

The National Institute of Engineering, Mysuru

**OBJECT ORIENTED PROGRAMMING**

**with C++ LAB (CS4L02)**

**IV Semester CSE (2021-22)**

***List of Programs***

|  |  |  |
| --- | --- | --- |
| ***Q. No*** | ***CO*** | ***Question*** |
| 1. | CO1 | A phone number, such as (044) 234-8900, can be thought of as having three parts: the area code (044), the exchange (234) and the number (8900). Write a program that uses a class to store these three parts of a phone number separately. Call the class phone. Create two class objects of type phone. Initialize one, and have the  user input a number for the other one. Display both the numbers. |
| 2. | CO1 | Create two classes DM and DB which store the value of distances. DM stores distance in meters and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out the addition operation. The object that stores the results may be a DM object or DB object, depending on the units in which the results are  required. The display should be in the format of feet and inches or meters and centimeters depending on the object on display. |
| 3. | CO1 | Write a C++ program to create a class Rectangle with data members: length, breadth, area and member functions as:   1. Rectangle& setDimensions(const tint&, const int&) – sets the length and breadth of arguments using constant references. Use *this* pointer to return the resized Rectangle by reference. 2. int computeArea() – to compute and returns area of a rectangle. 3. int computePerimeter() – to compute and return perimeter of a rectangle. 4. Two constructors, default constructor to initialize data members to zero and an overloaded constructor as arguments with breadth having a default value.   Define all the member functions outside the class. Create objects of Rectangle type and test all the functions. Delete the objects before program terminates. |
| 4. | CO2 | Write a C++ program to create a class called Complex and implement the following overloading member functions that return a Complex number after performing addition of input args.   1. Complex ADD(int a, Complex s) – where a is an integer(real part) and s is a complex number. 2. Complex ADD(Complex &s1, Complex &s2) |
| 5. | CO2 | Write a C++ program to create a class called STACK using an array of integers. Implement the following operations by overloading the operators ‘+’ and ‘-’.   1. S1 = S1 + element; where S1 is the object of class STACK and element is an integer to be pushed on the top of stack. 2. int element = S1--; where S1 is the object of class STACK. ‘—‘operator pops the top element.   Handle the STACK empty and full conditions and also display the contents after every operation by overloading << operator. |
| 6. | CO3 | Write a C++ program to read and print Employee information (name, empID, gender) with Department (deptName, workAssigned) and with Loan information (loanDetails,  loanAmt) using hierarchical inheritance. |
| 7. | CO3 | Write a C++ program to design a Student class representing USN and a Test class representing the scores of the student in various subjects and a Sports class  representing the score in sports. The Sports and Test classes is inherited by Result |

|  |  |  |
| --- | --- | --- |
|  |  | class having the functionality to add the scores and display the final result of a student. |
| 8. | CO3 | Write a C++ program to create a class called STUDENT with data members USN, Name and Age. Using inheritance, create the classes UGSTUDENT and PGSTUDENT having fields as Semester, Fees and Stipend. Enter the data for at least 5  students from UG and PG. Find the average age for all UG and PG students separately. |
| 9. | CO2, CO3 | Implement class *Shape* with the following specification*:*  *class Shape{*  *protected:*  *float area, perimeter; public:*  *Shape();*  *virtual void initialize()=0; virtual float computeArea()=0;*  *virtual float computePerimeter()=0; virtual ~Shape();*  *};*  Implement 2 classes *Triangle* and *Rectangle* publicly derived from class *Shape*, with suitable data members. Implement all the functions derived from class *shape* in each of the derived classes. Write a C++program to create objects of each of the derived class and assign to the base class (Shape) type pointer/reference. Demonstrate runtime  polymorphism by calling the functions of the derived class objects by using the base class pointer/reference. |
| 10. | CO4 | Write two function templates in C++ to   * 1. Sort the numbers   2. To search a given number   Demonstrate the above functions on array of integers and double. |
| 11. | CO4 | a) Write a simple calculator using class template in C++ |
| b) Write a program implementing stack and its operations using template class. |
| 12. | CO5 | 1. Perform these basic vector operation using Standard Template Library:    1. Find the number of elements in the vector.    2. Check whether the vector is empty or not.    3. Insert some elements into the vector.    4. Remove the element at a particular position.    5. Find the index of a particular element in a vector. |
| b) Make a vector of random numbers and sort it in descending order using STL and  also find its sum. |
| 13. | CO5 | Write a C++ program to illustrate working of RTTI. |