1.00 Introduction to Computers and Engineering Problem Solving

Quiz 1 March 5, 2004

Name:	
Email Address:	
TA:	
Section:	

You have 80 minutes to complete this exam. For coding questions, you do not need to include comments, and you should assume that all the necessary files have already been imported. Good luck.

Question	Points
Question 1	/ 10
Question 2	/ 15
Question 3	/ 20
Question 4	/ 20
Question 5	/ 17
Question 6	/ 18
Total	/ 100

Question 1. True/False (10 Points)

Answer the following questions by circling TRUE or FALSE as appropriate.

1) It is legal to have two methods whose signatures differ only by their return types.

   TRUE   FALSE

2) Because arrays are passed to methods as references, the called methods can modify the element values in the caller's original arrays.

   TRUE   FALSE

3) `final` variables must be initialized before they are used and cannot be modified thereafter.

   TRUE   FALSE

4) When an object `a` of a class `Test` has a reference to another object `b` of the same class, `a` can access all of `b`'s data and methods.

   TRUE   FALSE

5) If a method `m()` defines a local variable `a` with the same name as a variable with class scope, there is no way to access the instance variable from inside `m()`.

   TRUE   FALSE

Question 2. Objects and Variables (15 Points)

```
public class Robot
{
    private static int ID = 0;
    private int serialNumber;
    private String name;
    private int myID;

    public Robot(String robotName, int sNum)
    {
        serialNumber = sNum;
        name = robotName;
        myID= ID++;
    }
    public static int getID(){return ID;}

    public int getSerial(){return serialNumber;}

    public void setSerial(int sNum){serialNumber = sNum;}

    public int getMyID() { return myID;}
}
```

In order to test the above class, you write the following:

```
public class Test
{
    public static void main(String[] args)
    {
        System.out.println("Current ID: " + Robot.getID());
        Robot robot1 = new Robot("Rod", 3239);
        System.out.println("Robot 1 ID: " + robot1.getID() +
                            " Serial: " + robot1.getSerial());
        Robot robot2 = new Robot("Jon", 2364);
        System.out.println("Robot 2 ID: " + robot2.getID() +
                            " Serial: " + robot2.getSerial());
    }
}
```

What is the output of this program? (Program compiles and runs correctly and do not worry about white space.)

You decide to change the test code to the following:

```
public class Test
{
   public static void main(String[] args)
   {
1:     System.out.println("Current ID: " + Robot.getID());
2:     System.out.println("Current ID: " +
                          Robot.getSerial());
3:     Robot robot1 = new Robot("Rod",3239);
4:     System.out.println("Current ID: " + Robot.getID() +
                          " Serial: " + robot1.getSerial());
   }
}
```

You are about to compile the new test program. Please answer the following questions.

Will line 1 compile? If not, why?

_____

Will line 2 compile? If not, why?

_____

Will line 3 compile? If not, why?

_____

Will line 4 compile? If not, why?

_____

Question 3. Write some code (20 Points)

A number is said to be perfect if it is equal to the sum of its divisors except itself. For example, 6 and 28 are both perfect numbers because:

$6 = 1 + 2 + 3$

$28 = 1 + 2 + 4 + 7 + 14$

In the space below, complete the method `isPerfect()`, which returns `true` if its argument is a perfect number, and `false` otherwise. Assume the method correctly receives a positive integer argument `n`.

```
public static boolean isPerfect(int n)
{




















}
```

Question 4. Java Classes and Methods (20 Points)

Concerned that the glorification of violence in computer games such as Grand Theft Auto is contributing to urban crime, you begin writing a new, text-based computer game called Vigilantes of Benevolence (VoB). In VoB, the Player drives around committing random acts of kindness. Points are awarded for hugging citizens, picking up litter, administering CPR (Cardiopulmonary Resuscitation), and so forth. At a cost of 100 points, a Player can erect a homeless shelter. When a Player has built 10 homeless shelters, he or she wins the game. Conveniently, Vigilantes of Benevolence is implemented in Java. Your task is to complete the `Player` class in the spaces provided below. There are two parts, labeled (1) and (2).

```java
/* Represents the player's on-screen character */

public class Player
{
   private int points;
   private static boolean gameOver;

   // Number of homeless shelters owned by this Player
   private int homelessShelters;

   // Creates new Player with zero points and zero shelters
   public Player()
   {
      points = 0;
      homelessShelters = 0;
   }

   public void buildHomelessShelter()
   {
      // Part (1)
   }

   public void applyCPR(Citizen c)
   {
      // Part (2)
   }
}
```

```
/* Represents Citizens of the city our Player is perusing*/

public class Citizen
{
   private String name;

   // Is the citizen presently having a heart attack?
   private boolean cardiacArrest;

   public Citizen(String n)
   {
      name = n;
      cardiacArrest = false;
   }

   public boolean getCardiacArrest()
   {
      return cardiacArrest;
   }

   public void setCardiacArrest(boolean ca)
   {
      cardiacArrest = ca;
   }
}
```

1) Complete the method `buildHomelessShelter()`, which allows the Player to build a homeless shelter at the cost of 100 points. If a Player has enough points to build a shelter, `points` and `homelessShelters` are adjusted accordingly. If the Player is found to have 10 or more `homelessShelters`, the method outputs "YOU WIN!" and sets the `boolean gameOver` to `true`.

```
public void buildHomelessShelter()
{




















}
```

2) The following method, `applyCPR()`, allows the Player to apply CPR to a Citizen. If the Citizen is in cardiac arrest, the method should end his or her heart attack and give the Player 25 points. If the Citizen is **not** having a heart attack, he or she will probably not appreciate the CPR, and 5 points should be subtracted from the Player's total.

```
public void applyCPR(Citizen c)
{




































}
```

Question 5. Arrays and Vectors (17 Points)

Complete a static method, `vectorToArray()`, which takes a `Vector` of `Integer` objects as an argument and returns an array of `ints` with the same values. Since `Vectors` hold only objects (not primitives), you need to extract the `int` value from each `Integer` object in `Vector`.

For this question, you will find the `size()`, `get()` and/or `elementAt()` methods of the `Vector` class useful, as well as the `intValue()` method of the `Integer` class:

- `size()` returns the current number of elements in a `Vector`
- `elementAt(int a)` or `get(int a)` returns element of `Vector` at index a
- `intValue()` returns the value of the integer primitive value in the `Integer` object

The order of elements must remain the same.

```
public static int[] vectorToArray(Vector iVector)
{




















}
```

Question 6. Recursion (18 Points)

Given the method below:

```
public static int mystery(int x, int y)
{
   if (y == 0)
      return x;
   else
      return mystery(y, x % y);
}
```

Please answer the following three questions:

1) If the method is invoked with arguments x = 8 and y =12, what will it return?

2) If the method is invoked with arguments x = 4 and y =5?

3) Based on the results above, what does the method do in general? Your answer should
   be of the form: "this method computes the base b logarithm of a" or "this method
   raises a to the power b".