CMSC 132 Quiz 1 Worksheet

The first quiz for the course will be on Wed., June 7th. The following list provides additional information about the quiz:

- The quiz will be a written quiz (no computer) and you must be on campus to take it.
- **The quiz can start at any time from 6 to 7 PM.**
- Closed book, closed notes quiz.
- Answers must be neat and legible.
- The quiz has a maximum duration of **20 minutes.**
- Bring your student id.
- Please do not discuss a quiz after you have taken it or post anything about it on piazza.
- **Regarding Piazza -** Feel free to post questions in Piazza regarding the worksheet and possible solutions to problems.

The following exercises gives you practice with concepts that <u>may show up</u> on the quiz. Solutions to these exercises will not be provided, but you are welcome to discuss your solutions with the TAs during office hours. It is recommended that you try these exercises on paper first (without using a computer).

Exercises

1. Review CMSC 131 topics related to class design. See review examples from lab1 of first week.

2. Define a class Telephone according to the following information:

   Instance Variables (all private)

       a. area code → integer value
       b. three digit value → integer value
       c. four digit value → integer value
       d. user name → String reference

   Instance Methods

       a. **Constructor** - Allows you to initialize all the instance variables of the class. Name the parameters after the instance variables (i.e., you must use the **this** reference)
       b. **Default constructor** – Initializes the object to the number 555-555-5555 and the name to null. This constructor relies on the previous constructor for the object initialization (i.e., you must use the **this** reference)
       c. **Copy constructor**
       d. **Get/Set methods** – Define get/set methods for all instance variables of the class.
       e. **equals** – Two numbers are considered the same if they have the same area code, three and four digit values. Use instanceof to implement this method (the approach described in class).
       f. **toString –** Returns a string with the user name followed by the phone number of the person.

   Static Variable (private)

       a. **count** – keeps track of how many Telephone objects has been created.

   Static Method

       a. **getCount** – Returns the count value
       b. **getDigits** – Takes a String reference as a parameter and returns the number associated with the string. For example, if the String has the value "CAR" the method will return the integer 227

3. What is encapsulation? How does it relate to abstraction?

4. What is the difference between procedural abstraction and data abstraction?

5. Do you know examples of academic integrity violations? The following is the list you need to know:

       a. Hardcoding of results in a project assignment.
       b. Using any code available on the internet/web.
       c. Hiring any online service to complete an assignment for you.

d. Posting your implementation of any class project on the internet/web.
e. Discussing projects with your classmates.
f. Sharing your code or your student tests with any student.
g. Looking at another student's code.

6. Define an enumerate type named **Day** that represents the days of the week. Using the enhanced for loop construct, write a code fragment that prints all the days of the week.

7. The **PrinterJob** class is defined as follows:

```java
public class PrinterJob {
    private int id;
    private int size;

    public PrinterJob(int id, int size) {
        this.id = id;
        this.size = size;
    }

    public int getId() { return id; }

    public int getSize() { return size; }

    public String toString() {
        return "Id: " + id + " Size: " + size;
    }

    public int hashCode() {
        return id;
    }
}
```

a. Add a new private field "jobType" as an enumerated type with the values COLOR and BW.

b. Modify the class so it implements the Comparable interface, allowing you to compare PrinterJob objects based only on their id.

8. Review interfaces and understand that an interface creates a new type that can be assigned objects of the class that implement the interface (**is a** relationship).

9. Review the for each loop (enhanced for loop).

10. Review difference between reference copy, shallow copy, and deep copy.

11. Old Quiz 1 from past terms:

http://www.cs.umd.edu/class/summer2023/cmsc132/quizzes/OldQuiz1.zip

DO NOT JUST LOOK AT THE SOLUTION. You will not learn anything from that. Print out the quiz, complete it in 20 minutes without using eclipse, notes, etc. Check your answer using the key.