Piling Up!

FAIR

51 more points to get your gold badge!

Rank: 125772 | Points: 349/400

Your Piling Up! submission got 50.00 points. Share Tweet

You are now 51 points away from the gold level for your python badge.

Try the next challenge | Try a Random Challenge

Problem | Submissions | Leaderboard | Discussions | Editorial

There is a horizontal row of n cubes. The length of each cube is given. You need to create a new vertical pile of cubes. The new pile should follow these directions: if $cube_i$ is on top of $cube_j$ then $sideLength_j \geq sideLength_i$.

When stacking the cubes, you can only pick up either the leftmost or the rightmost cube each time. Print "Yes" if it is possible to stack the cubes. Otherwise, print "No". Do not print the quotation marks.

Input Format

The first line contains a single integer T, the number of test cases.
For each test case, there are 2 lines.
The first line of each test case contains n, the number of cubes.
The second line contains n space separated integers, denoting the sideLengths of each cube in that order.

Constraints

$1 \leq T \leq 5$
$1 \leq n \leq 10^5$
$1 \leq sideLength < 2^{31}$

Output Format

For each test case, output a single line containing either "Yes" or "No" without the quotes.

Sample Input

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

Sample Output

```
Yes
No
```

Explanation

In the first test case, pick in this order: **left - 4, right - 4, left - 3, right - 3, left - 2, right - 1**.
In the second test case, no order gives an appropriate arrangement of vertical cubes. **3** will always come after either **1** or **2**.

Author sandyeep
Difficulty Medium
Max Score 50
Submitted By 36133

NEED HELP?

View discussions
View editorial
View top submissions

RATE THIS CHALLENGE

MORE DETAILS

Download problem statement
Download sample test cases
Suggest Edits

Change Theme Python 3

```
from collections import deque
n=int(input())
for i in range(n):
    n1=map(int,input())
    j=map(int,input().split())
    d=deque(j)
    c=[]
    ds=[]

    while len(d)>0:
        if d[0]>=d[-1]:

            c.append(d[0])
            d.popleft()
        else:
            c.append(d[-1])
            d.pop()
    ds=sorted(c,reverse=True)
    if ds==c:
        print("Yes")
    else:
        print("No")
```

Line: 22 Col: 9

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

You have earned 50.00 points!
You are now 51 points away from the gold level for your python badge.
72% 349/400

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

Test case 0
Test case 1
Test case 2
Test case 3
Test case 4

Compiler Message

Success

Input (stdin) Download

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

Expected Output Download

```
Yes
No
```