85

(*University of Choice*)

MASINDE MULIRO UNIVERSITY OF

SCIENCE AND TECHNOLOGY

(MMUST)

MAIN CAMPUS

UNIVERSITY EXAMINATIONS

MAIN EXAM

2022/2023 **ACADEMIC YEAR**

THIRD YEAR FIRST SEMESTER **EXAMINATION**

FOR THE DEGREE OF BACHELORS OF SCIENCE IN

(ENGINEERING)

COURSE CODE: CSC 310

COURSE TITLE: OBJECT ORIENTED PROGRAMMING

DATE: 15/12/2022 **TIME: 12:00 – 2: 00 PM**

INSTRUCTIONS TO CANDIDATES:

ANSWER QUESTIONS ONE AND ANY OTHER TWO.

MMUST observes ZERO tolerance to examination cheating

Paper Consists of 11 Printed Pages. Please Turn Over

QUESTION ONE (COMPULSORY) [30 MARKS]

a. Explain situations you will prefer abstract classes and not interfaces. **[2 marks]**

b. Consider the program; comments indicate where missing needed components of the program are to be placed.

```
1   public class MainClass
2   {
3   //definition of a function that prints out a greeting
4   public static void main(String[] args)
5   {
6   // print the greeting
7   //construct a MyClass object called myObject
8   // update myObject
9   //    print myObject
10  }
11  }
12  public class MyClass
13  {
14  // definition of MyClass constructor
15  public static void greetings()
16  {
17  // definition of greetings
18  }
19  public void update(int num, String title)
20  {
21  // definition of update
22  }
23  public void print()
24  {
25  // definition of print
26  }
27  private int numOfItems;
28  private String reportTitle;
29  }
```

i. Suppose you are writing the definition of MyClass on **line 14** above. Which of the following method signatures will be appropriate? **[1 mark]**

A. public MyClass	*D. public void MyClass()*
B. public MyClass()	*E. public MyClass(void)*
C. public void MyClass	

ii. Suppose you wish to call the method that prints the greeting, on **line 6** above. Which of the following statements will call this method correctly? [**Note** that myObject is the object in MyClass.]

[1 mark]

A. MainClass.greetings();	*D. void result = greetings();*
B. myObject.greetings();	*E. greetings();*
C. MyClass.greetings();	

iii. Suppose you wish to construct a MyClassobject called myObject on **line 7**. Which of the following statements will correctly do this? **[1 mark]**

A. *MyClass myObject;*
B. *myObject.MyClass();*
C. *MyClass myObject = MyClass();*
D. *MyClass myObject = new (MyClass);*
E. *MyClass myObject = new MyClass();*

iv. Suppose you wish to call the `update` method on **line 8** above. Which of the following statements will call this method correctly? **[1 mark]**

A. *update(myObject(3, "Hi!"));*
B. *update(3, "Hi!");*
C. *MyClass.myObject.update(3, "Hi!");*
D. *myObject.update(3, "Hi!");*
E. *MyClass.update(3, "Hi!");*

c. A class is defined as

```
1  public class Date
2  {
3   private  int day;  // a valid day ranges between 1-31
4   private  int  month; // a valid month ranges between 1-12
5   private   int year; // assume a valid year  ranges between 2022 -2100
6  }
```

i. Write the definition of valid constructors of this class **[2 marks]**

ii. Write definition of appropriate set methods for day, month and year. **[3 marks]**

iii. Write definition of a toString() method such that it returns a Date object as String of the current Date in the form **day/month/year**. **[2 marks]**

iv. Write a line of code that will pass calls to methods defined in **(ii)** Pass **dummy** arguments to each method. **[2 marks]**

d. Consider the program below.

```
class ArrayDemo{
{
public static void main (String args []){
int twoD[][]= new int [4][];
twoD[0]= new int [1];
twoD[1]= new int [2];
twoD[2]= new int [3];
twoD[3]= new int [4];
int I, j, k=0;
for(i=0; i<4; i++)
for(j=0; j<i+1; j++){
twoD[i][j]=k;
k++;
}
for(i=0; i<4; i++){
for(j=0; j<i+1; j++)
System.out.print(twoD[i][j]+"");
System.out.println();
}
}
}
```

