ASSIGNMENT-I

Note : (1) Programs should have proper validations for the inputs
(2) Use proper variable names
(3) Program should contain program definition as comments and should have proper Indentation

Classes , Objects , Functions , Constructors and Destructors

(1) Demonstrate the use of static variables in a class by using it to count the number of objects created in the program.

(2) Define a class to represent a bank account. Include the following members :

DATA MEMBERS	MEMBER FUNCTIONS
------------------------	------------------------------
Name of depositor	(1) To assign initial values
Account Number	(2) To Deposit the amount
Type of Account	(3) To withdraw an amount after checking the Balance amount in account
	(4) To display name and balance

Write C++ program to handle 10 customers.

(3) A book shop maintains the inventory of books that are being sold at the shop.
The list includes details such as author, title, price, publisher and stock position. Whenever a customer wants a book, the sales person inputs the title and author and the system searches the list and displays whether it is available or not. If it is not, an appropriate message is displayed. If it is, then the system displays the book details and requests for the number of copies required. If the requested copies are available, the total cost of the requested copies is displayed; otherwise the message "Required copies not in stock" is display Design a system using a class called books with suitable member functions and constructors. Include the following features also :

(a) The number of successful and unsuccessful transaction should be recorded for the purpose of the statistical analysis. Use static data members to keep count of transactions.

(b) The price of the books should be updated as and when required. Use a private member function to implement this.

(4) Write a class to represent a vector. Include member functions to perform the following tasks:

1. To create the vector
2. To modify the value of a given element
3. To multiply by a scalar value
4. To display the vector in the form(10,20,30...)

(5) Define a supplier class. The class contains details about the suppliers. One of the details is the list of items supplied by the supplier. Create a class Item to store item details. The items supplied by any given supplier are different and varying. Use dynamic memory allocation in the constructor function to achieve the solution.

(6) Define a class Student. Add data members as Rollno, Name, Marks_obtained, Max_marks and Percentage. Write member functions for reading values, calculating percentage and printing values of student. Define one more class as MCA_II. MCA_II contains array of students. MCA_II class should contain member functions as Add, delete, modify and replace. MCA_II is to be defined as friend of Student class.

(7) Define a class Car. Add data members as Make, Color, Size, and Cost. Write member functions for reading values and printing values of car. Define one more class as CarCollection. CarCollection contains array of cars. CarCollection class should contain member functions as Add, delete, modify and replace. CarCollection is to be defined as friend of Car class. Write C++ programs to test your classes.

(8) Use Employee and EmpCollection classes. Employee class contains details about employee and EmpCollection contains collection of employees in form of an array. Provide GetSubordinates friend function which returns an object of EmpCollection class which contains details of subordinates of a manager. The employee object describing manager is to be passed as a parameter.

(9) For a supermarket, define a bill class. All the bill objects will contain bill number, name of clerk preparing the bill, each item with quantity and price, total amount to be paid. Total items in the bill are varying. Define dynamic memory allocation constructor for bill class such that any number of items from 1 to 50 can be accommodated in a single bill. There is an array describing each item with price. Price is to be picked up from that array. Now overload = operator and provide reasons for need of such operator. Use Exception Handling techniques to handle errors.

Exercises from Text Book : All practical exercises of chapter 3, 4 & 5

------------------------------------Assignment-2--------------------------------------

Operator Overloading

(1) Overload all the four arithmetic operators to operate on a vector class and also the overload the * operator to multiply scalar values to the vector class. Overload the >> operator to input a vector and the << operator to display the vector in the form (10,20,.....). Also overload the [] operator to access the individual member of the vector. Use Dynamic memory allocation to achieve the solution. Write appropriate constructor and destructure for the class.

(2) Write a menu driven program that can perform the following functions on strings.
(Use overloaded operators where possible). (Do not use predefined string class)

1. Compare two strings for equality (== operator)
2. Check whether first string is smaller than the second (<= operator)
3. Copy the string to another
4. Extract a character from the string (Overload [])
5. Reverse the string
6. Concatenate two strings (+ operator)

(3) Create two classes Salary and Deductions to store the value of total salary and total deductions. Salary has date members as MED for medical allowance, HRA for house rent, BAS for basic salary , DA for dearness allowance. Deductions had data member IT for income tax, and PF for provident fund. Write appropriate constructors to set values into the data members, overload >> operator to display the salary as well as deductions and the - operator to subtract object of deductions from salary and display the net salary. Use friend function to carry out the subtraction.

(4) Define a class coord, which has x and y coordinates as its data members . Overload the -- operator for the coord class. Create both its prefix and postfix forms. Overload the + operator for the coord class so that it is both a binary operator and unary operator. When used as unary operator, have the + make any negative coordinate value positive.

