C++ Lab Assignments 'ABC' 8-2-2023

1. **Structure Array**

Given an unsorted integer array containing many duplicate elements, first code to rearrange it such that the same element appears together and the relative order of the first occurrence of each element remains unchanged. You should use the same array only, should not use any other arrays.

For example,
Input1: A={1,2,3,1,2,1}
Output1: A={1,1,1,2,2,3}
Input2: A={5,4,5,5,3,1,2,2,4}
Output2: A={ 5, 5, 5, 4, 4, 3, 1, 2, 2 }

Write functions to read and print the array.
Now code a function called **array-to-struct(…)** to store the contents of the rearranged array in an array of structures S[].
The structure filed values are: number(integer), frequency(integer).
Write a function to print the array S as shown below.
The functions only should be called in main function.

Contents of S : for input1	**Contents of S : for input2**
1 3 2 2 3 1	5 3 4 2 3 1 1 1 2 2

2. **Structures-Pair**

Given set of pairs of integers, code to print all symmetric pairs, i.e., pairs that mirror each other.
For instance, pairs (x, y) and (y, x) are mirrors of each other. A pair is to be stored as a structure.
Type input as 3 4 1 2 5 2 7 10 4 3 2 5 to read into the array of structure.
Input: {3, 4}, {1, 2}, {5, 2}, {7, 10}, {4, 3}, {2, 5}
Output: (only print , no need to store anywhere)

{3, 4} | {4, 3}
{5, 2} | {2, 5}

3. **Structures-intervals**. Given a set of intervals, code to print all non-overlapping intervals after merging the overlapping intervals.

Input: {1, 5}, {2, 3}, {4, 6}, {7, 8}, {8, 10}, {12, 15}

Output: Intervals after merging overlapping intervals are {1, 6}, {7, 10}, {12, 15}.

4.

```
#include<iostream>
using namespace std;
int  Special( int w , int *x, int &y, int* &z)
{ static int B[4];
      int t = z;
      t =  y ;
      x = y +  z ;
     y =  z +  t ;
     z =   x + y ;
      w = t + w;
     cout << w <<" "<<  x <<" "<< y <<" "<<  z << endl;
     B[0] = w ; B[1]= x ; B[2] =y ; B[3] = z;
     return B;
}
int main()
{  int A[]={5,6,7,8};
   int *p=A;
   p = Special( A[0], A[1] ,(A+2), p ) ;
         for(int i=0;i<4;i++)
         cout<<(A+i)<<" ";
         cout<< endl;
         for(int i=0;i<4;i++)
         cout<<p[i]<<" ";
         return 0;
}
```

Note :

i) The **bold code statements** should not be changed.

ii) You **may have to add symbols only**, i.e. *** or &** as part of the change.

ii) You **should not add** any **statements**.

You may have to **adjust the other statements** (i.e. non-bold statements) in such a way that **any valid output should be printed.**

Syntactically adjust the non-bold statements with all possibilities and get different outputs.

A genius is often merely a talented person who has done all of his or her homework."

– Thomas Edison

"Trust yourself,
you know more than you think you do."

Benjamin Spock,

"You are braver than you believe,
Stronger than you seem and
Smarter than you think."

– A.A. Milne,

"The one of CSE who does not code has no advantage over the one who is not of CSE"

"Work while they sleep.
Learn while they party.
Save while they spend.
Think while they talk.
Code while they comment.
Win while they wish.
Live like they dream."

~KR