**Final assignment for C++**

Q1: Vectors: Write a C++ program, there you take two vectors of names. The program should return a vector that has all the names only once.

Example: vector1 = [Virendra Sachin, Saurav, Rahul, Laxman]

vector2 = [Harbhajan, Ashish, Rahul, Yuvraj, Saurav]

output = [Virendra Sachin, Saurav, Rahul, Laxman, Harbhajan, Ashish, Yuvraj]

Q2: The given code has a base class charInput, it has member methods add() and getvalue(). The class numInput inherits charInput. You need to implement:

add(): it should be inside charInput, this adds the given character to the current value

getValue(): it should be inside numInput, it returns the current value

add(): it should be inside numInput, this must override the base class method so that each non-numeric char is ignored.

#include <iostream>

#include <string>

using namespace std;

class charInput{

public:

void add(char c) { }

std::string getValue() { return NULL; }

};

class numInput : public charInput { };

int main(){

charInput\* inputC = new charInput();

numInput\* inputN = new numInput();

input->add('1');

input->add('a');

input->add('0');

cout << input->getValue();

// the output should be 10

}

Q3: You have a binary string with length N. You are allowed to do swap() operations of two elements only if the index parity of both elements is the same. This means that: you can swap() an element at index 2, with any element at index4, 6, 8 etc. (even parity), similarly the element at index 3 can be swapped with the element at index1, 5,7,9 etc.

Ex: Given string 1110, you can swap 2nd and 4th char to get 1011. But, you can never get 1101 if you follow the swap rules.

Now, the task is to find the number of times substring 01 can occur for all such possible swaps.

Ex: 1110 does not have any substring of 01. Now you can swap2nd and 4th to get 1011, this will have ‘one’ occurrence of 01. We can now perform more swaps but we will never get any more 01 substrings here.

Input: 00100 output should be 1, for: 01010-> 2, for 10001-> 2

Q4: In this question, you are given a binary string of length T. Now you need to create two permutations of this string: S1 and S2 such that the ‘longest common subsequence’ between the two newly created strings is smallest.

Ex: Given string: 101, you can find S1: 110 and S2: 011, The longest common subsequence between S1 and S2 is 11 and the length is 2. There cannot be any two permutations of the given string where the LCS is less than 2

Ex: Given 0111, then S1 should be: 1101, and S2: 0111, the smallest LCS will be 2 char long.

Q5: There are *p* white balls and *q* red balls, you need to put them into *r* bags such that:

No bag should be empty

A bag can contain only red balls or white balls, not both

No red ball can be left alone in a bag

Find the number of ways to put the balls in bags. All bags are identical, but each ball is unique (or numbered).

Since the total number can be very large for larger values of p and q, you can take modulo 99,999,999

Ex: White balls: 2, red balls: 1, bags: 4. Answer: 0

White balls: 2, red balls: 4, bags: 4. Answer: 3

Q6: The grade of a student can be based on the following criteria:

Attendance must be more than 50%

Tests scores must be more than .70

Total score must be more than 5600 (just go with this number)

The grades are awarded as follows:

Grade is 10: if all conditions are met

Grade is 9: if conditions 1 and 2 are met

Grade is 8: if conditions 3 and 3 are met

Grade is 7: if conditions 1 and 3 are met

Grade is 6: if only one condition is met

Grade is 5: if none of the conditions are met

Write a cpp program to display the grades, based on the scores students secured in individual criteria.

Ex: input 53, 0.6, 5602 res: 10

Input 45, 0, 4500 res: 6

Q7: Write a c++ program, this will take two input values for length and breadth of a chocolate bar. You will get more money if you sell the chocolate in square shapes. So, you would try to divide the chocolate into minimum possible number of square pieces, such that each piece has the same area and the piece divide the chocolate bar (original big bar) perfectly. Note that this will give you most profit for the entire bar.

Ex: Length: 10, breadth: 15, answer = 6

Length: 4, breadth: 6, answer = 6

Q8: There are two processes, A and B, that can access a common variable x. They can access it in sequence, A first and then B, or B first then A. But in one day they access it only once. A logbook is maintained by the OS showing which process accessed x when.

You got a series of entries of the log, but you want to make sure that this log has not been altered by anyone. Your output is the answer to the question: Is the log valid or not?

Ex: Input: AB, output: Yes

Input: ABAABB, output: No

SET-1: Q1, Q4, Q6

SET-2: Q2, Q3, Q7

SET-3: Q3, Q4, Q8

SET-4: Q2, Q5, Q6