# Sock Merchant

by Shafaet

Problem | Submissions | Leaderboard | Discussions | Editorial

John's clothing store has a pile of $n$ loose socks where each sock $i$ is labeled with an integer, $c_i$, denoting its color. He wants to sell as many socks as possible, but his customers will only buy them in matching pairs. Two socks, $i$ and $j$, are a single matching pair if $c_i = c_j$.

Given $n$ and the color of each sock, how many pairs of socks can John sell?

**Input Format**

The first line contains an integer, $n$, denoting the number of socks.
The second line contains $n$ space-separated integers describing the respective values of $c_0, c_1, c_2, \ldots, c_{n-1}$.

**Constraints**

- $1 \leq n \leq 100$
- $1 \leq c_i \leq 100$

**Output Format**

Print the total number of *matching pairs* of socks that John can sell.

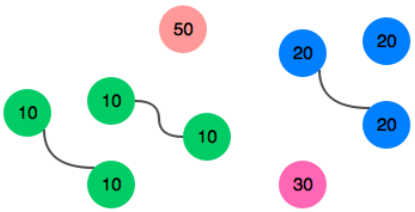
**Sample Input**

```
9
10 20 20 10 10 30 50 10 20
```

**Sample Output**

```
3
```

**Explanation**




As you can see from the figure above, we can match three pairs of socks. Thus, we print **3** on a new line.

Submissions: 18927
Max Score: 10
Difficulty: Easy

Rate This Challenge:
Thanks!
More

Current Buffer (saved locally, editable) C#

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
class Solution {

    static void Main(String[] args) {
     int n = Convert.ToInt32(Console.ReadLine());
        string[] c_temp = Console.ReadLine().Split(' ');
        int[] c = Array.ConvertAll(c_temp, e => int.Parse(e));
        Dictionary<int, int> frec = new Dictionary<int, int>();
        foreach (int elem in c)
        {
            if (frec.ContainsKey(elem))
            {
                frec[elem]++;
            }
            else
            {
                frec[elem] = 1;
            }
        }
        int ans = 0;
        foreach (KeyValuePair<int, int> kvp in frec)
        {
            int emparejados = kvp.Value;
            //if (emparejados % 2 != 0)
            //{
            //    emparejados--;
            //}
            ans += (int)Math.Floor( (double) emparejados / 2);
        }
        Console.WriteLine(ans);
    }
}

```

Line: 33 Col: 32

Upload Code as File  Test against custom input  Run Code  Submit Code

## Congrats, you solved this challenge!

- ✔ Test Case #0
- ✔ Test Case #1
- ✔ Test Case #2
- ✔ Test Case #3
- ✔ Test Case #4
- ✔ Test Case #5
- ✔ Test Case #6
- ✔ Test Case #7

Next Challenge