



Aluno(a): _____

Turma: _____

Data: _____

Esta avaliação deve ser respondida preferencialmente usando caneta esferográfica azul. Não utilize cor vermelha ou alguma tonalidade próxima. Será avaliado o domínio do assunto e a capacidade de expressão na norma culta da língua. Seja claro, formal e sucinto.
Códigos desnecessários que reduzam o desempenho do sistema serão penalizados.

Questão 1 (1,4) - Crie uma classe chamada Utils. Essa classe possuirá 1 método chamado inverter(String x). Esse método deve inverter a String e retornar a String invertida.

Ex. entrada: Casa → saída: asaC.

Crie uma classe Principal que receba do teclado 1 palavra. Utilizando o método da classe Utils, exiba a palavra invertida no console.

Questão 2 – (2,2) Considere que exista uma classe chamada Processamento com um método chamado contarPalavrasNaoExistentes que receba como argumentos dois arrays de String: um representando as frases de um texto e outro representando as palavras de um dicionário.

Processamento.contarPalavrasNaoExistentes(String [] frases, String [] palavras)...

Verifique o total de palavras existentes nas frases que não existem em palavras. Retorne o total. Suponha que nas frases não existe pontuação. Apenas palavras separadas por espaço. O método não deve ser *case sensitive*, ou seja, se existir a palavra “cAsa” na frase e estiver como “CaSa” no dicionário, o método deve reconhecer a palavra. Resumindo, nesse caso, a palavra não entra na contagem de palavras não-existentes.

Ex:

frases: “Minha casa é bonita”, “Meu jardim tem flores”

palavras: “abacate”, “casa”, “minha”

Questão 3 – O que sai no console? (2,0)

```
public class Principal {
    public static void main(String[] args) {
        Caneta c = new Caneta();
        c.setCor("preta");
        Caneta c2 = new Caneta();
        c2.setCor("azul");
        c2.setQuantidade(8);
        Caneta c3 = new Caneta();
        c3.setCor("rosa");
        metodoCan1(c);
        metodoCan2(c2);
        metodoCan3(c3);
        int i = c2.getQuantidade();
        qtd(i);
        System.out.println(i);
        System.out.println(c3.getQuantidade());
        System.out.println(c.getCor());
        System.out.println(c2.getCor());
        System.out.println(c3.getCor());
        System.out.println(c.getQuantidade());
        System.out.println(c2.getQuantidade());
        metodoCan4(c, c2);
        System.out.println(c.getCor());
        System.out.println(c2.getCor());
    }
    public static void metodoCan1(Caneta c) {
        c.setCor("vermelho");
        c.setQuantidade(9);
    }
    public static void metodoCan2(Caneta c) {
        c.setQuantidade(32);
        c = new Caneta();
        c.setCor("verde");
    }
    public static void metodoCan3(Caneta c) {
        c = new Caneta();
        c.setCor("laranja");
        c.setQuantidade(1);
    }
    public static void metodoCan4(Caneta c, Caneta c2) {
        c=c2;
    }
    public static void qtd(int i) {
        i=i+3;
    }
}
```

```

public class Caneta {
    private String cor;
    private int quantidade;
    public String getCor() {
        return cor;
    }
    public void setCor(String cor) {
        this.cor = cor;
    }
    public int getQuantidade() {
        return quantidade;
    }
    public void setQuantidade(int quantidade) {
        this.quantidade = quantidade;
    }
}

```

Questão 4 (0.2) – Crie uma interface chamada Ordenavel com um método void chamado ordenar(). Crie uma classe chamada Ordenadora que implementa Ordenavel. Faça com que tudo compile. Não se preocupe com saídas no console.

Questão 5 A (3,0)– Essa questão deve compilar e executar sem erros. Crie a classe abstrata CorpoCeleste. Essa classe deve ter dois atributos privados do tipo String chamados: diametro e identificador. Crie APENAS UM construtor que receba os dois atributos como argumento e atribua valores às variáveis de instância. Crie os getters e setters. Crie um método abstrato chamado getRaio() publico e que retorne um double. Não crie outros métodos em CorpoCeleste. Crie uma classe chamada Estrela, essa classe estende CorpoCeleste. Crie em Estrela os construtores que desejar. Seguindo o mesmo padrão de Estrela, crie a classe Planeta. Sabendo que o raio é a metade do diâmetro e que existe um erro clássico de cálculo de raio nas estrelas, quando retornar o raio de uma estrela, retorne o raio -0.1 (i.e. se o diametro for 200, retorne 99.9 -> $(200/2)-0.1$). Os planetas seguem a regra do diâmetro dividido por 2. Crie apenas em Planeta o método chamado public void verificarVida() que imprime no console: “verificando a existência de vida”.

Crie uma classe Principal. Faça o seguinte nessa classe: crie um ArrayList e preencha com 2 instancias de Estrela e uma de Planeta. Insira também nesse ArrayList as String “abc” e “cde”. Crie uma outra instância de Estrela e atribua a variável est e que tenha, nos atributos, os mesmos valores de umas das estrelas do arraylist (não insira essa no ArrayList). Observe a classe abaixo:

```

public class Utils {
    public static void imprime(CorpoCeleste c) {
        System.out.println(c);
    }
    public static boolean existe(ArrayList lista, CorpoCeleste c) {
        return lista.contains(c);
    }
}

```

Não mexa na classe Utils, essa classe já existe. Faça com que, quando o método `imprime(CorpoCeleste c)` for utilizado, saia no console o identificador e o diâmetro do corpo celeste, ex: `siriuskbx 200`, `solkbx 123`, etc. Da mesma forma, faça com que o método `existe(...)` funcione corretamente, sabendo que dois objetos do tipo `estrela` são iguais se ambos possuem o mesmo identificador e mesmo diâmetro.

Questão 5 B (0,8)– Crie uma nova classe chamada `Utils2` e crie o método `public static void verificarTudo(List lista)`. Esse método deve percorrer a lista e chamar o método `verificarVida()` nos planetas.

Questão 5 C (0.2) – Qual o método chamado em `ArrayList` que recebe como argumento um objeto e retorna o índice desse objeto?

Questão 5 D (0.1) - Se colocássemos a instrução abaixo dentro da classe `Principal` definida em 5 A, esse código compilaria?

```
CorpoCeleste c = new Estrela();
```

Questão 5 E (0.1)- Se colocássemos a instrução abaixo dentro da classe `Principal` definida em 5 A, esse código compilaria?

```
CorpoCeleste c = new CorpoCeleste();
```