

## Assignment Part 2

16. Write a simple class that represents a class of geometrical points each of which has three coordinates. The class should have appropriate constructor(s). Also add a member function distance() that calculates Euclidian distance between two points. Now create two points, find the distance between them and print it.
17. Write a class for the geometrical shape rectangle. Write suitable constructors and member functions. Add a member function area() that calculates the area of a rectangle. Create 4 rectangles and print their respective area.
18. Write a class that represents a class of wireless device. A device has a location (point object may be used), a fixed unique id, and a fixed circular transmission range. Write suitable constructors and member functions for this class. Instantiates 10 such devices. Choose location (coordinates) and transmission range of the devices randomly. Now, for each of these devices, find the neighbor devices (i.e. devices that belong to the transmission range). Suppose, all of these devices have moved to a new location (randomly chosen). Find out the new set of neighbors for each of these devices.
19. Write a class Vector for one dimensional array. Write suitable constructor/copy constructor. Also add member functions for perform basic operations (such as addition, subtraction, equality, less, greater etc.). Create vectors and check if those operations are working correctly.
20. Write a class IntArray for one dimensional integer array. Implement the necessary constructor, copy constructor, and destructor (if necessary) in this class. Implement other member functions to perform operations, such adding two arrays, reversing an array, sorting an array etc. Create an IntArray object having elements 1, 2 and 3 in it. Print its elements. Now, create another IntArray object which is an exact copy of the previous object. Print its elements. Now, reverse the elements of the last object. Finally print elements of both the objects.
21. Create a simple class SavingsAccount for savings account used in banks as follows: Each SavingsAccount object should have three data members to store the account holder's name, unique account number and balance of the account. Assume account numbers are integers and generated sequentially. Note that once an account number is allocated to an account, it does not change during the entire operational period of the account. The bank also specifies a rate of interest for all savings accounts created. Write relevant methods (such as withdraw, deposit etc.) in the class. The bank restricts that each account must have a minimum balance of Rs. 1000. Now create 100 SavingsAccount objects specifying balance at random ranging from Rs. 1,000 to 1,00,000. Now, calculate the interest for one year to be paid to each account and deposit the interest to the corresponding balance. Also find out total amount of interest to be paid to all accounts in one year.
22. Write some programs to understand the notion of constant member functions, mutable data members etc.

23. Write the definition for a class called Complex that has private floating point data members for storing real and imaginary parts. The class has the following public member functions:

setReal() and setImg() to set the real and imaginary part respectively.

getReal() and getImg() to get the real and imaginary part respectively.

disp() to display complex number object

sum() to sum two complex numbers & return a complex number

Write main function to create three complex number objects. Set the value in two objects and call sum() to calculate sum and assign it in third object. Display all complex numbers.

24. Complete the class with all function definitions for a stack

```
class Stack {
    int *buffer, top;
public :
    Stack(int);           //create a stack with specified size
    void push(int);       //push the specified item
    int pop();            //return the top element
    void disp();          //displays elements in the stack in top to bottom order
};
```

Now, create a stack with size 10, push 2, 3, 4 and 5 in that order and finally pop one element. Display elements present in the stack.

25. Complete the class with all function definitions for a circular queue

```
class Queue {
    int *data;
    int front, rear;
public :
    Queue(int ); //create queue with specified size
    void add(int);//add specified element to the queue
    int remove();//delete element from the queue
    void disp(); //displays all elements in the queue(front to rear order)
};
```

Now, create a queue with size 10 add 2, 3, 4 and 5 in that order and finally delete two elements. Display elements present in the stack.

26. Write a class for your Grade card. The grade card is given to each student of a department per semester. The grade card typically contains the name of the department, name of the student, roll number, semester, a list of subjects with their marks and a calculated CGPA. Create 60 such grade cards in a 3<sup>rd</sup> semester with relevant data and find the name and roll number of student having highest CGPA.

27. Create a class for Book. A book has unique isbn (string), title, a list of authors, and a price. Write relevant functions. Now write a class BookStore which has a list of books. There may be multiple copies of a book in the book store. Write relevant member functions. Write a function books() that returns list of unique isbn numbers of the books, a function noOfCopies() that returns the number of copies available for a given isbn number and a function totalPrice() that returns the total price of all the books. Create a book store having a number of books (multiple copies). Now, for each book, print number of copies of that book along with its title.