15-121 Fall 2018 Quiz 10

Up to 20 minutes. No calculators, no notes, no books, no computers. Show your work!
There are questions on *both sides* of this paper.

1. (10 points) **Code Tracing**

   Determine the output of the following program. Write your answer, and nothing else, in the box below the code.

```java
public class CT {
    public static int ct(Map<String, Integer> m) {
        Set<String> set = m.keySet();
        int y = 0;
        for (String s : set) {
            int x = s.length();
            if (s.charAt(x - 1) == 's') {
                x = 3 * x / 2;
            }
            int z = m.get(s) % x;
            System.out.println(z);
            y += z;
        }
        return y;
    }

    public static void main(String[] args) {
        TreeMap<String, Integer> tm = new TreeMap<String, Integer>();
        tm.put("cat", 10);
        tm.put("dog", 9);
        tm.put("horse", 8);
        tm.put("monkeys", 7);
        System.out.println(ct(tm));
    }
}
```

2. **Short Answer**

   (a) (2 points) What are the two main practical differences between a `TreeSet` and a `HashSet`?

   (b) (2 points) The theoretical worst case complexity of a hash table is $O(N)$, but in practice we treat them as $O(1)$. Why?

   (c) (3 points) Recall the `Person` class from HW2. It has a `firstName`, `lastName`, and `age`. Imagine Bob writes the following `hashCode` method for a `Person` class:

   ```java
   public int hashCode() {
       return firstName.hashCode();
   }
   ```

   Is this a good `hashCode` function or a bad `hashCode` function? Justify your answer.

   (d) (3 points) Write a good Javadoc comment for the the `findMatches` method of `AnagramTree` in HW6. For reference, its signature is: `public ArrayList<String> findMatches(String word)`