**Min-Max**

Attempted by: **1111**

/

Accuracy: **89%**

/

Maximum Score: **10**

/

5 Votes

Tag(s):

Basic Programming, Very-Easy

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

**ANALYTICS**

Given an array of integers . Check if all the numbers between minimum and maximum number in array exist's within the array .

Print **'YES'** if numbers exist otherwise print **'NO'**(without quotes).

**Input:**

Integer **N** denoting size of array

Next line contains **N** space separated integers denoting elements in array

**Output:**

Output your answer

**Constraints:**

1<= **N** <= 1000

1<= **a[i]** <= 100

**SAMPLE INPUT**

6

4 2 1 3 5 6

**SAMPLE OUTPUT**

YES

**Time Limit:**1.0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded when all the testcases pass.

**Allowed Languages:**C, C++, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Scala 2.11.8, Swift, Visual Basic

<https://www.hackerearth.com/practice/basic-programming/implementation/basics-of-implementation/practice-problems/algorithm/min-max-3/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static void Main(string[] args)

{

//int[] a = { 4, 2, 1, 3, 5, 6 };

int n = int.Parse(Console.ReadLine());

int[] a = Array.ConvertAll(Console.ReadLine().Split(' '), e => int.Parse(e));

string ans = "YES";

for (int i = a.Min(); i < a.Max(); i++)

{

if (!a.Contains(i))

{

ans = "NO";

break;

}

}

Console.WriteLine(ans);

Console.ReadLine();

}

}

}