Practice Midterm 2 – CMPS 12A Fall 2017

1.) Determine the output of the following program.

```java
public class Question1{
    public static void main(String[] args){
        Car x = new Car("gray", 100000);
        Car y = new Car("red", 125000);
        Car z = new Car("blue", 150000);

        x = y;
        roadTrip(x);
        z = x;
        System.out.println(z.mileage);
        System.out.println(y.color);
    }
    public static void roadTrip(Car c){
        Car a = c;
        a.drive();
        c.drive();
        c.drive();
        c.drive();
    }
}
public class Car{
    String color;
    int mileage;
    Car(String color, int mileage){
        this.color = color;
        this.mileage = mileagae;
    }
    void drive(){
        mileage++;
    }
}
```

2.) Write a class called `Rational`. `Rational` will consist of a numerator and a denominator – both fields are private. `Rational` will also have a method called `times()` that will multiply itself by another `Rational` number, and return the result as a `Rational` number. It is not necessary to reduce fractions to lowest terms.

3.) Write a method `insert()` that accepts the following parameters: an array of integers *a*, an integer index *n*, and an integer to be inserted *val*. This method should return an array of integers with the integer inserted at the appropriate index, preserving the order of the remaining elements of the array.

```
public class Question3{
    public static void main(String[] args){
        int[] x = {2, 4, 5, 6, 12, 15, 21};
        x = insert(x, 3, 100); // insert value 100 at index 3
    }
    // your method goes here




}
```

4.) Determine the output of the following program.

```
public class Question4{
    public static void main(String[] args){
        Driver a = new Driver("Jake", 45);
        Driver b = new Driver("Elwood", 47);
        Driver c = new Driver("Steve", 40);
        Driver d = new Driver("Duck", 35);
        System.out.println(a.numDrivers);
        System.out.println(a.ID);
        System.out.println(d.numDrivers);
        System.out.println(d.ID);
    }
}
public class Driver{
    int ID;
    static int numDrivers = 0;
    String name;
    int age;
    Driver(String name, int age){
        this.name = name;
        this.age = age;
        numDrivers++;
        ID = numDrivers;
    }
}
```

5.) Write a *recursive* method `sum()`, that takes a single `int` *n* as input and returns the sum of the positive integers from 1 to *n*.

```
public static int sum(int n){ . . . }
```

6.) Write a method, `collate()`, that accepts two integer arrays as parameters (you can assume these arrays are the same size), and returns an array of the collated values of the two integer arrays accepted. The collated array of arrays `x` and `y` will result in an array containing values `{x[0],y[0],x[1],y[1],...,x[n],y[n]}`. For example, for arrays `x:{1,3,5}` and `y:{2,4,6}`, a call to `collate(x,y)` will return array `{1,2,3,4,5,6}`.

7.) Determine the output of the following program.
```
public class Question7 {
    public static void main(String[] args) {
        Node list = null;
        for (int i = 5; i < 10; i++){
            list = new Node(i, list);
        }
        mystery(list);
    }
    static void mystery(Node list) {
        if (list != null) {
            mystery(list.next);
            System.out.println(list.data);
        }
    }
}
public class Node{
    int data;
    Node next;
    Node(int d, Node n){
        data = d;
        next = n;
    }
}
```

8.) Write a class called `RandomArray`, that has one instance variable: an array of integers that holds 5 values. The class's constructor should accept no parameters, and should initialize all elements in the array to random values between 1 and 50. The class should have the following instance methods:

        `highestIndex()` – returns the index of the highest value in the array.
        `lowestIndex()` – returns the index of the lowest value in the array.
        `average()` – returns the average of the elements of the array.

9.) Write a method bubbleSort() below such that an array of doubles is accepted as a parameter, the contents are sorted in ascending order (you are not allowed to use Arrays class), and returns the sorted array.

```
public class Question9{
    public static void main(String[] args){
        double[] x = {3.4, -2.3, 5.5, 1.2, -4.3, 3.1, 0.1, 4.2};
        x = bubbleSort(x);
    }

    // Your method goes below










}
```

10.) Write a static method equals() that accepts two Student objects as parameters, and returns true if the instance variables of the two students objects are the same, and false otherwise. Use the following Student class for reference.

```
public class Student{
    String name;
    int ID;
    double GPA;
    Student(String n, int i, double g){
        name = n;
        ID = i;
        GPA = g;
    }
}
```
// your static method goes below