RAJARATA UNIVERSITY OF SRI LANKA
FACULTY OF APPLIED SCIENCES

B.Sc. (General) Degree in Information and Communication Technology
First Year Semester II Examination April/May 2015

ICT 1306 OBJECT ORINTED PROGRAMMING

Time Allowed: 3 hours.

INSTRUCTIONS TO CANDIDATES

- This paper contains five (05) questions on 10 pages. Answer **All** Questions.
- This examination accounts for 60% of the course assessment. The total maximum mark attainable is 100. The marks assigned for each question and section, thereof are indicated in square brackets.
- This is a closed book examination.
- Mobile phones or any other communication devices are not permitted.
- Clearly state the assumptions you make. If you have any doubts regarding the interpretation of the wording of a question, make your own decision, but clearly state it on the script.

1 (a) What is the software that you need to install on your computer before you run a java application on it?

[1 mark]

(b) Briefly describe two (02) advantages and two (02) disadvantages of OOP.

[4 marks]

(c) Below is a Java class named ***helloWorld***.

```
public class helloWorld
{
        public static void main (String[] args)
        {
                System.out.println("Hello World!");
        }
}
```

Name of the above class is "***helloWorld***". It is starting with a lowercase letter. Do you think it is going to affect the program when you compile and run it? Briefly describe your answer.

[3 marks]

(d) Will the bellow program compile successfully? Will it run successfully?

```
class D {
        void printThis() {
                System.out.println("D");
        }
        public static void main(String[] args){
                D d = new D();
                d.printthis();
        }
}
```

[2 marks]

(e) ***"Student"*** class should be a class which is accessible to anyone. But the variables inside ***"Student"*** class should only be accessible within ***"Student"*** class. Re-write this class using encapsulation principle.

```
class Student
{
        String name;
        int ID;
}
```

[4 marks]

(f) (i) Do the bellow classes ***A*** and ***B*** compile successfully? Assume both classes are in the same folder, and compile class ***A*** first and then class ***B***.

```
class A
{
        private String name;

        public void setName(String name)
        {
                this.name = name;
        }

        protected  String getName()
        {
                return name;
        }
}
```

```
class B
{
// this method calls the methods in A
        void setName()
        {
                A a = new A();
                a.setName("Nimal");
        }

// this method class the methods in A
        void getName()
        {
                A a = new A();
        System.out.println(a.getName());
        }
}
```

(ii) Write a java class (with a main method) that calls the ***setName()*** and ***getName()*** methods in class ***B***.

(iii) Write the output of the above completed program 1. (f) (ii).

[1+3+2 marks]

2 (a) This program does not compile because it contains an error. Remove the error and re-write the program. Do not remove or add any method. The corrected program should print "I'm TestOne" when it runs.

```
protected class TestOne
{
        void say()
        {
                System.out.println("I'm TestOne");
        }
        public static void main(String[] args)
        {
                TestOne to = new TestOne();
                to.say();
        }
}
```

[2 marks]

(b) This program compiles and runs without errors. Write down the output of this program.

```
class A {
        private String name;

        public void setName(String name){
                this.name = name;
        }
        protected  String getName()   {
                return name;
        }
}

class AB_Demo {
        void setName(){
                        A a = new A();
                        a.setName("Nimal");
                }
        void getName(){
                        A a = new A();
                        System.out.println(a.getName());
        }
        public static void main(String[] args ){
                AB_Demo ab = new AB_Demo();
                ab.setName();
                ab.getName();
        }
}
```

[2 marks]

(c) Bellow program compiles and runs without errors. Write down its output.

```
class Test
{
        int a;
        static int b;

        Test(int a, int a)
        {
                this.a = a;
                this.b = b;
        }
}
```

```
class TestTwo
{
        public static void main(String[] args)
        {
                Test t1 = new Test(4, 5);
                Test t2 = new Test(8, 9);
                Test t3 = new Test(10, 11);


                System.out.println(t1.a);
                System.out.println(t1.b);
                System.out.println(t2.a);
                System.out.println(t2.b);

                t1 = t3;
                System.out.println(t1.a);
                System.out.println(t1.b);
                System.out.println(t3.a);
                System.out.println(t3.b);

                t2 = t1;

                System.out.println(t1.a);
                System.out.println(t1.b);
                System.out.println(t2.a);
                System.out.println(t2.b);
        }
}
```

[6 marks]

(d) Briefly describe the use of "***final***" keyword in java using examples.

[3 marks]

(e) Compare and contrast an interface with an abstract class. You may use examples.

[3 marks]

(f) Can the following two instance methods exist inside the same class (can they be instance methods of the same class)? Justify your answer.

```
private void printThis(int a){}
public int printThis(int b){return 5;}
```

[2 marks]

(g) Bellow program compile and runs without errors. Write down the output of the following program.

```
class C {

        public void printThis(String value){
                System.out.println(value);
        }
}

class E extends C {

        public void printThis(String value){
                System.out.println("value is : "+value);
        }
}

class TestThree {

        public static void main(String[] args)
        {
                C c = new C();
                c.printThis("C");
                c = new E();
                c.printThis("E");
        }
}
```

[2 marks]

3 (a) Identify and explain the different data types in the following ***Student*** class.

```
class Student
{
        String name;
        int ID;
        Student student;

}
```

[6 marks]

(b) Explain what would happen if you try to convert "***int***" in to a "***float***" and vice versa? You can use examples.

