

5ª Lista de Exercícios

Vetores de Objetos, Relacionamentos, Herança e Polimorfismo

Crie ou altere as classes conforme definido nos itens abaixo e crie um programa para testar essas classes:

1. Crie a classe `Triangulo` usando a classe `Ponto`. Defina um construtor para `Triangulo` que recebe os três pontos que o formam. Defina os métodos `float perimetro()` e `float area()`. Para calcular a área do triângulo use:

$$área = \sqrt{S \cdot (S - a) \cdot (S - b) \cdot (S - c)}$$

onde a , b e c são os lados do triângulo e S é o perímetro dividido por 2, ou seja $S = (a+b+c)/2$.

2. Implemente a classe `Pentágono` que tem como atributo um vetor de 5 pontos. Crie um construtor que recebe o vetor de pontos que define o pentágono. Implemente também os métodos:
 - a) `perimetro()`: para calcular o perímetro do pentágono
 - b) `area()`: para calcular a área do pentágono (dica: use a fórmula da área do triângulo definida anteriormente)
3. Crie a classe `DataHora` usando as classes `Data` e `Hora`, vistas anteriormente, sem usar herança, ou seja, `Data` e `Hora` serão atributos de da classe `DataHora`. Crie o construtor adequado e implemente os métodos `void incrementaDia()` e `void decrementaDia()` para incrementar e decrementar a data em um dia, respectivamente; e `void incrementaSeg()` e `void decrementaSeg()` para incrementar e decrementar a hora em um segundo, respectivamente. Atenção: quando a hora avançar para o dia seguinte, a data deve avançar e quando a hora voltar para o dia anterior, a data deve retroceder um dia.
4. Crie as classes `Pessoa` e `Certidao` para representar o seguinte cenário: toda pessoa tem nome e pode ter ou não uma certidão de nascimento. Toda certidão de nascimento possui a data de emissão e a pessoa a qual ela se refere. Note que uma pessoa pode não ter uma certidão, mas uma certidão sempre referencia a uma pessoa. Crie também um método para associar uma certidão de nascimento à pessoa. Dica: use a classe `Data` implementada em sala de aula.
5. Crie as classes `Aluno`, `Livro` e `Emprestimo` para representar o seguinte cenário em uma biblioteca: um aluno possui matrícula e nome, um livro possui código, título e nome do autor. Um empréstimo registra a data do empréstimo, o aluno que pegou o empréstimo e o livro emprestado. Dica: use a classe `Data` implementada em sala de aula.
6. Crie a classe `Computador` e as demais classes necessárias (com seus respectivos atributos e construtores) para representar o seguinte cenário: um computador é composto de HD, placa de vídeo, placa de rede, placa mãe, processador e fonte de alimentação. HD tem capacidade (Gb), placa de vídeo tem modelo (String) e quantidade de memória (Gb) e velocidade do processador (GHz), placa mãe tem modelo (String), processador tem velocidade (GHz) e modelo (String), fonte de alimentação tem potência (Kw) e placa de rede tem velocidade (mbits). Implemente o método `void imprimir()` na classe `Computador` para imprimir a configuração do mesmo.

7. Use seus conhecimentos de herança (generalização/especialização) para definir uma hierarquia de classes que represente as classes abaixo, sem a duplicidade de atributos, onde for possível:

Carro
- modelo : String - numLugares : int - numPortas : int - comprimento : int - ano : int - cor : String - placa : String

Aviao
- modelo : String - numLugares : int - prefixo : String - comprimento : int - ano : int - cor : String - numTurbinas : int

Lancha
- modelo : String - numLugares : int - comprimento : int - ano : int - cor : String - numMotores : int

8. Use seus conhecimentos de herança (generalização/especialização) para definir uma hierarquia de classes que represente as classes abaixo, sem a duplicidade de atributos, onde for possível:

Cachorro
- raca : String - distanciaFaro : double - cor : String - intensidadeLatido : double - preco : double - nascimento : Date

Gato
- raca : String - alturaPulo : double - cor : String - peloLongo : boolean - preco : double - nascimento : Date

Coelho
- raca : String - cenourasPorDia : int - cor : String - peloLongo : boolean - preco : double - nascimento : Date

9. Uma Biblioteca possui em seu acervo Revistas (título e número) e Livros (título e nome do autor). Na Biblioteca crie métodos para:
- Adicionar e remover Livros e Revistas do acervo;
 - Emprestar e devolver Revistas e Livros;
 - Listar as Revistas e Livros existentes na biblioteca com seu status (emprestado ou disponível).
10. Implementar as classes Imovel (valor, zona (Norte, Sul, Oeste ou Centro)) e tamanho em metros quadrados), ImovelUsado (ano de construção) e ImovelNovo. Calcular o IPTU sabendo que o valor é de R\$ 20,00/m². Imóveis novos na zona norte tem desconto de 10%, no centro tem desconto de 5% e na zona sul tem acréscimo de 5%. Imóveis usados tem 1% de desconto para cada ano de idade, limitado a 50% de desconto.
11. Usando a classe Veiculo, Carro e Caminhao, vistas em sala, implemente a classe CarroTurbinado, que carro com turbo. A velocidade máxima do carro turbinado aumenta 20% em relação ao carro comum, caso o turbo esteja ligado.

12. Implemente a seguinte hierarquia de classes:

