L4: Language Tools Practice Questions

1. Hussain set - https://www.codechef.com/problems/COOK82C
2. https://www.codechef.com/problems/VOTERS
3. https://www.codechef.com/problems/PERMPAL
4.

1-Ass : Love for Characters

Send Feedback

Ayush loves the characters ‘a’, ‘s’, and ‘p’. He got a string of lowercase letters and he wants to find out how many times characters ‘a’, ‘s’, and ‘p’ occurs in the string respectively. Help him find it out.

Input:

First line contains an integer denoting length of the string.
Next line contains the string.

Constraints:

1<=n<=10^5
‘a’<= each character of string <= ‘z’

Output:

Three space separated integers denoting the occurrence of letters ‘a’, ‘s’ and ‘p’ respectively.

Sample Input:

6
aabsas

Sample output:

3 2 0

```
#include<bits/stdc++.h>
using namespace std;
void aspcount(string str, int l){
    int a = 0,s = 0,p=0,i = 0;

    while(str[i] != '\0'){
        char ch = str[i];
        if(ch == 'a'){
            a++;
        }else if(ch == 's'){
            s++;
        }else if(ch == 'p'){
            p++;
        }
        i++;
    }
    cout << a << " " << s << " " << p << endl;
}

```

```
int main()
{
    int l;
    cin >> l;
    string str;
    cin >> str;
    aspcount(str,l);
          return 0;
}
```

2-Ass : Different Names

Send Feedback

In Little Flowers Public School, there are many students with same first names. You are given a task to find the students with same names. You will be given a string comprising of all the names of students and you have to tell the name and count of those students having same. If all the names are unique, print -1 instead.

Note: We don't have to mention names whose frequency is 1.

Input Format:

The only line of input will have a string ‘str’ with space separated first names of students.

Output Format:

Print the names of students along with their count if they are repeating. If no name is repeating, print -1

Constraints:

1 <= |str| <= 10^5

Time Limit: 1 second

Sample Input 1:

```
Abhishek harshit Ayush harshit Ayush Iti Deepak Ayush Iti
```

Sample Output 1:

```
harshit 2
Ayush 3
Iti 2
```

Sample Input 2:

```
Abhishek Harshit Ayush Iti
```

Sample Output:

```
-1
```

```
#include<bits/stdc++.h>
#include <iostream>
using namespace std;
void stringcount(string str){
    map<string,int> mymap;
    string temp;
    stringstream X(str);

    while( getline(X,temp,' ') ){
        mymap[temp]++;
    }
    bool flag = false;
    map<string,int> :: iterator it;
    it = mymap.begin();

    while(it != mymap.end() ){
        if(it->second >= 2){
            flag = true;
            cout << it->first << " " << it->second << endl;
        }
        it++;
    }

    if(!flag){
        cout << -1;
    }

}
int main() {
    // Write your code here
    string str;
    getline(cin,str);
    stringcount(str);
}
```

3-Ass : Extract Unique characters

Send Feedback

Given a string S, you need to remove all the duplicates. That means, the output string should contain each character only once. The respective order of characters should remain same, as in the input string.

Input format:

The first and only line of input contains a string, that denotes the value of S.

Output format :

The first and only line of output contains the updated string, as described in the task.

Constraints :

0 <= Length of S <= 10^8

Time Limit: 1 sec

Sample Input 1 :

```
ababacd
```

Sample Output 1 :

```
abcd
```

Sample Input 2 :

```
abcde
```

Sample Output 2 :

```
abcde
```

```
#include<bits/stdc++.h>
string uniqueChar(string str) {
	    // Write your code here
    string str1 = "";
    map<char,int> mymap;
    int i = 0;
    while(str[i] != '\0'){
        mymap[str[i]] = 1;
        i++;
    }
    i = 0;
    while(str[i] != '\0'){

        if(mymap[str[i]]){
            str1 += str[i];
            mymap[str[i]] = 0;
        }
        i++;
    }

    return str1;
}
```

4-Ass : Warm Reception

Send Feedback

There is only one beauty parlour in the town CodingNinjasLand. The receptionist at the beauty parlor is flooded with appointment requests because the “Hakori” festival is round the corner and everyone wants to look good on it.

