



- **Leia a avaliação com atenção. A interpretação das questões faz parte da avaliação.**
- **A avaliação DEVE ser resolvida de forma individual.**
- **Ao final da avaliação, um arquivo compactado com o seu nome, contendo somente o código fonte usado para resolver as questões, deverá ser enviado para a tarefa AVALIAÇÃO 01 disponível no Moodle.**
- **É de responsabilidade do aluno a gravação correta dos arquivos da avaliação.**
- **O tempo de avaliação é das 18h00min às 19h40min.**
- **Boa Sorte!**

1) Implementação do **padrão de projeto GoF Strategy**:

Crie uma interface (estratégia abstrata) chamada Entrega contendo um método chamado calcular.

Crie duas classes estratégias concretas que implementam a estratégia abstrata apresentando ao usuário informações diferentes sobre a forma de entrega:

a) EntregaPadrao: Entrega acrescida de frete calculado com base no envio pelos Correios via PAC.

b) RetiradaNaLoja: Entrega sem custos mediante retirada do produto pelo próprio cliente na loja física.

Crie uma classe abstrata CarrinhoDeCompras que contém um atributo do tipo da estratégia abstrata criada e um método chamado consultarEntrega que irá invocar o método calcular disponível na estratégia abstrata.

Crie duas classes contexto concretas que estendem a classe CarrinhoDeCompras e implementam o método herdado instanciando estratégias diferentes: ComprasContextoA (Entrega Padrao); ComprasContextoB (Retirada na Loja).

Demonstre o uso do padrão de projeto implementado fazendo simulações de compras com diferentes formas de entrega.

Peso (3,0)

2) Aplique o **padrão de projeto GoF Factory Method** na aplicação criada na questão 1, e comente o código para indicar o que foi alterado e com que objetivo o padrão foi empregado. **(Peso 2,0)**

3) Crie uma pequena aplicação que possui uma classe referente ao cenário de um jogo. Crie essa classe com o nome "CenarioSeuNome", e garanta que em todo o sistema a mesma tenha uma única instanciação, ou seja, crie a classe empregando o **padrão de projeto GoF Singleton (Peso 2,0)**.

- a) Nessa classe deve ser criado um método chamado detalhamento(), que pode ter parâmetros ou não, e irá imprimir na tela o seu nome, composto por nome e sobrenome (como autor do cenário) e uma descrição genérica do que compõe o cenário.
- b) Apresente de alguma forma uma evidência de que a classe foi instanciada uma única vez.
- c) Para testar a Singleton criada, faça na classe Main da aplicação uma chamada ao método detalhamento().

2) Apresente uma aplicação do **padrão de projeto GoF Adapter** no mesmo projeto utilizado para resolver a questão anterior referente ao padrão Singleton. Comente o código de forma a indicar com que objetivo o padrão foi aplicado, e quais alterações foram realizadas no código para atender o que sugere o padrão escolhido. **(Peso 3,0)**