CS203 Java Programming and Application
Fall, 2016

Assignment 4
Assigned Date: Sunday, November 13, 2016
Due Date: Midnight Sunday, November 27, 2016

QUESTION 1 (20 MARKS)

Assignment 3 Question 1 defined the Triangle class with three sides. In a triangle, the sum of any two sides is greater than the other side. The Triangle class must adhere to this rule. Create the IllegalTriangleException class, and modify the constructor of the Triangle class to throw an IllegalTriangleException object if a triangle is created with sides that violate the rule. The constructor of IllegalTriangleException must encapsulate all three sides of the triangle and a string message, as follow:

```
public IllegalTriangleException(double side1, double side2, double side3, String msg)
```

Write a test program to test your IllegalTriangleException by creating 2 instances of triangle with one of them violating the rule. Print the perimeter and area of the legal triangle. Print the sides and string message of the illegal triangle from the IllegalTriangleException caught.

QUESTION 2 (20 MARKS)

Write a program to create a binary data file named RandomInt.dat that consists of 10 random integer numbers using DataOutputStream. Use DataInputStream to read the integers from RandomInt.dat just created and print them on the console in the same program.

QUESTION 3 (20 MARKS)

Design an interface named Drawable with a void method named draw(). Every class of a drawable object must implement the Drawable interface. Design a class named Rectangle that implement the Drawable interface. You may define a new Rectangle class or use the Rectangle class from Assignment 3. In order to draw a rectangle the draw( ) method must know the height and width of a rectangle. Implement the draw( ) method in your Rectangle class to draw a filled rectangle with "*".

Write a test program called Artist that contains a drawObject( ) method. The drawObject( ) method can draw any object of type Drawable. Call the drawObject( ) method by passing in the Rectangle object. Show that the Rectangle object can be drawn by the drawObject method.

A sample run of the program is as follow:

Drawing Rectangle Object:

```
**********
**********
**********
**********
**********
**********
**********
**********
```

QUESTION 4 (20 MARKS)

The Animal classes are given for this exercise. Modify the Animal class by implementing the Comparable interface and Cloneable Interface. Add the weight property in the Animal class with getter and setter methods. The two animals are compared based on their weights.

Use the following main method to run your test program.

```
public static void main(String[] args) {
  Animal[] list = new Animal[5];
  list[0] = new Chicken();
  list[0].setWeight(4.5);
  list[1] = new Tiger();
  list[1].setWeight(46.6);
  list[2] = new Chicken();
  list[2].setWeight(1.5);
  list[3] = (Animal)(list[0].clone());
  list[3].setWeight(7.5);
  list[4] = (Animal)(list[1].clone());

  java.util.Arrays.sort(list);
  for (int i = 0; i < list.length; i++) {
    System.out.println("weight: " + list[i].getWeight());
  }
}
```

QUESTION 5 (20 MARKS)

(i) A company maintains some of its item price in a random access file. The file stores the item number and its price. Write a program that creates 10 blank records first and then allow user to enter any item price using any item number from 1 to 10. Insert each new record into a file at a location that is equal to the item number. The program stops when user input -999 as item number. Your program should also check that the item number entered is between 1 and 10 and that the price cannot be negative. Save your program as **CreateItemRecord.java**

(ii) Write a program that uses the file created in part(i) and displays all existing item numbers and their price order by item number. Save your program as **DisplayRecordSequentially.java.**

(iii) Write a program that uses the file created in part(i) and allow user to check an item price by entering an item number. User should be allowed to check any item price randomly. The program stops when user enters -999 as item number. You may assume in this program that user always enter correct item number. Save your program as **AccessRecordRandomly.java**