**UNIVERSIDADE POSITIVO**

**TÓPICOS ESPECIALS EM DESENVOLVIMENTO DE SOFTWARE**

**Prof Escobar**

**Avaliação prática de Orientação a Objetos em C#**

* Responda às questões teóricas e desenvolva o código para as questões neste documento;
* Quando a questão solicitar implementação de código, deve-se implementar, também, classes consumidoras e que permitam os testes das classes desenvolvidas;
* Os prompts de GPT, Copilot ou qualquer outra IA generativa devem ser entregues junto com a sua avaliação;
* Pontos possíveis: 2,5;
* Realização individual;
* Data de entrega: 02/05/2024, 23h59 impreterivelmente.

Realize o que se pede.

Link script ChatGPT: <https://chat.openai.com/share/cacad043-c4b0-4d52-9ea3-d0a39951c4ce>

* Explique o que é herança em orientação a objetos e como ela é implementada em C#. Dê um exemplo prático de como a herança pode ser utilizada em um sistema de gerenciamento de funcionários.

R: A herança é basicamente “puxar” as propriedades de uma classe superior para outra classe. É usada quando as propriedades de uma classe vai servir em outra, ou seja, uma classe deriva as propriedades de outra.

* O que são interfaces em orientação a objetos e qual é a sua importância na construção de sistemas em C#? Dê um exemplo de como uma interface pode ser utilizada para garantir a interoperabilidade entre diferentes classes em um sistema de pagamento online.

R: Interface são tipo contrato, que garante que alguns métodos e propriedades estejam dentro da classe. Quando a propriedade se encaixa em duas classes, mas não podem ser herdadas de duas classes diferentes, utiliza-se a interface para lidar com o mesmo.

* Diferencie sobrecarga e sobreposição (ou sobrescrita) e forneça exemplos de cada um em C#.

R: a sobreposição ( override ) sobrescreve o método da classe mãe para classe mais específica e a sobrecarga ( overloading ) é feito para utilizar o mesmo método várias vezes, mas utilizando parâmetros diferentes.

* Crie uma classe Produto que represente um produto em uma loja online. A classe deve ter os seguintes atributos: Nome, Preco, QuantidadeEmEstoque. Implemente métodos para adicionar e remover unidades do estoque, e um método para calcular o valor total do produto em estoque. Os métodos devem alterar o estado do objeto instanciado
* Implemente um sistema para uma locadora de filmes. Crie uma classe Filme com os seguintes atributos: Titulo, Genero, Duracao, Disponivel (indicando se o filme está disponível para locação). Implemente métodos para registrar a locação de um filme, registrar a devolução de um filme e verificar se um filme está disponível para locação. Não há necessidade de implementar rotinas de bancos de dados.
* Implemente uma hierarquia de classes para representar diferentes tipos de veículos, como carros, motos e bicicletas. Cada classe deve herdar da classe base Veiculo. Adicione atributos específicos para cada tipo de veículo, como número de portas para carros, cilindrada para motos e número de marchas para bicicletas.
* Desenvolva um sistema de gerenciamento de contas bancárias com diferentes tipos de contas, como conta corrente, conta poupança e conta empresarial. Utilize herança para representar a relação entre as classes de contas e implemente métodos específicos para cada tipo de conta, como calcular juros para contas poupança e verificar saldo mínimo para contas empresariais.
* Crie uma aplicação para uma escola de idiomas que oferece diferentes tipos de cursos, como inglês, espanhol e francês. Utilize herança para representar a relação entre as classes de cursos e implemente métodos específicos para cada tipo de curso, como calcular a média de notas para cursos avançados e gerar certificados para cursos concluídos.
* Desenvolva um sistema de processamento de pagamentos com diferentes métodos de pagamento, como cartão de crédito, boleto bancário e transferência bancária. Crie uma interface IMetodoPagamento com métodos para realizar o pagamento e verificar o status do pagamento. Implemente classes para cada método de pagamento que herdem dessa interface e forneça a lógica específica para cada método de pagamento.
* Projete um sistema de reservas de voos para uma companhia aérea. Crie uma interface IReserva com métodos para reservar um voo, cancelar uma reserva e verificar o status da reserva. Implemente classes para diferentes tipos de reservas, como reserva de voo regular, reserva de voo com upgrade de classe e reserva de voo para grupos grandes. Utilize a interface para garantir a interoperabilidade entre os diferentes tipos de reservas.