Zigzag Array

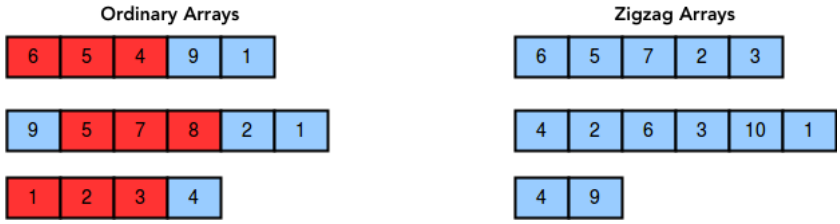
by kevinsogo

Problem Submissions Leaderboard Discussions

We say an array of n distinct integers, $A = [a_0, a_1, \ldots, a_{n-1}]$, is *zigzag* if no three consecutive elements in the array are either increasing or decreasing.

In other words, if there are three elements a_i, a_{i+1}, a_{i+2} in the array such that $a_i < a_{i+1} < a_{i+2}$ or $a_i > a_{i+1} > a_{i+2}$, the array is not *zigzag*.

For example:



Given A, find and print the minimum number of elements you must remove to make the given array zigzag.

Input Format

The first line contains n, denoting the number of elements.
The second line contains n space-separated integers describing the respective values of $a_0, a_1, \ldots, a_{n-1}$.

Constraints

- $1 \leq n \leq 100$
- $1 \leq a_i \leq 100$
- The elements of A are distinct.

Output Format

Print the minimum number of elements you must remove to make the given array zigzag.

Sample Input 0

```
6
4 2 6 3 10 1
```

Sample Output 0

```
0
```

Explanation 0

The array $[4, 2, 6, 3, 10, 1]$ is already zigzag, so we return 0.

Sample Input 1

```
5
5 2 3 6 1
```

Sample Output 1

```
1
```

Explanation 1

The array $[5, 2, 3, 6, 1]$ is not zigzag because here $a_1 < a_2 < a_3$ $(2 < 3 < 6)$.

If we remove 6, the array becomes $[5, 2, 3, 1]$ which is zigzag. Because we only needed to remove one element, we return 1 as our answer.

Contest ends in 2 days

Submissions: 2348
Max Score: 20
Difficulty: Easy

Rate This Challenge:

More

Current Buffer (saved locally, editable) C#

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

    static int minimumDeletions(int[] a){
        // Complete this function
         if (a.Length < 3) return 0;

            int i = 0;

            int borrar = 0;

            while (i < a.Length)
            {
                if (i + 1 < a.Length && a[i] < a[i+1])
                {
                    int cont = 1;
                    while (i + 1 < a.Length && a[i] < a[i + 1])
                    {
                        i++;
                        cont++;
                    }
                    if (cont >= 3)
                    {
                        //borrar += (cont - 3);
                        borrar += (cont - 2);
                    }
                    i--;
                }
                if (i + 1 < a.Length && a[i] > a[i+1])
                {
                    int cont = 1;
                    while (i + 1 < a.Length && a[i] > a[i + 1])
                    {
                        i++;
                        cont++;
                    }
                    if (cont >= 3)
                    {
```

```
                        borrar += (cont - 2);
                    }
                    i--;
                }

                i++;

            }
            return borrar;
    }

    static void Main(String[] args) {
        int n = Convert.ToInt32(Console.ReadLine());
        string[] a_temp = Console.ReadLine().Split(' ');
        int[] a = Array.ConvertAll(a_temp,Int32.Parse);
        // Return the minimum number of elements to delete to make the array zigzag
        int result = minimumDeletions(a);
        Console.WriteLine(result);
    }
}

```

Line: 50 Col: 27

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