Spring 2018- ITEC 3150 Coding Questions

Complete 4 (four) of the following 6(six) problems. Two of those problems must be 1 and 2 – binary tree and multithreading. Each problem is worth 18.75 points. You are not being graded on coding standards, but please include your name in a comment at top of each file turned in. Coding Standards will not be considered in your grade. You may do a fifth problem for extra credit ( up to 18.75 pts), but I will only give partial credit for 4 problems.

1. Binary Tree problem – File needed BST.java- REQUIRED

Using the file BST.java, modify the countRightNodes() method to correctly return the number of right nodes (nodes identified as right by their parent) in the tree. You may modify everything about countRightNodes except the method signature. You may add additional methods if desired, but you may not change any other methods within BST.java

1. **Multithreading Problem- Files needed HashSetSynchornized.java - REQUIRED**

* Use the attached file named HashSetSynchronized.java. This file currently launches two threads with empty run() methods. Create the run() methods for both Thread1 and Thread2. Both threads will access the shared HashSet named hashSet.
* Task1 should add 20 values to the hashset. It should wait 1 second between adding each value. You must make sure the code is thread safe as Task1 is not the only thread using hashSet.
* Task2 is going to read and print the values in hashSet every 1.5 seconds. Like Task1, it must access hashSet in a thread safe way. Task2 will terminate when it discovers that hashSet has 20 or more values after printing them. (be sure to put a blank line in between each iteration)
* You should not modify the main() or constructor method of the HashSetSynchronized class. Everything else may be changed

1. **HashSet- Files needed HashSetExample.java**

In the main method of HashSetExample.java, two HashSet’s of type String are defined named set1 and set2. Create a new HashSet named set3 and populate it with only the Strings that are contained in both set1 and set2. Print the contents of set3.

1. **HashMap: Files Needed: HashMapUser.java**

In the main method, create a HashMap (may use the java.util.HashMap implementation). Add the names in the initialSet array as the keys to the maps. The value associated with each name should be an int. That int will be the hashCode of the String being added as the key.

Then print the contents of the map in the format- do NOT just print the hashMap- iterate and print as shown: (hashcodes are not accurate values in the example)

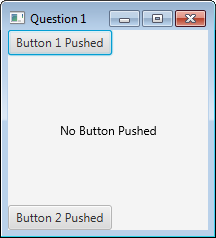
Key: Romeo, Value: -45678

Key: Juliet, Value: 78990900

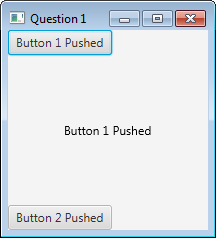
Etc

**5. JavaFX – no file provided**

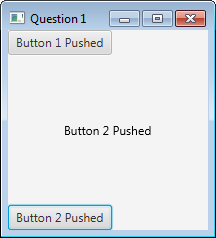
1. Create a GUI that looks like the following: (Hint: I used BorderPane and set stage size to 200, 200) (no code provided)



When Button 1 is pushed, the text is the center should change like so:



When Button 2 is clicked, the text should change to this:



**6. Binary File – Files needed Game.java**

Given the file Game.java (attached). Write a Java class with a main that creates three instances of the Game class and writes them to a binary file named games.ser

You may use either (but not both) DataOutputStream or ObjectOutputStream

Be sure to turn in the Game.java file along with your Java file and the games.ser file.