Centro Federal de Educação Tecnológica –

**Disciplina: Programação e Desenvolvimento de Sist.**

Prof. Gustavo Guedes

****

**Aluno(a): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Turma:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Data: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Códigos desnecessários e que reduzam o desempenho do sistema serão penalizados.**

**Utilize as boas práticas de programação.**

**Utilize as boas práticas de programação, sempre que possível. Vale lembrar que, quando possível, declarar atributos de instância como privados são uma boa prática de programação.**

**LEIA AS QUESTÕES ATÉ O FINAL ANTES DE COMEÇAR.**

**Questão 1 (3,5) – Desenvolva o código conforme pedido abaixo:**

A – Escreva uma classe Animal com 3 atributos: id (String), nome (String) e idade (int). Crie os getters e setters apenas se precisar. Crie em Animal APENAS UM construtor, que recebe o id como argumento. Implemente um método em uma classe chamada Utils com a seguinte assinatura: public static boolean existe (List<Animal> x, Animal y). Escreva esse método de forma que seja verificada a existência do objeto Animal representado por y na lista representada por x, retorne verdadeiro se existir e falso se não existir. Considere que dois objetos Animal são iguais se possuem o **mesmo id**. Não é permitida qualquer iteração para realizar esse item, ou seja, não use *for*, *iterator*, etc.

B – Crie um método em Utils com a seguinte assinatura public static void ordena (List<Animal> x). Esse método deve ordenar os objetos Animal em x por ordem alfabética de nome. Prepare a classe Animal para que isso ocorra corretamente. Não é permitida qualquer iteração para realizar esse item, ou seja, não use *for*, *iterator*, etc.

C – Ao utilizar o System.out.println em um objeto Animal, deve sair no console o id, nome e idade do Animal.

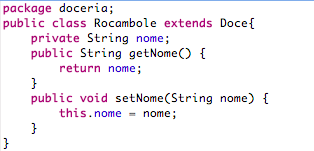
D- Crie 2 subclasses da classe Animal: Leao e Urso.

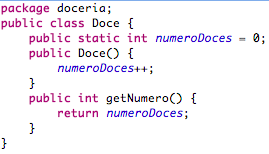
E - Dada a classe Utils, crie o método public Map<String, Animal> retornaDados(Set conjuntoAnimais), utilizando o ITERATOR nessa questão.

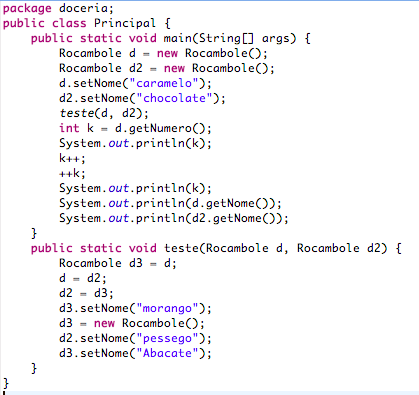
Considere que o conjunto recebido como argumento (conjuntoAnimais) contém Strings no seguinte formato: id#nome#idade#tipo. Por exemplo, considere os elementos desse conjunto como (154-3#Simba Silva#13#L, 553-2#Juju Juba#10#U, etc.). Esses valores representam id, nome, idade e tipo do animal (Leao ou Urso).

Dessa maneira, implemente o método *retornaDados* de forma que seja retornado um mapa da seguinte forma: os elementos de *conjuntoAnimais* devem ser percorridos, o id de cada elemento é a chave do Mapa e os valores do mapa são objetos do tipo Leao ou Urso. Resumindo, você irá criar um objeto Leao (se o último caractere da string for L) ou Urso representando cada elemento em *conjuntoAnimais* (se o último caractere da string for U) e adicionar ao mapa.

Questão 2 – (1.5) – Observe o código abaixo e diga o que sai no console.

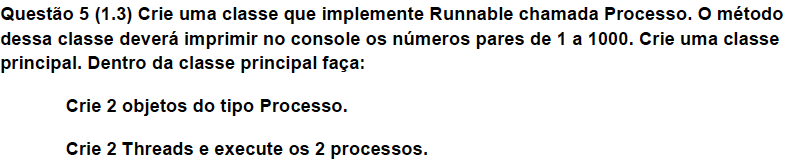






Questão 3 – (0.5) – Escreva os nomes de todos os tipos primitivos em java e todas as suas Wrapper Classes.

Questão 4 – (1.0) – Imagine uma classe X que tenha um método chamado fazerAlgo que não retorne nada, tenha visibilidade apenas dentro do mesmo pacote, que não tenha argumentos e que não possa ser sobrescrito. Escreva a assinatura desse método.



Questão 6 (1.7) – Crie uma classe chamada Substitui com um método estático que não retorna nada chamado processa (String str). Esse método recebe uma String e imprime no console a mesma String toda em caixa alta (letras maiúsculas) e com as letras A, E, I, O substituídas por 4, 3,1,0 respectivamente. Dessa forma, se a String passada for “paralelismo”, o método exibirá no console a String “P4R4L3L1SM0”.