A. Understanding Code Segments

[15 Marks]

For each of the code segments below, write the output of the main method in the respective Driver Class.

1. [3 marks]

```java
class Test {
    protected int x, y;
}

class Main {
    public static void main(String args[]) {
        Test t = new Test();
        System.out.println(t.x + " " + t.y);
    }
}
```

2. [3 marks]

```java
public class A extends B {
    public static String sing()
    {
        return "fa";
    }
    public static void main(String[] args)
    {
        A a = new A();
        B b = new A();
        System.out.println(a.sing() + " " + b.sing());
    }
}
class B {
    public static String sing()
    {
        return "la";
    }
}
3. [5 marks]

abstract class Vibrate
{
    static String s = "-";
    Vibrate()
    {
        s += "v";
    }
}

public class Echo extends Vibrate
{
    Echo()
    {
        this(7);
        s += "e";
    }
    Echo(int x)
    {
        s += "e2";
    }
    public static void main(String[] args)
    {
        System.out.print("made " + s + " ");
    }
    static
    {
        Echo e = new Echo();
        System.out.print("block " + s + " ");
    }
}

4. [4 marks]

interface Animal {
    public void eat();
    public void sound();
}

interface Bird {
    int numberOfLegs = 2;
    String outerCovering = "feather";
    public void fly();
}

class Eagle implements Animal, Bird {
    public void eat() {
        System.out.println("Eats reptiles and amphibians.");
    }
    public void sound() {
        System.out.println("Has a high-pitched whistling sound.");
    }
    public void fly() {
        System.out.println("Flies up to 10,000 feet.");
    }
class TestEagle {
    public static void main(String[] args) {
        Eagle myEagle = new Eagle();

        myEagle.eat();
        myEagle.sound();
        myEagle.fly();

        System.out.println("Number of legs: " + Bird.numberOfLegs);
        System.out.println("Outer covering: " + Bird.outerCovering);
    }
}

B. Coding in Java [20 Marks]

Consider this UML class diagram showing part of a program to manage the membership information for a professional society:

(a) Write Java source code of class ManagementCttee assuming it has this constructor: public ManagementCttee(). [4 marks]
(b) Class Member is an abstract class. Explain the role of an abstract class. [2 marks]
(c) Write a Java source code of class Member assuming it has this constructor: public Member(String name, String address) and that the method getFee() is abstract. [4 marks]
(d) Write a Java source code of class StandardMember assuming it has this constructor: public StandardMember(String name, String address) and the standard membership fee is fixed at LKR 1500. [3 marks]
(e) Write a Java source code of class SeniorMember assuming it has this constructor: public SeniorMember(String name, String address, int fee) where the membership fee is set when a SeniorMember object is created. [3 marks]
(f) Write a Java source code of class Society assuming it has this constructor: public Society(String societyName). [4 marks]
C. Solving Problems: the Object-Oriented way

Read the given passage and design a solution to the said problem using Object Oriented Programming Concepts.

Consider a veterinary hospital that treats cats and dogs. The hospital maintains a list of owners and vets; both must contain a name. The owners also store an address; the vets store the license. An owner may have several pets; for each pet the system should store the pet birth date, the pet name, the pet ID and the pet owner. Whenever a pet arrives to the hospital it is seen by a vet; in this appointment the pet may receive a treatment. A treatment should identify the disease diagnosed, the medicine to apply and the date of treatment.

* * * * * *