Simulate the out of this program. **[2 marks]**

e. Explain what happens when a variable, a method and a class is declared as final **[3 marks]**

f. A class hierarchy is defined as below:

```
1  public class Point {
2      private int x;
```

```
3       private int y;
4      public Point (int x, int y )
5        {
6         this.x=x;
7         this.y=y;
8        }
9   }
10  public class Line extends Point {
11       private int a = 10;
12       private int b = 20;
13  }
```

i Write a java code to be included in class Line such that it inherits constructor(s) of class Point. **[2 marks]**

ii Assuming class Line has a line of code you provided as answer to (i) above, what will be the effect of the following two line of code. **[2 marks]**

```
1  Line line= new Line (3,5);
2  Line lone = new Line ( );
```

g. Consider the definition of the classes below.

```
final class A {
final void display()  { System.out.println("my method"); }
}
class  B extends A {
void display ()       {System.out.println("I am printing");  }
}
```

Identify errors in the above code extract. **[2 marks]**

h. The volume (V) of a cylinder is given by $\pi r^2 h$ where (π) is pie which is constant, (r) is radius of the cylinder. Write a program that takes the values of r, as input and value of PI from Math class compute and display the volume (V) as an output. Use the JoptionPane for both inputs and output routine.

[4 marks]

QUESTION TWO [20 MARKS

a. Explain the difference between checked and unchecked exceptions in Java, and mention why these two types of exceptions exist. **[4 marks]**

b. You can approximate *e* using the series

$$e= 1+\frac{1}{1!}+\frac{1}{2!}+\frac{1}{3!}+\ldots+\frac{1}{i!}$$

Hint: Because $i! = i \times (i-1) \times (i-2) \cdots \times 2 \times 1$, *therefore,* $\frac{1}{i!} = \frac{1}{i(i-1)!}$ where i is the number of times you approximate e. Using a for loop, write definition of a method declared below such that it returns the approximated value of **e** after **i** approximations.

```
1  public double approximate (int i);
```

[4 marks]

c. Study the class definition bellow and answer the questions that follow.

```
1  public class Sphere{
2        private static int count;
3        private double radius;
4   //initialize the counter to zero here
5      public Sphere ( ) {     count ++;      }
6      public Sphere( double radius) { //implementation  goes here        }
7      public void setRadius(double r ) {   radius=r;   }
8    public  double getRadius( ) { return radius ; }
9    public void dispay ( ) {
10   System.out.println("Radius = " + getRadius( ) ) ;
11  System.out.println(count + ("Sphere Created." ) ;
12   }
13  public static void main (String [ ] args) {
14    Sphere s= new Sphere ( );
15    c.setRadius( 4);
16    c.display( );
17    Sphere s1= new Sphere ( 2);
18    s1=s;
19     s.display( );
20     System.out.println( s );
21     System.out.println( s1 );
22      }
23  }
```

i. Write the line of code omitted at line 4 such that it initializes the variable count to 0.

[1 mark]

ii. Write the definition of the constructor at line 6 such that it increments the value of count as well as initialize radius with the value passed to it. **[2 marks]**

iii. What will be the output of line 16 and 19 when the codes are executed? **[2 marks]**

iv. Will the last two lines (i.e. line 20 and 21) have identical outputs? Explain.

[2 marks]

d. Consider class definition below:

```
public  class A {
   public A( ) {
    System.out.println ("A constructor invoked ");
        }
}
```

```
public class B extends A
 {
   public B ( ) {
      super ( );
    }
public class final C extends B {
    private final int a=21;
public final int getValue ( ) {
    return a;
}
public class D extends C {    }
```

Explain the compilation of the following statements from the main method. **[3 marks]**

```
A a= new B ( );
B b = new D ( );
D d = new C ( );
```

e. Study the code below and then answer the questions that follow.

```
public abstract class X {
private int x ;
public  int y ;
public X(int x,  int y) {
 this.x = x;
 this.y = y;
}
public abstract int test();
}
```

