MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY

Department Of Computer Science And Engineering

LIST OF PROGRAMS -OBJECT ORIENTED PROGRAMMING

1. Write a program for multiplication of two matrices using OOP.

2. Write a program to perform addition of two complex numbers using constructor overloading. The first constructor which takes no argument is used to create objects which are not initialized, second which takes one argument is used to initialize real and imag parts to equal values and third which takes two argument is used to initialized real and imag to two different values.

3. Write a program to find the greatest of two given numbers in two different classes using friend function.

4. Create a class Student, which have data members as name, branch, roll no., age, sex, five subjects. Display the name of student & his percentage who has more than 70%. Use array of objects. [Array of objects]

5. Implement a class string containing the following functions:

a. Overload + operator to carry out the concatenation of strings.

b. Overload = operator to carry out string copy.

c. Overload <= operator to carry out the comparison of strings.

d. Function to display the length of a string.

e. Function tolower( ) to convert upper case letters to lower case.

f. Function toupper( ) to convert lower case letters to upper case.

6. Write a C++ program to define function power to raise a number m to a power n the function takes a double value for m. And integer value for n and return the result correctly use a default value of 2 for n to make the function calculate squares when this argument is omitted.

7.Write a program to overload new and delete operator.

8.Write a program to overload unary increment (++) operator.

9.Write a program to generate a Fibonacci series using Copy Constructor.

10.Consider a class point containing x and y coordinates. Write necessary functions for the following cases:
1. to accept a point
2. to display it
3. to find distance between two points using operator overloading (-)(Use friend function).

11.Create a class which keep track of number of its instances. Use static data member, constructors and destructors to maintain updated information about active objects.

12.Create a base class basic info with data members name, roll no, sex and two member functions getdata and display. Derive a class physical fit from basic_info which has data members height and weight and member functions getdata display. Display all the information using object of derived class.(Single Inheritance)

13.Create class first with data members book no, book name and member function getdata and putdata. Create a class second with data members author name, publisher and members getdata and showdata. Derive a class third from first and second with data member no of pages and year of publication. Display all this information using array of objects of third class. (Multiple Inheritance)

14. Design three classes STUDENT, EXAM, and RESULT. The STUDENT class has data members such as roll no, name. Create a class EXAM by inheriting the STUDENT class. The EXAM class adds data members representing the marks scored in six subjects. Derive the RESULT from the EXAM class and has its own data members such as total marks. WAP to model this relationship.(Multilevel Inheritance)

15.Create a base class called SHAPE. Use this class to store two double type values. Derive two specific classes called TRIANGLE and RECTANGLE from the base class. Add to the base class, a member function getdata to initialize base class data members and another member function display to compute and display the area of figures. Make display a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes design a program that will accept driven of a TRIANGLE or RECTANGLE interactively and display the area.

16.Write a program to define the function template for swapping two items of various datatypes such as integers, float and characters.

17.Using the concept of function overloading, write function for calculating area of triangle, circle and rectangle.

18.Create a class called LIST with two pure virtual function store() and retrieve().To store a value call store and to retrieve call retrieve function. Derive two classes stack and queue from it and override store and retrieve.

19.Create a class TIME with members hours, minutes and seconds. Take input, add two-time objects and passing objects to function and display the result.

20. Write a program to define the function template for calculating the square of given numbers with different data types.

21.Write a program to illustrate how Template function can be overloaded.

22.Write a program to illustrate how to define and declare a class template for reading two data items from the keyboard and to find their sum.

23. Write a program to find the sum of two numbers declared in a class and display the numbers and sum using friend class.

24.Write a program to demonstrate the use of friend function with Inline assignment.

25. Write a program to demonstrate the use of special functions, constructor and destructor in the class template. The program is used to find the bigger of two entered numbers.

26.Write a program to perform the deletion of white spaces such as horizontal tab, vertical tab, space ,line feed ,new line and carriage return from a text file and store the contents of the file without the white spaces on another file.

27.Write a program to read the class object of student info such as name , age ,sex ,height and weight from the keyboard and to store them on a specified file using read() and write() functions. Again the same file is opened for reading and displaying the contents of the file on the screen.

28.Write a program to raise an exception if any attempt is made to refer to an element whose index is beyond the array size.
29. Write a program to read a set of lines from the keyboard and to store it on a specified file.
30.Write a program to read a text file and display its contents on the screen.
31.Write a program to copy the contents of a file into another.