**Chef and Rainbow Array**Problem Code: **RAINBOWA**

******

[Submit](https://www.codechef.com/AUG17/submit/RAINBOWA)

Chef likes all arrays equally. But he likes some arrays more equally than others. In particular, he loves Rainbow Arrays.

An array is Rainbow if it has the following structure:

* First **a1** elements equal **1**.
* Next **a2** elements equal **2**.
* Next **a3** elements equal **3**.
* Next **a4** elements equal **4**.
* Next **a5** elements equal **5**.
* Next **a6** elements equal **6**.
* Next **a7** elements equal **7**.
* Next **a6** elements equal **6**.
* Next **a5** elements equal **5**.
* Next **a4** elements equal **4**.
* Next **a3** elements equal **3**.
* Next **a2** elements equal **2**.
* Next **a1** elements equal **1**.
* **ai** can be any non-zero positive integer.
* There are no other elements in array.

Help Chef in finding out if the given array is a Rainbow Array or not.

**Input**

* The first line of the input contains an integer **T** denoting the number of test cases.
* The first line of each test case contains an integer **N**, denoting the number of elements in the given array.
* The second line contains **N** space-separated integers **A1**, **A2**, ..., **AN** denoting the elements of array.

**Output**

* For each test case, output a line containing "yes" or "no" (without quotes) corresponding to the case if the array is rainbow array or not.

**Constraints**

* 1 ≤ **T** ≤ 100
* 7 ≤ **N** ≤ 100
* 1 ≤ **Ai** ≤ 10

**Subtasks**

* **Subtask 1** (100 points) : Original constraints

**Example**

**Input**

3

19

1 2 3 4 4 5 6 6 6 7 6 6 6 5 4 4 3 2 1

14

1 2 3 4 5 6 7 6 5 4 3 2 1 1

13

1 2 3 4 5 6 8 6 5 4 3 2 1

**Output**

yes

no

no

**Explanation**

The first example satisfies all the conditions.

The second example has **1** element of value **1** at the beginning and **2** elements of value **1** at the end.

The third one has no elements with value **7** after elements with value **6**.

<https://www.codechef.com/AUG17/problems/RAINBOWA>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Collections;

namespace ConsoleApplication1

{

class Program

{

static bool rainbow(int[] arr)

{

int indice7 = Array.IndexOf(arr,7);

if (indice7 < 0) return false;

Dictionary<int, int> antes = new Dictionary<int, int>();

Dictionary<int, int> despues = new Dictionary<int, int>();

for (int i = 1; i <= 7; i++)

{

antes[i] = 0;

despues[i] = 0;

}

for (int i = 0; i < indice7; i++)

{

if (!antes.ContainsKey(arr[i]))

{

return false;

}

else

{

antes[arr[i]]++;

}

}

int last7 = Array.LastIndexOf(arr,7);

for (int i = last7 + 1; i < arr.Length; i++)

{

if (!despues.ContainsKey(arr[i]))

{

return false;

}

else

{

despues[arr[i]]++;

}

}

for (int i = 1; i <= 6; i++)

{

if (antes[i] == 0) return false;

if (despues[i] == 0) return false;

if (antes[i] != despues[i])

{

return false;

}

}

return true;

}

static void Main(string[] args)

{

// int[] arr = Array.ConvertAll("1 2 3 4 4 5 6 6 6 7 6 6 6 5 4 4 3 2 1".Trim().Split(' '), e => int.Parse(e));

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

while (t-- > 0)

{

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

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

Console.WriteLine(rainbow(arr) ? "yes" : "no");

}

Console.ReadLine();

}

}

}

---------solucion por un chino-----------

<http://www.vkadoo.cn/4671DF09B5FCE12AFAB4A40C4D739E50.AHtml>

#include<cstdio>

#include<cstring>

#include<algorithm>

using namespace std;

const int inf=0x3f3f3f3f;

int read(){

int ans=0,f=1,c=getchar();

while(c<'0'||c>'9'){if(c=='-') f=-1; c=getchar();}

while(c>='0'&&c<='9'){ans=ans\*10+(c-'0'); c=getchar();}

return ans\*f;

}

int T,n,s[257],c[257];

int main()

{

T=read();

while(T--){

bool f=false;

memset(c,0,sizeof(c));

n=read();

for(int i=1;i<=n;i++){

s[i]=read();

if(s[i]>7||s[i]<1) f=1;

else c[s[i]]=1;

}

for(int i=1;i<=7;i++) if(!c[i]) f=1;

if(f){printf("no\n"); continue;}

int cnt=(n+1)/2;

s[0]=-inf; s[n+1]=-inf;

for(int i=1,j=n;i<=cnt;i++,j--)

if(s[i]!=s[j]||s[i]<s[i-1]||s[j]<s[j+1]){

printf("no\n");

f=1;

break;

}

if(!f) printf("yes\n");

}

return 0;

}