**UNITY UNIVERSITY**

**DEPARTMENT OF COMPUTER SCIENCE**

1. ***WRITE YOUR:-***

***NAME\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***IDNO\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***DEPARTMENT\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***TIME ALLOWED:-2:00 hour***

1. ***INSTRUCTION:***
2. **DO NOT TURN THE PAGE UNTIL YOU ARE TOLD TO DO SO**
3. **SWITCH OFF YOUR CELL PHONES**
4. **MAKE SURE THAT THE EXAM BOOKLET CONTAINS 4 PARTS**
5. **WRITE YOUR ANSWERS ON THE SPACES PROVIDED AFTER EACH QUESTION.**
6. ***FOR EVALUATION PURPOSE ONLY***

|  |  |  |  |
| --- | --- | --- | --- |
| ***CONTINIOUS ASSESSMENT*** | ***FINAL EXAM*** | ***TOTAL*** | ***GRADE*** |
|  |  |  |  |

***Part One: Read the questions 1 to 7 carefully and give short and precise answer for each questions in the boxes provided under each questions (20 points).***

1. When an overridden method is called from within the sub-class? Explain the two events observed both in the super class and subclass of overridden method (2 point)
2. Write the difference between Overloading and Overriding methods (2 point)

|  |  |
| --- | --- |
| **Overloading Method** | **Overriding Method** |
|  |  |

1. Write the difference between classes and interfaces in java programming language(3 point)

|  |  |
| --- | --- |
| **Classes** | **Interface** |
|  |  |

1. Write down the syntax or general form to define a subclass, abstract class and to define an interface (3 point).
2. **Syntax to define Subclass**
3. **Syntax to define abstract class**
4. **Syntax to define interface**
5. List down the advantages of using ***super*** keyword in a program and write the syntax for the two forms of the super keyword (2 points).
6. List and explain the three ***access modifiers or specifiers*** used by java programming language to access members of the class by giving examples (3 point).
7. Explain the following java programming features by giving examples (5 points).

|  |  |  |
| --- | --- | --- |
| **Constructor** | **Upcasting** | **abstract class** |
|  |  |  |

***Part Two:- Read the java code that is described in each of the following questions (8- 10) carefully and identify the syntax errors you observed and write your answers inside the box provided under each questions (15 points)***

1. Observe the following simple java program and identify the ***syntax errors and explain the reason that makes error*** in this program and ***write only the correct code to make the program error free***(4 points)

**Syntax Errors**

final class Father {

final void getFather() {

System.out.println("This is a final method.");

}

}

class Child extends Father {

void getFather() {

System.out.println("This is from sub class B");

}

}

1. Identify the syntax errors you observed for the following java program and write your answer inside the box provided below (6 points).

package inheritpoly;

import javax.swing.JOptionPane;

**Syntax Errors**

class Employee{

double salary;

double allwance;

double tax(){

return salary\*.35;

}

double deduction(){

return tax()+ pension();

}

double growthSalary(){

return (salary+allwance);

}

double netPay(){

return (growthSalary()-deduction());

}

}

class TestSalary{

public static void main(String args[]){

//==============================================================

JOptionPane.showMessageDialog(null, "WELCOME TO EMPLOYEES SALARY CALCULATION"+ " ", "", JOptionPane.PLAIN\_MESSAGE);

//=============================================================

emp=JOptionPane.showInputDialog("Enter Employee ID");

empsal.empid= Integer.parseInt(emp);

//==========================================================

emp=JOptionPane.showInputDialog("Enter Name of Employee");

empsal.name= String.format(emp);

//==============================================================

emp=JOptionPane.showInputDialog("Enter Salary of the Employee");

empsal.salary=Integer.parseInt(emp);

//==============================================================

emp=JOptionPane.showInputDialog("Enter the allowance of Employee");

empsal.allwance=Integer.parseInt(emp);

