



CURSO DE CIÊNCIAS DA COMPUTAÇÃO

Programação Orientada a Objetos

Turmas: P3

Professor: Jeofton Costa Melo

Atividade Exclusiva em Sala de Aula

Entrega – 23/05/2024 – 21:40

Até 1 ponto

Objetivos: Implementar um sistema de estacionamento que permita a entrada e saída de veículos, cálculo do valor a ser pago e gestão de vagas. O sistema deve explorar os conceitos de orientação a objetos, como herança, polimorfismo e encapsulamento.

Requisitos:

- A. O sistema deve permitir a entrada de veículos de diferentes tipos (Carro, Moto);
- B. O sistema deve calcular o valor a ser pago com base no tempo de permanência no estacionamento;
- C. Deve ser possível visualizar a quantidade de vagas disponíveis;
- D. O sistema deve utilizar herança para os diferentes tipos de veículos;
- E. Encapsulamento deve ser utilizado para proteger os atributos das classes;
- F. Polimorfismo deve ser utilizado para o cálculo do valor a ser pago, dependendo do tipo de veículo.

Orientações:

- A. Os grupos devem ter entre 04 e 06 integrantes, apenas um envia o link do repositório no github, do sistema desenvolvido através do Blackboard;
- B. O sistema deve ser desenvolvido em 02 horas e a apresentação será feita mediante a execução do sistema por 01 dos integrantes de cada grupo, no tempo máximo de 05 minutos;

- C. Apenas um dos integrantes deve enviar o link no Blackboard, bem como o nome e o RGM dos integrantes do grupo;
- D. O projeto deve ser criado em Java;
- E. Testar o sistema com diferentes cenários, adicionando e removendo veículos e verificando o cálculo do valor pago;

Critérios de Avaliação:

- A. Implementação correta das classes e métodos;
- B. Uso adequado dos conceitos de herança, polimorfismo e encapsulamento;
- C. Funcionamento correto do sistema conforme requisitos e diagrama UML apresentado;
- D. Clareza e organização do código;
- E. Execução da atividade (implementação e apresentação) dentro dos tempos definidos.

Dicas:

- A. A classe veículo deve ser abstrata;
- B. Para cadastrar os veículos, use a classe Java ArrayList;