State whether the class will compile without error. If it will not compile correctly, give all the problems preventing correct compilation.

```
class Y extends X {
 public X(int x, int y) {
   super(x,y);
     }
 public int test() {
       return x;
     }
}
```

[2 marks]

QUESTIONTHREE [20 MARKS]

a. Explain the meaning of the following terms and concepts as used in object-oriented programming.

i Data abstraction **[2 marks]**

ii Polymorphism **[2 marks]**

b. Consider an abstract method and polymorphism to perform payroll calculations based on an enhanced employee inheritance hierarchy that meets the following requirements where a company pays its employees on a weekly basis. The employees are of four types:

1. SalariedEmployees who are paid a fixed weekly salary regardless of the number of hours worked,

2. HourlyEmployees who are paid hourly and also receive overtime pay (i.e., 1.5 times their hourly salary rate) for all hours worked in excess of 40 hours,
3. CommissionEmployees are paid based on percentage of their weekly sales and
4. BaseSalariedCommission employees receive a basic weekly salary plus a percentage of their weekly sales. For the current pay period, the company has decided to reward salaried-commission employees by adding 10% to their base salaries.

Each employee, regardless of the way his or her earnings are calculated, has a first name, a last name and a social security number,

i. Draw a UML diagram to model Payroll Scenario **[4 marks]**

ii. Write a definition of the Employee class **[4 marks]**

iii. Why do you consider declaring earning method as abstract? **[2 marks]**

iv. Write definition of Class SalariedEmployee who earns a weekly salary such that it implements the method earning ( ) in Empioyee. SalariedEmployee earns 85% of their monthly salary and pay 15% as TAX. **[4 marks]**

v. Assuming a salaried employee salary is set to 10000.00, what would be the output of the code below if getSalary( ) and earning( ) have identical implementation in SalariedEmployee? Explain your answer. **[2 marks]**

```
Employee  e= new Employee( );
Employee ee= new Employee( );
SalariedEmployee se=new SalariedEmployee( );
e = se;
System.out.println(e.earning( ) );
System.out.println(e.getSalary( ));
System.out.println(ee.earning( ));
```

QUESTION FOUR [20 MARKS]

a. A student was tasked to create a registration form using GUI objects. Explain why he/she may require both classes of swing and AWT packages. **[4 marks]**

b. Write a java code extract that initializes a generic object of a generic class Date with three types: int for day, String for month and int for the year. **[2 marks]**

c. Explain situations you will prefer abstract classes and not interfaces. **[2 marks]**

d. i. Create an interface called Bank that will force all the classes to implement the following functionality. **[4 marks]**

A constant called bankName that would be set to "Central Bank".
A method deposit that would have one parameter of type double.
A method sendMoney that would have one parameter of type double.
A method withdraw that would have one parameter of type double.
A method getBalance that would return a double.

ii. Define a class called MobileApp that implements the Bank interface. MobileApp will implement the basic functionality that all Bank should have. Add the following fields to Mobile App: amount and balance of double type and name (String) to store the bankName.

[5 marks]

iii. Write a driver class that will be used to manifest MobileApp objects capabilities. Use appropriate test data or values. **[3 marks]**

QUESTION FIVE [20 MARKS]

a. Class Date is declared as

```
1  class Date{
2  private static const int months [12 ];
3  private static const int daysPerMonth[months];
4  private static int count;
5  private int day, month, year;
6  public void increment();
7  public bool endMonth(int);
8  public bool leapYear(int);
9  }
```

i. Write lines of code that initialize **count** to **0** and **daysPerMonth** to **{31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}** such that they are accessible to all objects of the class Date.

[2 marks]

ii. Write mutaters (setters) and accessors (getters) methods for the three instance variables: day, month and the year. **[4 marks]**

iii. Write definition of method declared at line 7 such that it returns true if the day passed to it is the last day of the month and returns false otherwise. **[2 marks]**

iv. Write definition of method declared at line 6 such that it adds one day to the current day of the date object. **[2 marks]**

b. You are provide with a window in figure 1 below. Study it and answer the questions that follow.

