Sock Merchant ☆

Editorial by Shafaet

To solve this challenge, we go through each color i and count its frequency, $f(i)$. Once we've calculated all the frequencies, we calculate the number of pairs of each kind of sock as $\frac{f(i)}{2}$ (using integer division). Finally, we print the total sum of all pairs of socks.

Set by Shafaet

Problem Setter's code:

C++

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


int main() {
        int n;
    cin>>n;
        int freq[101] = {};
        for(int i = 0; i < n; i++) {
        int c;
        cin >> c;
        freq[c]++;
    }

        int res = 0;
        for(int i = 0; i <= 100; i++){
         res += freq[i] / 2;
     }
        cout << res << endl;
        return 0;
}
```

Python 2.7

```
from itertools import groupby
n = int(raw_input())
c = map(int, raw_input().split())

ans = 0
for val in [len(list(group)) for key, group in groupby(sorted(c))]:
    ans = ans + val/2
print ans
```

Tested by AllisonP

Problem Tester's code:

Java

```
import java.util.*;


class Solution {
```

```
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int n = scan.nextInt();
        HashMap<Integer, Integer> colors = new HashMap<Integer, Integer>();

        while(n-- > 0) {
            int c = scan.nextInt();
            Integer frequency = colors.get(c);

            // If new color, add to map
            if(frequency == null) {
                colors.put(c, 1);
            }
            else { // Increment frequency of existing color
                colors.put(c, frequency + 1);
            }
        }
        scan.close();

        // Count and print the number of pairs
        int pairs = 0;
        for(Integer frequency : colors.values()) {
            pairs += frequency >> 1;
        }
        System.out.println(pairs);
    }
}
```

Feedback

Was this editorial helpful?

Yes No