



MC302 - Programação Orientada a Objetos

Instituto de Computação - Unicamp

Primeiro Semestre de 2024

Professora: Esther L. Colombini (esther@ic.unicamp.br)

PEDs: Cristiano Gabriel de Souza Campos / Wladimir Arturo Garces Carrillo

Projeto Prático 1 - Entrega 19/5/24

1 Objetivo

Estudo e implementação dos conceitos de programação orientada a objetos abordados na disciplina para o projeto e construção de um sistema a ser definido pelo grupo.

2 Descrição

Este trabalho contempla a primeira etapa da construção de um sistema orientado a objetos. Nesta fase, o sistema deverá ser descrito em sua versão mais completa, mas apenas os elementos base do mesmo deverão ser implementados nesta entrega.

3 Tópicos Abordados

Os seguintes conceitos deverão ser adequadamente incorporados no sistema a ser desenvolvido:

- **Classes, variáveis e métodos:** o sistema deve contemplar classes, suas propriedades e métodos e sobrecarga de métodos;
- **Visibilidade:** o sistema deve conter a aplicação adequada de modificadores de visibilidade;

- **Herança:** o sistema deve conter ao menos uma estrutura de herança;
- **Variáveis e Métodos estáticos:** o sistema deve conter ao menos uma variável de classe e um método de classe;
- **Arrays:** devem ser empregados sempre que necessário;
- **Enumeração:** o sistema deve contemplar um exemplo de uso de enumerações;
- **Entrada e saída de dados:** o sistema deve permitir a entrada e saída de dados;
- **Relacionamentos (associação, agregação ou composição):** o sistema deve contemplar relacionamentos;
- **Classes Abstratas:** ao menos uma classe abstrata deve ser proposta;
- **Polimorfismo:** ao menos um exemplo de polimorfismo de tipo e de método deve ser proposto.

Além disso, utilizaremos diagramas de classe UML para representação do projeto de classes.

4 Atividades

Dentre as atividades previstas para a construção do sistema, o grupo deverá entregar:

- Um resumo descrevendo o sistema projetado
- O levantamento de todas as funções que devem ser executadas no sistema (por exemplo: cadastrar usuário, pedir carona, oferecer carona, adicionar membro ao grupo, etc.)
- A estrutura das classes que resolve o problema abordado incluindo:
 - Atributos
 - Métodos (construtores, acesso, métodos gerais)
 - Relacionamentos
- A implementação do sistema
- O teste das funcionalidades do sistema

4.1 Grupos

Os grupos deverão ser compostos de 3-4 alunos.

4.2 O trabalho será avaliado segundo os seguintes critérios:

- **Corretude (25%):** O código não deve possuir warnings ou erros de compilação e não deve emitir exceptions em nenhuma situação.
- **Projeto (40%):** A qualidade do projeto realizado de acordo com os princípios de orientação a objetos. Quanto mais conceitos corretamente aplicados, maior a nota do projeto. Este critério irá considerar, inclusive, o diagrama UML construído para representar o projeto do sistema proposto.
- **Aderência ao Enunciado (25%):** A implementação deve realizar o que é requisitado (atividades + tópicos abordados).
- **Comentários (5%):** Os comentários devem ser suficientes para explicar os trechos mais importantes da implementação.
- **Convenções (5%):** A implementação deve seguir as convenções do Java. O código deve ser indentado, modularizado e hierarquizado para melhor organização.

A nota final do trabalho poderá ser composta por múltiplas avaliações realizadas segundo esse critério. A nota de cada aluno no grupo será computada a partir da nota do trabalho e da participação de cada membro do grupo no desenvolvimento do projeto.

4.3 Observações

O trabalho deverá ser entregue via github. Note que as atividades de cada aluno poderão ser observadas pelo site, podendo fazer parte da avaliação individual.