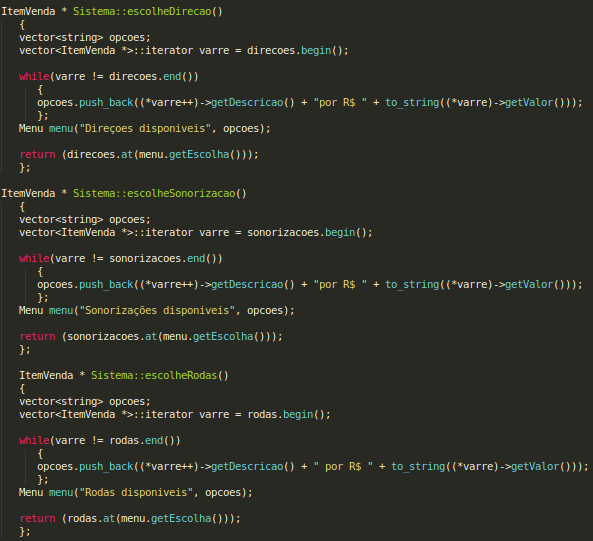




10. Confira: **Sistema**::*imprimirOrcto()* ***não*** deve precisar de modificações; ✅

11. Altere todos os métodos **Sistema**::*escolheXXXX* para apresentarem também o valor do item oferecido ao cliente; inspire-se em *escolheMotor()*, que já faz isto, e tome cuidado para não usar "++” sobre um *iterator* mais vezes que o devido; ✅





12. Apresente o diagrama de classes e o projeto rodando ***em aula***, conforme o cronograma da turma; ✅

13. Gere **um** (1) arquivo .zip com os fontes e o diagrama de classes e carregue-o no Ensino Aberto; ✅

14. *Estude muito bem o código-fonte para aprender técnicas e opções de codificação; algumas delas poderão ser solicitadas nas próximas avaliações.*  ✅

**Fizemos um diagrama de classes que representa todo o programa (após as implementações solicitadas no exercício).**