//==============================================================

JOptionPane.showMessageDialog(null, "The ID of an Employee is"+ " " +empsal.empid,"",JOptionPane.PLAIN\_MESSAGE);

JOptionPane.showMessageDialog(null, "The Name of an Employee is"+ " " +empsal.name,"",JOptionPane.PLAIN\_MESSAGE);

JOptionPane.showMessageDialog(null, "The Salary of an Employee is"+ " " +empsal.salary,"",JOptionPane.PLAIN\_MESSAGE);

JOptionPane.showMessageDialog(null, "The Allowance of an Employee is"+ " " +empsal.allwance,"", JOptionPane.PLAIN\_MESSAGE);

//==============================================================

JOptionPane.showMessageDialog(null, "The Growth Salay of an Employee is"+ " "+empsal.growthSalary(),"",JOptionPane.PLAIN\_MESSAGE);

JOptionPane.showMessageDialog(null, "The Tax deducted from an Employee is"+ " " +empsal.tax(),"", JOptionPane.PLAIN\_MESSAGE);

JOptionPane.showMessageDialog(null, "The Pension of an Employee is"+ " " +empsal.pension(),"", JOptionPane.PLAIN\_MESSAGE);

JOptionPane.showMessageDialog(null, "The Total deduction of an Employee is"+ " " +empsal.deduction(),"", JOptionPane.PLAIN\_MESSAGE);

JOptionPane.showMessageDialog(null, "The Net Payment Of Employee is"+ " " +empsal.netPay(),"", JOptionPane.PLAIN\_MESSAGE);

JOptionPane.showMessageDialog(null, "THANKS A LOT FOR ALL"+ "", "", JOptionPane.PLAIN\_MESSAGE);

}

}

1. Determine the syntax errors you observed for the following java program and write your answer inside the box provided below (5 points)

package inheritpoly;

class Case{

protected int i;

public Case(int i){

this.i = i;

}

public void Increment(Case parent){

System.out.println("Incremented I of superclass = "+parent.i);

**Syntax Errors**

}

}

class Reason extends Case{

public Reason(int i){

super(i);

}

public void Decrement(){

child.i--;

System.out.println("Decremented I of subclass = "+child.i);

}

}

class TestTwo{

public static void main (String args[]){

Reason b1, b2;

b1 = new Reason(11);

b2 = new Reason(22);

a1 = new Case(110);

a1.Increment(a1);

a1.Increment(b1);

b1.Decrement(b2);

b1.Decrement(b1);

}

}

***Part Three:-Read the java code that is described in each of the following questions (11-13) carefully and determine the output of the program you expect and write your answers inside the box provided under each questions (15 points)***

1. Determine the output of the following java program and identify that what object oriented programming features are implemented by this program (6 points).

package inheritpoly;

class Figure {

double dim1;

double dim2;

Figure(double a, double b) {

dim1 = a;

dim2 = b;

}

double area() {

**Output**

System.out.println("Area for Figure is undefined.");

return 0;

}

}

class Rectangle extends Figure {

Rectangle(double a, double b) {

super(a, b);

}

// override area for rectangle

double area() {

System.out.println("Inside Area for Rectangle.");

return dim1 \* dim2;

}

}

class Trig extends Figure {

Trig(double a, double b) {

super(a, b);

}

// override area for right triangle

double area() {

System.out.println("Inside Area for Triangle.");

return dim1 \* dim2 / 2;

}

}

class RunTimePolymorphism {

public static void main(String args[]) {

Figure f = new Figure(10, 10);

Rectangle r = new Rectangle(9, 5);

Trig t = new Trig(10, 8);

Figure figref;

figref = r;

System.out.println("Area is " + figref.area());

figref = t;

System.out.println("Area is " + figref.area());

figref = f;

System.out.println("Area is " + figref.area());

}

}

1. Determine the ***output of the following java program*** (4 points).

package inheritpoly;

class A {

int i, j;

void showij() {

System.out.println("i and j: " + i + " " + j);