(5) Create a date class with the following capabilities :

(a) Output the date in multiple formats such as

DD MM YYYY
MM/DD/YY
May 14, 2001

(b) Use overloaded constructors to create Date objects initialized with date format in section (a)

(c) Overload operators for testing equality of two dates and for comparing dates to determine if one date is prior to, or after, another date.

(d) Create a member function nextday to increment the day by day. The date object function should always remain in consistant state. Be sure to test following cases : (i) Incrementing to next month (ii) Incrementing into the next year.

(6) Create a class say coordinate class with variables x and y. Overload the == and != operators to perform required operations on the object of that class. Overload the * so that it performs multiplication operation with object of coordinate class either in the form int * obj or obj * int. Also define the -- operator to decrement values when applied to an object. Also write appropriate conversion function to convert coordinate object into int.

(7) Design a class date which sets date of object to dd, mm ,yyyy format .Overload + and _ operators for the class date to add given no of days to a particular date object as d1 = d1 + 10 and d1 = 10 + d1 where

d1 is set using a constructor to some valid date and to find difference between two given date. Display the original dates , after addition and the difference. The date objects created should be validated in the constructor itself.

(8) Define a singly linked list class, which is a made up objects of node class. Provide addition, deletion of nodes, with operator overloading.

(9) Define a matrix class, which allows addition, subtraction, multiplication with another matrix, multiplication with a scalar value, and inverse of a matrix(Use operator overloading). Use dynamic constructors and destructors for allocation and de allocation of memory. Overload function call operator () to allow the more common form of double array subscripting. Instead of saying x[row][column] for an array of objects, overload the function call operator to allow the alternate form x(row, column)

(10) Define a class of your own choice. Overload all the operators supported by C++, which can be overloaded for your class. List all the operators which can not be overloaded for your class, with proper justification.

Type Conversions

(1) Construct a class distance having member variables int feets and int inches. Design the class to make the following possible :

(i) to convert this class into the basic data type int which will represent the total no. of inches of the class.
(ii) to convert a basic data type int to distance class
(iii) to convert distance class to length class having member variables int meters and int centimeters

Write a C++ program to test your class.

(2) Define a class Integer containing an int variable and also two more variables, UpperBound and LowerBound. Provide a function Validate to the same class such that when that function is called, it checks to see if the value is between upper and lower bound. Integer class should be defined such that when we write following code,

```
Integer Int1(5);
int int1;
int1 = Int1;
```

Should work properly; i.e. it casts int to Integer object if validated.

(3) Create two classes DM and DB which store the value of distances. DM stores distances in metres and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object DM with another object 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.

(4) Define two class Polar and Rectangle to represent points in the Polar and Rectangle systems. Use conversion routines to convert from one system to another.

(You need to use the following trigonometric formulae:

```
x = r * cos(a)
y = r * sin(a)
a = atan(x/y)  // arc tangent
r = sqrt(x*x + y*y)
```

-----------------------------------Assignment-3-------------------------------------

Inheritance

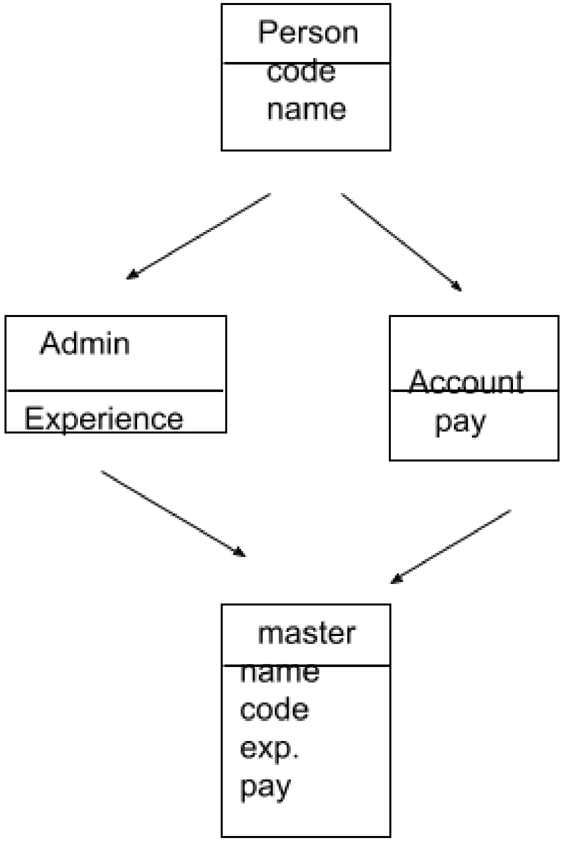
(1) Define a class Employees. Also define classes of MaleEmp and FemaleEmp inheriting from that. Define classes Officers, Clercks and peons again inheriting from Employee class. Define an array which contains 10 different types of employees. Define a function ReadDetails() in all above classes. All array elements should be able to be accessed in the same routine irrespective of their type.

