Assignment-1:

Program 1: Write a program to find the cube of given integer using inline function

Program 2: Design a class to represent a bank account with data members name, account-number, account-type, and balance. Define functions to assign initial values, to deposit an amount, to withdraw an amount after checking balance, and to display the name and balance.

Program 3: Using a class write a program that receives inputs, principle amount, time and rate. Keeping rate 8% as the default argument, calculate the simple interest for three customers.

Program 4: Write a program according to the specification given below:

- Create a class Teacher with data members, tid & subject and member functions for reading and displaying data members.
- Create another class Staff with data members, sid & position, and member functions for reading and displaying for data members.
- Derive a class Coordinator from the Teacher and Staff and the class must have its own data member department and member functions for reading and displaying data members.
- Create two objects of the Coordinator class and read and display their details.

Program 5: Create a class called Mountain with data members name, height, location, a constructor initializes the members to the values passed to it as parameters, a function called CmpHeight( ) to compare two objects and DisplayInf( ) to display the information of Mountain. In main, create two objects of the class mountain and print the information of the mountain which is of greatest height.

Program 6: Write a program to input two vectors coordinates from a base class named 'Base'. Class 'Derived' inherits all the properties of class Base. Class 'Derived' must contain a function named add_vector( ) that add the two vectors input from the base class and finally display the result from a function display() that is friend with the base class.

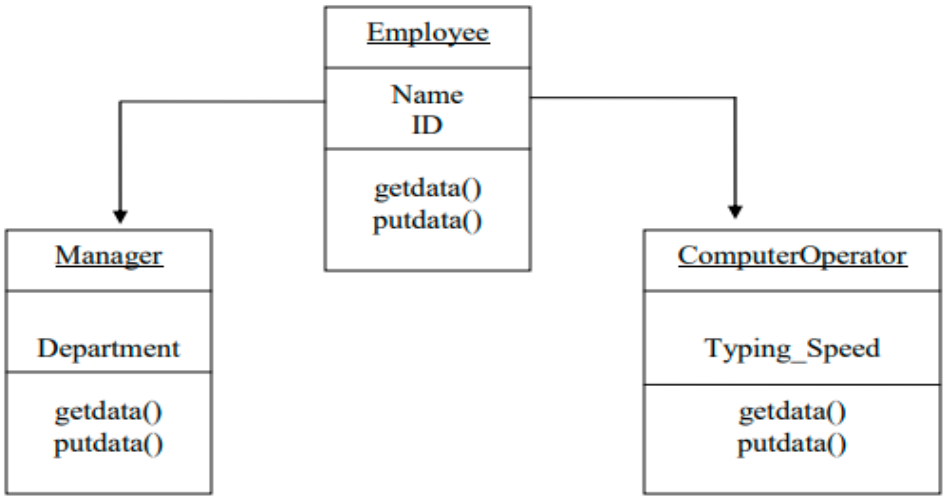
Program 7: Create a class called Employee with data member Code, Name, Address and Salary. Create a constructor to initialize the member of the class. Also create another constructor so that we can create an object from another object. Define member function display( ) to display the information of the class.

Program 8: Write a program to create a class called Room in which data items are length, breadth and height. Create an array of objects in the room. And finds its area and volume.

Program 9: Create a class Account with data members acc no, balance, and min_balance(static).

- Include methods for reading and displaying values of objects
- Define static member function to display min_balance
- Create array of objects to store data of 5 accounts and read and display values of each object.

Program 10:



Write a C++ program to represent the above inheritance scheme. Also write a main() function to test the classes, Manager and Computer Operator, by creating their objects, taking input and displaying the corresponding values.

Program 11: Create a class time having hour, minute and second as data members. The constructor is used to initialize these members. Define a function add that takes two class objects as arguments and add them respectively then display the aggregate result.

Program 12: Write base class that asks the user to enter Time (hour, minute and second) and derived class adds the Time of its own with the base. Finally, make third class that is friend of derived and calculate the difference of base class time and its own time.

Program 13: Write a program to add two complex numbers using the concept of constructor.

Program 14: Write a program to enter the information of n students and then display it using the concept of multilevel inheritance.

Program 15: Create classes called class1 and class2 with each having one private member. Add member function to set a value (say setvalue) on each class. Add one more function max( ) that is friendly to both classes. max( ) function shound compare two private member of two classes and show maximum among them. Create one-one object of each class then set a value on them. Display the maximum number among them.

Program 16: Write a program to swap variables of two classes using friend function.

Program 17: Write a program to add two complex numbers using friend function.

Program 18: Write a program to add time in hours and minutes of two different objects using friend function

Program 19: Write a program to find the area of a triangle (when its sides are given) using the concept of overloaded constructor.

Program 20: Write a program to overload multiplication operator(*) showing the multiplication of two objects.

Program 21: Write a program to add two complex numbers. Your program should have three objects. Each object contains two attributes (ie. real and imaginary part). Now add each attribute and save them into third object separately. Use the concept of '+' operator overloading.

Program 22: Write a program to find the sum and difference of any two complex numbers by overloading '+' and '-' operator.

Program 23: Design a Soccer player class that includes three integer fields: a player's jersey number, number of goals, number of assists and necessary constructors to initialize the data members. Overload the > Operator (greater than). One player is considered greater than another if the sum of goals plus assists is greater than of the others. Create an array of 11 soccer players, then use the overloaded > operator to find the player who has the greatest total of goals plus assists.

Program 24: Write a program to overload '+' operator to concatenate two strings.

Program 25: Write a program to add two Times expressed in hours, minutes and seconds using operator overloading.

(Note: Demonstrate the output at the end of each program)