

Aluno(a): _____

Turma: _____

Data: _____

Esta avaliação deve ser respondida preferencialmente usando caneta esferográfica azul.
Faça distinção clara entre maiúsculas e minúsculas.
Seja claro, formal e sucinto.
Códigos desnecessários e que reduzam o desempenho do sistema serão penalizados.
Utilize as boas práticas de programação.
LEIA AS QUESTÕES ATÉ O FINAL ANTES DE COMEÇAR.
Sempre que puder, declare os atributos de instância privados.

Questão 1 (5) – Considere a classe abaixo:

UTILIZE GENERICS SEMPRE QUE POSSÍVEL NESSA QUESTÃO.

```
public class Carro {  
    private String placa;  
    private String cor;  
    private int ano;  
    private String renavan;  
  
    //getters e setters  
  
}
```

Considere a classe Carro acima criada e com os getters e setters.

1.1 – Crie uma classe chamada Acessorio. Essa classe terá diversas subclasses, não permita a criação de objetos da classe Acessorio. Essa classe possui apenas dois atributos (marca e preço). Utilize os tipos apropriados. Crie os getters e setters. Crie um método chamado “calcularImposto” que retorne um double. Crie esse método de maneira que todas as subclasses de Acessorio sejam obrigadas a implementar esse método. Crie apenas um construtor em Acessorio. Esse construtor deve receber a marca e o preço.

1.2 – Crie duas subclasses de Acessorio. Som e VidroEletrico. O imposto de Som é dado por (preço * 0.3) e o de VidroEletrico é dado por (preço * 0.2).

1.3 – Modifique a classe Carro adicionando uma lista de acessórios. A lista DEVE ser privada. Essa lista conterá os acessórios que o carro vai ter. Não amarre o sistema. Crie mais métodos se necessário (não precisa reescrever a classe carro toda, só a declaração da classe e o que for inserir a mais, além do que já aparece na figura).

1.4 – Crie uma classe principal que seja responsável por criar uma lista de carros. Crie dois carros e insira na lista. Em cada carro, adicione um Acessorio Som e um Acessorio VidroEletrico.

1.5 – Crie uma classe chamada Utils2 que possua apenas um método estático sem retorno: imprimirCustos. Esse método receberá a lista de carros. Imprima o total do preço de todos os acessórios de todos os carros e o total de imposto de todos os acessórios de todos os carros.

1.6 – Crie um método, também em Utils2 denominado transformaListaEmMapa(List carros): Map que recebe uma lista de carros e retorna um mapa, em que as “chaves” são as placas dos carros e os “valores” são os objetos do tipo carro. Não modifique a assinatura do método. Utilize o iterator nessa questão.

Questão 2 – (3) – Considere que existe uma classe denominada Utilidades. Nessa classe existe um método estático com a seguinte assinatura: getPalavroes():Set. Esse método retorna uma lista de palavras existentes para a língua portuguesa. Não crie essa classe nem esse método, eles já existem para você utilizar.

Crie um programa que receba do console (entrada do teclado) uma frase. Em seguida, esse programa deve chamar o método Utils.verifica(String texto). Esse método é responsável por verificar se o texto (variável “texto”) tem menos de 10 caracteres e, caso tenha, deve lançar uma exceção checked denominada TextoCurtoException. Crie também a classe dessa exceção. Quando essa exceção for lançada, deve ser passada uma mensagem para o construtor dela, informando quantos caracteres possui o “texto” inserido pelo usuário. Ex: texto inserido com 4 caracteres.

Em seguida, crie o método Utils.moderar(String texto): String. Esse método deve substituir todos os palavras encontrados em “texto” por “#####” -> Cada caractere do palavra deve ser substituído por um caractere #. Ou seja, se o palavra tiver 6 caracteres, o palavra no texto deverá ser substituído por #####.

PS: É importante deixar claro que nessa questão os métodos devem considerar que se um palavra for escrito em letra minúscula ou letra maiúscula, este deve ser substituído.

PS2: Considere que os palavras são palavras simples, não compostas, ou seja, não existem espaços.

Exemplo de entrada e saída: A casa palavra1 é minha. -> A casa ##### é minha.

Questão 3 (2) – Observe a questão abaixo. O que sai no console?

```
import java.util.ArrayList;
import java.util.List;
public class Console {
    public static void main(String[] args) {
        List a = new ArrayList();
        String x = "virgem";
        a.add(x);
        a.add(x);
        a.add(x);
        x = "aquario";
        a.add("leao");
        a.add("escorpiao");
        a.add("libra");
        int numero = 13;
        metodoA(numero);
        System.out.println(numero);
        metodoC(a);
        metodoB(a);
        metodoD(a);
        System.out.println(a);
        int i = 0;
        metodoE(i);
        System.out.println(i);
    }
    public static int metodoA (int numero) {
        numero = 11;
        System.out.println(numero);
        return numero;
    }
    public static void metodoB (List c) {
        c.add("gemeos");
    }
    public static void metodoC (List a) {
        a = new ArrayList();
        a.add("virgem");
    }
    public static void metodoD (List c) {
        metodoB(c);
        metodoC(c);
    }
    public static void metodoE (int numero) {
        ++numero;
        numero++;
    }
}
```