(2) An educational institution wishes to maintain a database of its employees. The database is divided into a number of classed whose hierarchical relationships are shown in fig-1. The figure also shows the minimum information required for each class. Specify all the classes and define function to create the database and retrieve individual information as and when required. Write parameterized constructor for each class in the hierarchy.

(3) The database created in Ex-2 does not include educational information of the staff. It has been decided to add this information to teachers and officers (and not for typists) which will help the management in decision making with regards to training, promotion , etc. Add another data class called education that

holds two pieces of educational information , namely highest qualification in general education and highest professional qualification. This class should be inherited by the class teacher and officer. Modify the program of Ex 2 to incorporated these additions.

(4) Consider a class network of fig . The class master derives information from both account and admin classes which in turn derived derive information from the class person. Define all the four classes and write a program to create , update and display the information contained in master objects.



(Fig – 2)

(5) In Exercise 4, the classes teacher , Officer and typist are derived from the class staff. As we know, we can use container classes in place of inheritance in some situations. Redesign the program of Ex-8.21 such that the classes teacher, officer and typist contain the objects of staff. Write parameterized constructor for each class in the hierarchy.

Exercises from Text Book :

Run Time Polymorphism

(1) Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive three specific classes calledtriangle, rectangle and circle from the base shape. Add to the base class, a member function get_data( ) to initialize base class data members and another member function display_area( ) to compute and display the area of figures.Make display_area( ) as a virtual function and redefine this function in derived classes to suit their requirements. Using these three classes design a program that will accept dimensions of a triangle or rectangle interactively and store it in one array . After having read all the input display the area of all the figures whose area has been read in the program. Remember the two values given as input will be treated as lengths of two sides in the case of rectangle and as base and height in case of triangle. In case of circle only

one value should be accepted which will be taken as the radius and the default value of the next parameter should be 0.

(2) Consider a class student from which the classes test and sports are derived. The class student has the name and rollno of the student. The class test has the marks of the internal test and the sports class has the marks of the sports test. The class student contains a virtual function display() which are implemented in the classes test and sports. Write a program which will take relative information and display it using pointer of the base class.

(3) ABC publishing company markets both book and audio cassette versions of its work. Create a class called publication that stores the title(a string) and price(type float) of a publication. From this class derive two classes : book , which adds a page count (type int); and tape, which adds playing time in minutes (type float). Write a main program that reads both book and tape information in one array. When the user has finished entering data for all books and tapes, displays the resulting data for all the books and tapes entered. Also count no of book and cassette entries in the array using runtime identification feature of C++.

Exercises from Text Book

<u>Templates and Exception Handling</u>

(1) Write a generic function that will sort a character string, integer and float value. Create a menu with appropriate options and accept the values from the user.

(2) Implement binary search as a generic function (function template). The function should take arguments as array name, the size and the element to be searched.

(3) Write a template function called find(). This function searches an array for an object. It returns either the index of the matching object (if one is found) or -1 if no match is found.

(4) Write a object oriented program to implement a generic Stack. Incorporate all the possible operation on Stack in the program. Rework stack class so that stack overflows are handled as exceptions.

(5) Write a object oriented program to implement a generic Queue. Incorporate all the possible operation on Queue in the program.

(6) Write a program to create template class called "Safearray". Rules for this class are as follows :

(1) Size is equal to 100
(2) Index from 0 to size -1
(3) If array is sought outside bound the program aborts
(4) Function safeput() is used to assign value to an array element.
(5) Function safeget() is used to return the array element.

Make necessary provisions so the program terminates gracefully when unsafe action is attempted.

(7) Define a class Integer containing an int variable and also two more variables, UpperBound and LowerBound. Provide a function Validate to the same class such that when that function is called, it checks

to see if the value is between upper and lower bound (Use exception Handling) . Integer class should be defined such that when we write following code,

```
Integer Int1(5);
int int1;
 int1 = Int1;
```

Should work properly; i.e. it casts int to Integer object if validated.

(8) Create and demonstrate a generic linked list class.

(9) Design a stack class so that it can store pairs of different type of objects on the stack. Demonstrate your solution. Rework stack class so that stack overflows are handled as exceptions.

(10) Implement template sort with a non type size.

(11) Use Time class to provide overloaded -. Here the time query is also to be recorded in file. Use C text file to store the query. If a calling function provides expression Time1 – Time2, then operator – function should throw an exception if Time2 is a later time then Time1. Before throwing exception, though, the operator – function should close the file.