**Sample questions**

Q1. Hours, minutes, and seconds (HH:MM:SS) should be the three data members of the class Time. Use the overload + operator to add two more time objects. The first line of input comprises hours, minutes, and seconds (separated by spaces). The second line provides the same information again, again separated by spaces: hours, minutes, and seconds. Your job is to sum these two times and output the result.

Input Format:  
First line : hours , minutes and seconds (space separated)

Second line: hours , minutes and seconds (space separated)

Output Format:

Addition of two time

Q. Create a matrix class to represent two-dimensional arrays. To add two 2D arrays, use the overloading of the + operator. To accomplish addition, include the overloaded operators and required member functions. Give a succinct implementation of how it's done.

Input:

First line contains number of rows ‘M’ and number of columns ‘N’

Second line contains MxN elements on first matrix

Third line contains MxN elements on second matrix

Output:

Addition of two matrix after operator overloading

Q. Create a C++ class called ArraySorter, which has a constructor, destructor, and member function called sortArray. This function uses the Bubble Sort algorithm to sort an array of numbers in ascending order. The class member variable should be initialised by the constructor using an integer array that is passed in as input. Any dynamic memory allocation should be cleaned up by the destructor. The Bubble Sort algorithm should be used by the sortArray function to sort the array.

Input:

First line contains length of array

second line contain array elements

You are required to design a C++ program for managing a library system. Implement a class Book to represent a book with the following attributes: title, author, and publicationYear. Q.

Q. Implement the necessary constructor and destructor for the Book class. The constructor should initialize the attributes, and the destructor should display a message indicating the destruction of the object.

Implement the class and demonstrate the usage of the constructor and destructor by creating objects and observing their creation and destruction.

Implement the following class:

**Book Class:**

Properties: title (string), author (string), publicationYear (integer)

Methods:

**Book(string title, string author, int publicationYear)** - A constructor to initialize the title, author, and publicationYear of the book.

Destructor - Display a message indicating the destruction of the object.

**Input:**

First line contains books name

second line contains author name

Third line contains year of publication

**Example Test case:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| Harry Potter  J.K. Rowling  1965 | Book Created: Harry Potter  Book author: J.K. Rowling  Destruction of Book: Harry Potter |

Q> Creating a class named Matrix that represents a two-dimensional integer matrix is the assignment assigned to you. It is your responsibility to overuse the \* operator to multiply two matrices and the + operator to add two matrices. You also need to write a printMatrix function that outputs the matrix's elements.

Print the + operator result on two matrices at the end, followed by the \* operator result, separated by an empty line.

Q> Two strings, String1 and String2, are provided to you.

Create the StringManipulator C++ class, which has the following features:

(i) To compare two instances of StringManipulator, overload the > operator. If String1 is lexicographically larger than String2, it should return true; if not, it should return false.

(ii) To compare two instances of StringManipulator, overload the < operator. If String1 is lexicographically smaller than String2, it should return true; if not, it should return false.

(iii) To compare two instances of StringManipulator, overload the == operator. If String1 and String2 are equal, it should return true; if not, it should return false.After completing each process, print the outcome on a different line.

Q>Create a class Shape with method getArea(), inherit two classes: Square and Circle from Shape and override the getArea() method from Shape in Square and Circle to Calculate the area of Circle and Square by creating

an instance of both Square and Circle.

Q> create a class Complex , overload the operators ++ , – (both pre increment and post increment) which increments and decrements the values of real and imag (both) accordingly.

Finally display the Complex number after performing post increment and pre decrement in the format : real + i imag

Q> Make a class named Fruit with a data member to calculate the number of fruits in a basket. Create two other class named Apples and Mangoes to calculate the number of apples and mangoes in the basket. Print the number of fruits of each type and the total number of fruits in the basket.

Q>Create two classes named Mammals and MarineAnimals. Create another class named BlueWhale which inherits both the above classes. Now, create a function in each of these

classes which prints "I am mammal", "I am a marine animal" and "I belong to both the categories:

Mammals as well as Marine Animals" respectively.

Now, create an object for each of the above class

and try calling

1 - function of Mammals by the object of Mammal

2 - function of MarineAnimal by the object of MarineAnimal

3 - function of BlueWhale by the object of BlueWhale

4 - function of each of its parent by the object of BlueWhale

**Sample questions ET**

Q1. You have given two sorted array you need to merge them and create a new array in which all elements in given arrays must present in new array but in sorted.

**Input:**

First line contains two integers m and n which defines the length of two given sorted array

Second line contains the elements of first array

Third line contains the elements of second array

**Output:**

Print M+N elements in resultant array in sorted order

Example test case:

|  |  |
| --- | --- |
| Input | Output |
| 3 4  2 3 7  1 3 4 8 | 1 2 3 3 4 7 8 |

Q2. Given an array of integers. All numbers occur twice except one number which occurs once. Find the number in O(n) time & constant extra space.