**Output**

}

}

// Create a subclass by extending class A.

class B extends A {

int k;

void showk() {

System.out.println("k: " + k);

}

void sum() {

System.out.println("i+j+k: " + (i+j+k));

}

}

class SimpleInheritance {

public static void main(String args[]) {

A superOb = new A();

B subOb = new B();

// The superclass may be used by itself.

superOb.i = 10;

superOb.j = 20;

//System.out.println("Contents of superOb: ");

superOb.showij();

// The subclass has access to all public members of its superclass

subOb.i = 7;

subOb.j = 8;

subOb.k = 9;

//System.out.println("Contents of subOb: ");

subOb.showij();

subOb.showk();

System.out.println("Sum of i, j and k in subOb:");

subOb.sum();

}

}

1. Observe the following Java code and determine the ***output of this program and indicates the features of java implemented by this program***(5 points)

class Hide{

protected float x;

**Output**

protected int y;

public Hide (float x, int y){

this.x = x \* x;

this.y = y;

}

}

class Member extends Hide {

private int x;

private float z;

public Member (int x, int y, float z){

super(x,y);

this.x =x;

this.z = z;

}

public void M1(){

z = (super.x + y)/x;

System.out.println("Superclasses version of x: " +super.x);

System.out.println("Subclasses version of x: " +x);

System.out.println(" Z = " +z);

}

}

classHideSuperClassField{

public static void main(String args[]){

Member o = new Member (8, 4, 10);

o.M1();

System.out.println("Y = " +o.y);

}

}

***Part Four:-Write java program based on the information given to you for the following five questions (14-18). Make your writing neat and provides comment if it is possible to make the program more readable. Use the attached blank paper for your answers (30 points)***

1. Write simple java program to implement both method overloading and overriding based on the following information. Use Class A as a super class and class B and C as a subclass, where B inherit properties and methods of class A and C inherits both the properties and methods of class B and A. You can use any kind instance variables and methods which does not violate the java convention about method and variable declaration to implement this activity (7 points).
2. Define ***abstract class by the name Service and abstract method charge()*** and define a ***subclasses*** by the name ***Vat, ServiceCharge, TotalPrice, TotalDeduction and TotalCost*** that all these subclasses inherits the super class Service and the abstract method charge() is left to all these subclass . Define the necessary attributes constructors, methods, declare reference variables of a super class and assign the reference variable of a super class to all subclasses to implement your work. Consider the following information for your work(8 points)

* TotalPrice=(UnitPrice \* Quantity)
* Vat is 15% of TotalPrice
* ServiceCharge is 5% of TotalPrice
* TotalDeduction=(Vat + ServiceCharge)
* TotalCost=(Vat + ServiceCharge + TotalDeduction)

1. Write simple java program to implement both method overloading and overriding based on the following information. Use Class Example1 as a super class and class Example2 and Example3 as a subclass, where Example2 inherit properties and methods of class Example1 and Example3 inherits both the properties and methods of class Example2 and Example1. You can use any kind instance variables and methods which does not violate the java(5 points)
2. Define ***abstract class by the name Bank and abstract method calculate ()*** to perform different kinds of bank operations such as register customer, deposit, withdrawals and net balance and declare all the necessary instance variables extended to subclasses. Define a subclass by the name ***Register, Deposit, Withdrwals and NetBalance*** by extending the super class Bank. The ***implementation of abstract method calculate ()*** is left to all these subclasses. Use the following information for your work( 5 points)

* Interest Rate=(Initial Deposit)\* time\* rate of interest-Use 6 month for time and 6% for rate
* Withdrwals=(Initial Deposit-Amount withdraw)
* NetBalance=(InitialDeposit+InterestRate)

1. Write Java Program to calculate Sum and Average of two numbers by defining Sum as a super class and Average as a Sub class. Define sum() and average() methods for super and subclass respectively and call both methods through the subclass objects. Suppose the program is expected to accept input from the user(5 points)