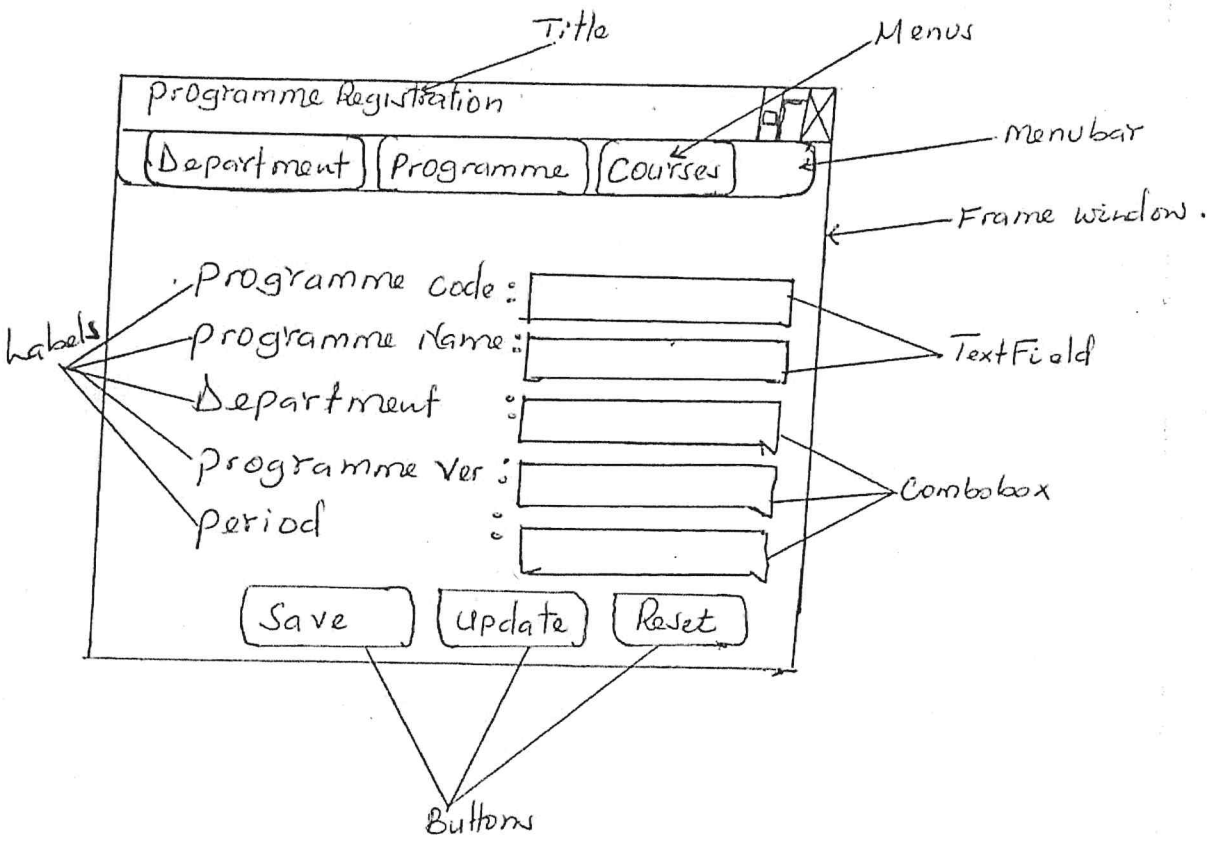

Figure 1: Programme Registration

i Using relevant GUI objects, Data values for comboboxes and layout managers, write a java program that creates the window in figure 1. **[7 marks]**

ii Using relevant Dialog boxes, write java code extract that will:

- Show error message with a String argument "Ensure all the field area entered!!" when save Button is clicked on.
- Show a dialog box for confirmation with a String argument ("Are you sure you want to Update?") if update is clicked on.
- Show an information message "Reset is done successfully. ", when Reset button is clicked on.