Data Structure and Algorithm (CSE-2001)

Assignment-II

Object Oriented Design

1. Define a class **Person** with two instance variables:

   - *name and age*

   and two member methods:

   - *setData()*: set the details of the person.
   - *displayData()*: display the details of the person.

   Now, create two objects of class *person* and initialize one object value directly (by using the **dot(.)** operator *name: "Rohan" and age: 20*). Accept your name and age through the keyboard and set them to another object using the *setData()* method. Now display both the member variables using the *displayData()* method. Also, check who is younger.

2. Define a class called **Complex** with instance variables *real, imag* and instance methods

   - *void setData(),*
   - *void display(),*
   - *public Complex add(Complex, Complex)*

   Write the main method in a separate class to implement the addition of two complex number with the given method signature as above.

3. In a super market each product is having minimum details like prodId, price, quantity that is used during the biling process. Keeping this in mind prepare a class named as **Product** having the member variables

   - *prodId*, *price*, *quantity*
   - a static variable *totalPrice*

   Initialize the value of product through parameterized constructor. It consists of a *display()* method to display the value of instance variables. A person went to market and purchase 5 different products. Using the above mentioned class, display the details of products that the person has purchased. Also, determine how much total amount the person will pay for the purchase of 5 products.

4. Define a class **Deposit**. The instance variable of the class Deposit are mentioned below.

Instance Variable	Data Type
Principal	Long
Time	Integer
Rate	Double
TotalAmt	Double

Initialize the instance variables Principal, Time, rate through constructors. Constructors are overloaded with the following prototypes.
Constructor1: *Deposit ( )*
Constructor2: *Deposit (long, int, double)*
Constructor3: *Deposit (long, int)*
Constructor4: *Deposit (long, double)*
Apart from constructor, the other instance methods are
(i) display ( ): to display the value of instance variables,
(ii) calcAmt( ): to calculate the total amount.
totalAamt = Principal + (Principal×rate×Time)/100;

5. Define a base class **Person** with instance variable *name, age.* The instance variables are initialized through constructors. The prototype of constructor is as below.

   Person (string, int)

   Define a derived class **Employee** with instance variables *Eid, salary.* The instance variables are initialized through constructors. The prototype of constructor is as below.

   Employee (string, int, int, double).

   Another instance method of Employee class is *empDisplay()* to display the information of employee details.

6. Create an abstract class **Marks** with three instance variables (*markICP*, *markDSA*, and *percentage*) and an abstract method *getPercentage()*. Create two classes: **CSE** with instance variable *algoDesign*, and **NonCSE** with instance variable *enggMechanics*. Both classes inherit the abstract class **Marks** and override the abstract method *getPercentage()*. The constructor of class **CSE** takes the marks in three subjects (*markICP*, *markDSA*, and *algoDesign*) as its parameters, and the constructor of class **NonCSE** takes the marks in three subjects (*markICP*, *markDSA*, and *enggMechanics*) as its parameters. Create an object for each of the two classes and print the percentage of marks for both students.

7. Define an interface **DetailInfo** to declare methods *display( )* & *count( ).* Another class **Person** contains a static data member *maxcount*, instance member *name* & method *display( )* to display name of a *person, count* the no. of characters present in the name of the person.

8. Design a package that contains two classes **Student** & **Test**. The Student class has data members as *name, roll* and instance methods *inputDetails()* & *showDetails()*. Similarly the Test class has data members as *mark1, mark2* and instance methods *inputDetails()*, *showDetails()*, Student is extended by Test. Another package carry interface Sports with 2 attributes *score1, score2.* Find grand total mark & score in another class.

Home Assignment

1. A sales person is paid commission based on the sales he makes as shown by the following table:

SALES	**COMMISSION**
Under Rs. 100	2% of SALES
Rs 500 and under Rs 5000	5% of Sales
Rs 5000 and above	8% of sales

   Write a class, Commission, which has:
   An instance variable, sales; an appropriate constructor; and a method, getCommission() that returns the commission.
   Now write a **Demo** class in Java to test the **Commission** class by reading a sale from the user, using it to create a Commission object after validating that the value is not negative. Finally, call the *getcommission()* method to get and print the commission.
   If the sales are negative, your **Demo** class should print the message "*Invalid Input*".

2. Define a class called **Book** with instance variables *BName, BEdition, BPrice*. Use constructor to initialize the instance variables of the class. Another instance method *display ( )* to display the book information. A person purchase 5 different books. Display the book details which has the maximum price.

3. Create a class **Bank** with two instance variables: *bankNname*, *depositAamount* and a class variable *totalAamount*. Defined three member methods: *setBankName()* to set the bank name, *setAmount()* to set the deposit amount, and *showData()* for displaying the bank name and deposit amount. A person visited 5 different banks and deposite money. Create 5 array object of class **Bank** set the bank name and deposit amount (the minimum deposit amount is 1000) using member methods *setBankName() and setAmount()*. Call the *showData()* method to display the information with *totalAmount* deposited by the person. Also, design a method that displays the bank name where the person deposits the minimum amount.

4. Write a Java code that would represent **Distance** object (meters, centimetres) using classes. The class definition include a parameterized

constructor, *void display ( )* to display the instance variables, *void sum (Distance, Distance)* to add two distances. Now write a main function that creates a couple of Distance objects and demonstrates the addition of two distances.

5. A point in the x-y plane is represented by its x-coordinate and y-coordinate. Design a class, PointType in Java, that can store and process a point in the x-y plane. You should then perform operations on the point, such as showing the point, setting the coordinates of the point, printing the coordinates of the point, returning the x-coordinate, and returning the y- coordinate. Every circle has a centre and a radius. Given the radius, we can determine the circle's area and circumference. Given the centre, we can determine its position in the x-y plane. The centre of a circle is a point in the x-y plane. Design a class, CircleType that can store the radius and centre of the circle. Because the center is a point in the x-y plane and you designed the class to capture the properties of a point from PointType class. You must derive the class CircleType from the class PointType. You should be able to perform the usual operations on a circle, such as setting the radius, printing the radius, calculating and printing the area and circumference, and carrying out the usual operations on the center.