She needs your help. The problem is they don’t have chairs in reception. They are ordering chairs from NinjasKart. They don’t want to order more than required. You have to tell the minimum number of chairs required such that none of the customers has to stand.

Input Format :

First line contains the number of customers that will come. Second line contains N space-separated integers which represent the arrival timings of the customer. Third line contains N space-separated integers which represent the departure timings of the customer. Arrival and departure timings are given in 24-hour clock.

Constraints:

1<= N <= 100

Arrival and departure timings lie in the range [0000 to 2359]

Time Limit: 1 second

Output Format :

You have to print the minimum number of chairs required such that no customer has to wait standing.

Sample Test Cases:

Sample Input 1 :

```
5
900 1000 1100 1030 1600
1900 1300 1130 1130 1800
```

Sample Output 1:

```
4
```

Explanation:

4 because at 1100 hours, we will have maximum number of customers at the shop, throughout the day. And that maximum number is 4.

```
#include<bits/stdc++.h>
using namespace std;
int main() {
        // Write your code here
    int n;
    cin >> n;
    int arrival[n];
    int departure[n];

    for(int i=0;i<n;i++){
        cin >> arrival[i];
    }
    for(int i=0;i<n;i++){
        cin >> departure[i];
    }
    sort(arrival,arrival+n);
    sort(departure,departure+n);
    map<int,int> m;
    map<int,int>::iterator it;
    for(int i=0;i<n;i++){
        int count = i+1;
        int temp = arrival[i];
        for(int j=0;j<n;j++){
            if(departure[j] <= temp){
                count--;
            }
        }
        m[temp] = count;
    }
    int max = 0;
    for(it=m.begin();it!=m.end();it++){
        if(it->second > max){
            max = it->second;
        }
    }
    cout << max << endl;
}
```

5-Ass : Tell the positions

Send Feedback

In a class there are ‘n’ number of students. They have three different subjects: Data Structures, Algorithm Design & Analysis and Operating Systems. Marks for each subject of all the students are provided to you. You have to tell the position of each student in the class. Print the names of each student according to their position in class. Tie is broken on the basis of their roll numbers. Between two students having same marks, the one with less roll number will have higher rank. The input is provided in order of roll number.

Input Format:

First line will have a single integer ‘n’, denoting the number of students in the class.
Next ‘n’ lines each will have one string denoting the name of student and three space separated integers m1, m2, m3 denoting the marks in three subjects.

Output Format:

Print ‘n’ lines having two values: First, the position of student in the class and second his name.

Constraints:

1 <= n <= 10^5
0 <= m1, m2, m3 <= 100

Sample Input:

```
3
Mohit 94 85 97
Shubham 93 91 94
Rishabh 95 81 99
```

Sample Output:

```
1 Shubham
2 Mohit

3 Rishabh
```

```
#include<bits/stdc++.h>
using namespace std;

struct student{
    int roll;
    string name;
    int marks;
};

bool compare(student c1,student c2){
    if(c1.marks > c2.marks){
        return 1;
    }else if(c1.marks < c2.marks){
        return 0;
    }else if(c1.marks == c2.marks){
        if(c1.roll < c2.roll){
            return 1;
        }else{
            return 0;
        }
    }
}

int main()
{
    long n;
    cin >> n;
    student arr[n];
    int roll = 1;
    for(int i=0;i<n;i++){
        string name;
        int m1,m2,m3;
        cin >> name;
        cin >> m1 >> m2 >> m3;
        int marks = m1+m2+m3;
        arr[i] = {roll,name,marks};
        roll += 1;
    }
    sort(arr,arr+n,compare);// important h, ye dekh lo how the third parameter compare works.
    int count = 0;
    for(int i=0;i<n;i++){
        count++;
        cout << count << " " << arr[i].name << endl;
    }

             return 0;
}
```