Ahmadu Bello University
Department of Mathematics
First Semester Examinations – Month 2013
COSC211: Introduction to Object Oriented Programming I

Attempt **ALL** the questions Time: **120** mins

1. Examine the following Java code and answer the questions that follow.
[Note that line numbers are not part of the code and are for reference only.]

```
 1.  //Tree.java
 2.
 3.  public class Tree{
 4.     private double girth;
 5.     private double height;
 6.
 7.     public Tree(double girth, double height){
 8.        setGirth(girth);
 9.        setHeight(height);
10.     }
11.
12.     public Tree(){
13.        this(0.01, 0.05);
14.     }
15.
16.     public double getGirth(){
17.        return girth;
18.     }
19.
20.     public double getHeight(){
21.        return height;
22.     }
23.
24.     public void setGirth(double girth){
25.        if(girth < 0.01) girth = 0.01;
26.        this.girth = girth;
27.     }
28.
29.     public double setHeight(double height){
30.        if(height < 0.05) height = 0.05;
```

```
31.         this.height = height;
32.      }
33.
34.      public String toString(){
35.         return String.format("Girth %8.2f\t" +
                "Height %8.2f", girth, height);
36.      }
37.   }//end of class Tree
```

(*a*) The class has two fields, both *private*. Explain why such fields are normally kept private.

(*b*) There are two *constructors*, one on lines 7 to 10 and one on lines 12 to 14. Describe two characteristics that these constructor methods have.

(*c*) When there is more than one constructor defined in a class how does the JVM distinguish between them? Explain the term *signature*.

(*d*) There are two *getter methods*. Identify them and explain what they are for.

(*e*) There are two cases of *data validation* performed by this code. What lines are they on and how do they work?

(*f*) The keyword `this` is used in two different senses. First on line 13 it is used in a constructor. Second on lines 26 and 30 it is used in a mutator. Explain each of these usages fully.

(*g*) There is a `toString()` method. What is it used for? What characteristics must a `toString()` method have?

(*h*) Write the Java source code for a program that will instantiate two `Tree` objects and display their field values.

2. The following code, stored in the files *Worker.java* and *WorkerPay.java*, has ***eight*** errors in it.
[Note that line numbers are not part of the code and are for reference only.]

```
1.   //Worker.java
2.
```

```
 3.  public Worker{
 4.     private static final double HOURLY_PAY =
            2000.00;
 5.     private static final int BASE_HOURS = 40;
 6.     private hours;
 7.
 8.     public Worker(int hours){
 9.        setHours(hours);
10.     }
11.
12.     public void setHours(int hours){
13.        if(hours > 70) hours = 70
14.        this.hours = hours;
15.     }
16.
17.     public double calculatePay(){
18.        double pay;
19.        if hours > BASE_HOURS {
20.           pay = BASE_HOURS * HOURLY_PAY;
21.           pay += (hours - BASE_HOURS) *
                 HOURLY_PAY * 1.5;
22.        }else{
23.           pay = hours * HOURLY_PAY;
24.        return pay;
25.
26.  }//end of class Worker
```

```
 1.  //WorkerPay.java
 2.
 3.  public class WorkerPay{
 4.      public void main(String[] args){
 5.          Worker worker = new Worker(56);
 6.          System.out.printf("Pay is %.2f\n",
                worker.calculatePay());
 7.      }
 8.  }//end of class WorkerPay
```

(*a*) Locate each of the ***eight*** lines that contain an error and rewrite it correctly.

(*b*) Describe the purpose of the method `calculatePay()` on lines 17 to 25

and explain how it works. (In answering this refer to the code you have corrected since the original may have errors.)

(*c*) Reproduce the screen display created when running the program *WorkerPay.class*, (compiled after the corrections have been applied).

3. (*a*) Explain ***in full*** the effect of the following lines of code if they appeared in a `main()` method..

(*i*) `Dog myDog = new Dog();`
(*ii*) `int myNum = 27;`
(*iii*) `String myName = "Fatima Abubakar";`
(*iv*)
```
if(height >= 2.0)
     System.out.println("You are tall.\n");
else
     System.out.println("You aren't tall.\n");
```

(b) There are two principal data types in the Java Programming Language: *primitive data types* and *objects*. Distinguish carefully between them.

(c) How many bits of storage do the following data types require: `int`, `byte`, `short`, `long`, `float`, `double`.

4. (a) Write the code for a class called `Course` that has the following private fields: `code` (`String`); `title` (`String`), `level` (`int`) and `semester` (`int`).

There should be a constructor that sets the field values, passed as arguments. There should also be a *no-args* constructor that gives each field a suitable default value.

There should be *accessor* and *mutator* methods for each field. The mutator methods for `level` and `semester` should include suitable *validation checks*.

There should be an appropriate `toString()` method.

(b) Now write the code for a program that will instantiate the class you created in (a) above. The field values should be "COSC211", "An Introduction to object-oriented programming I", 200 and 1, repectively. The program should display these field values.

5. (*a*) Some methods return a value, others do not. Describe the differences between them in (*i*) how the methods are defined, and (*ii*) how the methods are called.

(*b*) Design a method that will take three integer arguments and return the smallest.

(c) Why is the `main()` method declared to be static.

6. (*a*) The following statements appear at the very start of a Java file.

```
import javax.swing.JFrame;
import static java.lang.Math.*;
```

Explain what they are for.

(*b*) Write a Java program that will prompt the user for the size of the angles A and B, and the length of the side BC of a triangle ABC. The program should calculate and display the length of the side AC. [Sine rule.]