[4 marks]

(c) Bellow program doesn't contain any error. Write down its output.

```
class Animal
{
        public String name;
}

class AnimalDemo
{

        private Animal setName(String name, Animal animal)
        {
                animal.name = name;
                System.out.println(animal.name);
                return animal;
        }

                public static void main(String[] args)
        {
                AnimalDemo ad = new AnimalDemo();

                Animal cat = new Animal();
                cat.name = "cat";
                cat = ad.setName("paul", cat);
                System.out.println(cat.name);
        }
}
```

[2 marks]

(d) Explain the advantageous and disadvantages of Inheritance using examples.

[5 marks]

(e) The following class does not contain any syntax errors (it compiles correctly). But it does not follow java best practices correctly. Explain what is wrong with it.

```
abstract class SimpleCalculator
{
        float managerSalary(float basic, int duration)
        {
                return basic+basic*duration*5/100;
        }

        float workerSalary(float basic, int duration)
        {
                        return basic+basic*duration*20/100;
        }
}
```

[3 marks]

4 (a)

```
class Vehicle
{
        public String name;
}

class Car extends Vehicle
{
        void move(){}

}

class VehicleDemo
{
Vehicle vehicle;
private Vehicle setName(String name, Vehicle vehicle)
        {
                vehicle.name = name;
                System.out.println(vehicle.name);
                return vehicle;
        }
}
```

(i) Explain whether ***Vehicle*** class and ***Car*** class are tightly coupled or loosely coupled.

(ii) Explain whether ***Vehicle*** class and ***VehicleDemo*** class are tightly coupled or loosely coupled.

(iii) Explain whether tightly coupling or loosely coupling is better.

[2+2+4 marks]

(b) Explain the concept of polymorphism using examples.

[4 marks]

(c) Program below does not contain errors. Write down the output of the following program.

```
class Bike{
        void speed() {
          System.out.println("bellow 100km");
        }
}
class RacingBike extends Bike{
        void speed() {
                System.out.println("bellow 220km");
        }

        public static void main(String args[]){
        Bike b = new RacingBike();
        b.speed();
        }
}
```

[2 marks]

(d) Explain the advantages of Encapsulation.

[3 marks]

(e) The following ***Bicycle*** interface does not compile successfully. Explain what the problem is and re-write the correct program.

```
interface Bicycle
{
void speed()
        {
                System.out.println("any speed");
        }
}
```

[3 marks]

5 (a) You are developing system for the following scenario.

"You are developing a system for a sales record system for a retail shop. The customers buy their goods and at the exit they need to go to the cashier and pay for their items.

Cashier inserts their purchased items to system and a bill will be auto generated. With each purchase the shops inventory will also be updated. The manager and owner of the shop can also view the sales record and the inventory"

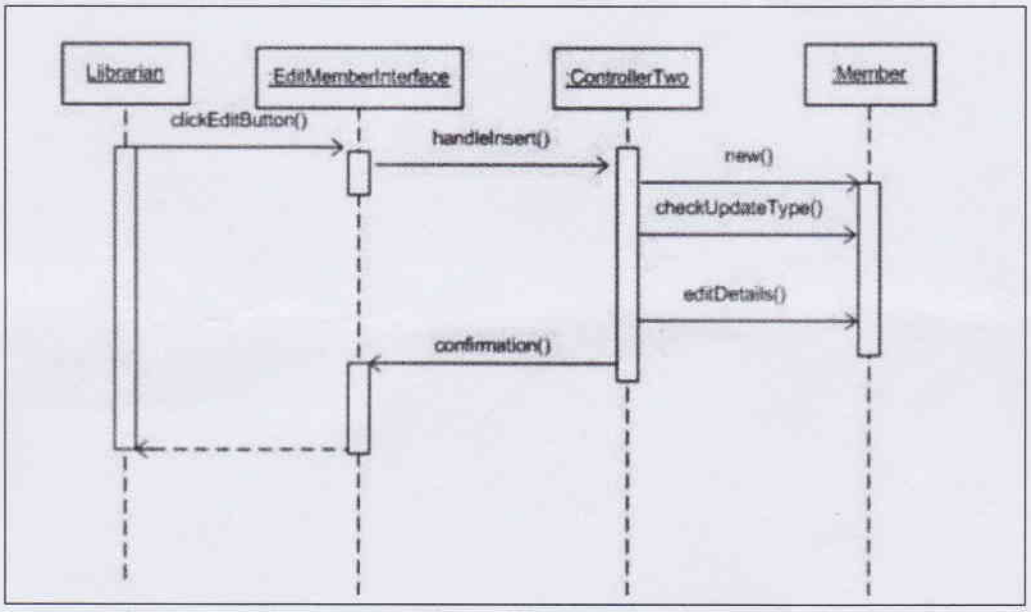
List down the possible actors in this system.
List down the possible use cases in this system.

[4 marks]

(b) Identify and list down the classes which will be used in the above 5 (a) scenario. Draw a simple class diagram (using those identified) for the above 5 (a) scenario. Clearly identify the classes and relationships among them.

[4+6 marks]

(c)



Identify and list down the classes used in the above 5 (c) sequence diagram. Also identified the methods which are used and list them down under the appropriate class.

[3+3 marks]