United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

Midterm Exam, Trimester: Spring 2024

Course Code: CSE-1115, Course Title: Object Oriented Programming

Total Marks: 30, Duration: 1 Hour 30 Minutes

Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules.

QUESTION 1 [3+2+2.5 MARKS]

Consider the following codes:

```
class Point2D
{
   int x, y;
   public Point2D(int x, int y){
            this.x = x;
            this.y = y;
            System.out.println("Point2D constructor");
   }
   public String Display(){
          //write codes here
   }
}
class Point3D extends Point2D{
   int z;
   //write codes here
}
```

```
public class Test{
   public static void main(String args[]){
            Point2D p2D = new Point2D(1, 2);
            System.out.println(p2D.Display());
            Point3D p3D = new Point3D(5, 4, 3);
            System.out.println(p3D.Display());
   }
}
```

Now:

I. Complete the "Display" method of the Point2D class that prints all the instance variables,

II. Add a constructor in the Point3D class that uses the base class constructor,

III. Add another method "Display" in the Point3D class. You have to use the parent's "Display" method here,

so that the **output** looks like this:

```
Point2D constructor
x= 1, y = 2
Point2D constructor
Point3D constructor
x = 5, y = 4, z = 3
```

1. In String return we can return String like print.
2. Very much careful I need to see the output sample.

QUESTION 2 [3+2+1.5+1 MARKS]

Modify the following program by including/excluding the some codes without changing the highlighted parts.

```
public class Myparent {
   private int p;
   public final int myfunction(){
      return p*p;
   }
   public void set_p(int Q){p = Q;}
  // Write your code here
}
public class Mychild extends Myparent{
   public Mychild(int K){ super(K); }

   public final int myfunction(){
      return p*p+1;
   }
  // write your code of myroot() that finds the square
 //   root of p in class Myparent
  // write other necessary codes here
}
```

```
public class Mytest {
   public static void main(String[] args) {
      Myparent c1, c2;
      c1 = new Mychild(2);
      c2 = new Mychild();
      c2.set_p(2);
      int x = c2. myfunction ();
      double y = ((Mychild) c1).myroot(); // find square
root of p in class Myparent
      System.out.println("x = " + x + ", y = " + y);
   }
}
```

QUESTION 3 [7.5 MARKS]

Write the output of the following program:

```
class Person{
   int id;
   String name;
   static int s = 10;
   {
         System.out.println("3");
   }
   public Person(){
         this.id = 1;
         this.name = "M";
         System.out.println("1");
         s++;
   }
```

```
public Person(int id, String name){
         this();
         this.id = id;
         this.name = name;
         System.out.println("2");
         s++;
   }

 public static void main(String args[]){
         Person p = new Person(1, "N");
         Person p1 = new Person();
         System.out.println(p1.s);
         p.s = 11;
         System.out.println(Person.s);
   }
}
```

QUESTION 4 [7.5 MARKS]

Suppose that you visit a village market where fresh vegetables and fishes are sold. The sellers sell their items with a profit of z% of their production cost *c*. Typical items are given by the following Table:

Food Items	Type *t*	production cost *c* per Kg	profit of *z*%
vegetable	Spinach	20	15
vegetable	Cauliflower	25	18
fish	Carp	300	15
fish	medium	250	20
fish	small	200	25

The class FoodItem that includes type *t*, production cost *c* and profit *z* as public variables and a method findprice() is given as follows:

```
public class FoodItem {
   public double c, z;
   public String t;
   public double getprice(double amount){
      return c*amount*(1+z/100);
   }
}
```

Next, two derived classes Vegetable and Fish are given as follows.

```
public class Vegetable extends FoodItem{
  public void setparameter(){
     if(t == "Spinach"){ c = 20; z = 15; }
     else if(t == "Cauliflower"){c = 25; z = 18;}
  }
  public Vegetable(String t){
     this.t = t;
  }
}
```

```
public class Fish extends FoodItem{
   public void setparameter(){
     if(t == "Carp"){ c = 20; z = 15; }
     else if(t == "medium"){c = 25; z = 20;}
     else if(t == "small"){c = 200; z = 25;}
  }
  public Fish(String t){
     this.t = t;
  }
}
```

Now as a programmer, test the above classes in the main method in a new class *MyTest* by finding your total purchase price if you buy 3Kg fish of type *small* and 2 Kg vegetable of type *Cauliflower*. [Make 2 objects of FoodItems and use the child class references to a FoodItems class object. Then call appropriate methods e.g., **setparameter, getprice**.]