**Input :**

First line contains a integer N which specify size if array

Second line contains N integer elements of array

**Output:**

Print a Integer which occurring only once in given array

Example test case:

|  |  |
| --- | --- |
| Input | Output |
| 7  1 3 2 3 4 4 2 | 1 |

Q3. Given an integer x, return true if x is a palindrome, and false otherwise.

**Input:**

First line contains an integer X

**Output:**

Print true or false

Example test case:

|  |  |
| --- | --- |
| Input | Output |
| 14341 | true |

Q.4

Implement a C++ program demonstrating polymorphism using a base class ‘Shape’ and its derived classes ‘Circle’ and ‘Rectangle’. The Shape class should have a virtual function ‘calculateArea()’ to calculate and display the area of the shape. Implement the derived classes with appropriate data members and member functions to calculate the area of a circle and rectangle respectively.

Write a program that creates objects of Circle and Rectangle classes and demonstrates polymorphism by calling the calculateArea() function on each object.

**Input:**

First line contains radius of circle

second line contain length and breadth of rectangle respectively

**Example Test case:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4.0  6.0 5.0 | Area of the circle: 50.24  Area of the rectangle: 30 |

Q5. Create a C++ class Matrix representing a 2x2 matrix. Implement operator overloading for multiplication operations for matrices. Provide a member function to display the matrix. Write appropriate test cases to demonstrate the functionality of the overloaded operators.

**Input:**

First line contains four integers which are elements of first 2x2 matrix

Second line contains four integers which are elements of second 2x2 matrix

**Example Test case:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 3 2 4  3 2 6 5 | Multiplication:  21 17  30 24 |

Q6. You're given an N x M matrix where each row and column is sorted in ascending order. Your task is to determine if a given number x is present in the matrix.

**Input Format:**

-The first line contains two space-separated integers, N and M, denoting the number of elements in a row and column, respectively.

-The second line of each test case contains N \* M space-separated integers representing the elements in the matrix in row-major order.

-The third line of each test case contains a single integer x, which is the element to be searched within the matrix.

**Output Format:**

Print 1 if the element is present in the matrix, else 0.

Constraints: 1 <= N,M <= 300 <= A[i] <= 100

|  |  |
| --- | --- |
| Input | Output |
| 3 3  3 30 38  44 52 54  57 60 69  62 | 0 |

Q.7 You are given N integers, your task is to sort the array in decreasing order using priority\_queue container.

Input Format:  
First Line: an integer N denoting the number of elements to sort

Second Line: N space separated integers.

Output Format:  
N integers sorted in decreasing order

Constraints:  
1 <= N <=1000

|  |  |
| --- | --- |
| Input | Output |
| 10  3 5 1 7 8 18 2 7 18 100 | 100 18 18 8 7 7 5 3 2 1 |

Q.8 You are given N integers, your task is to find kth largest element in the array

Input Format:  
First Line: an integer N denoting the number of elements in array and integer K determine Kth greatest element

Second Line: N space separated integers.

Output Format:  
Kth largest element

Constraints:  
1 <= N <=1000

|  |  |
| --- | --- |
| Input | Output |
| 10 3  3 5 1 7 8 18 2 7 18 100 | 18 |

Q 9. Given an array of **N** elements, where each element is at most K away from its target position, devise an algorithm that sorts in O(N log K) time.

Hint: Use priority queue

Input:

First line contains length of array n and k which represent the how much the array element away from its target position.

Second line contains the all n elements present in array

Output:

All elements of sorted array.

Example test case:

|  |  |
| --- | --- |
| Input | Output |
| 6 3  2 6 3 12 56 8 | 2 3 6 8 12 56 |

Q10. Find the frequency of all unique elements in vector with

Input:

First line contains the length of vector N.

Second line contains N elements of vector.

Output:

Frequency of every unique element in vector

Example test case:

|  |  |
| --- | --- |
| Input | Output |
| 10  1 2 1 4 5 3 3 2 1 4 | 3 2 2 2 1 |

Q11. Implement a C++ function combinationSum that takes in a vector of distinct integers candidates, a target integer target, and returns a vector containing all unique combinations of candidates where the chosen numbers sum up to the target. Each number in candidates can be used an unlimited number of times.

The function should avoid duplicate combinations. The order of combinations in the output vector does not matter.

Input:

First line contains the length of vector N and target T, which sum of few elements of vector

Second line contains N elements of vector.

Output:

All combination vectors whose elements make sum up equals to target

Example test case:

|  |  |
| --- | --- |
| Input | Output |
| 10 7  4 5 3 3 2 1 4 | 4 3  4 2 1  5 2  3 3 1  3 4  2 1 4 |