https://course.acciojob.com/idle?question=864309e9-49cf-4bd9-b2a5-1fa5989e3cf9

- EASY
- Max Score: 30 Points
-
-
-
-

Find Hinged Element

Given an array of length N, find an element before which all elements are smaller than it, and after which all are greater than it. Return the index of the element if there is such an element, otherwise, return -1.

NOTE:

- If the right most element is the largest element in the array, it is the hinged element.
- First element can never be hinged element.

Complete the given function.

Input Format:

The first line contains integer N

The second line contains N space separated integers of arr[i].

Output Format:

Print the required index (0-based).

Example 1

Input:

```
9
5 1 4 3 6 8 10 7 9
```

Output:

```
4
```

Explanation:

All elements on the left of arr[4] are smaller than it and all elements on right are greater.

Example 2

Input:

```
4
5 1 4 4
```

Output:

```
-1
```

Explanation:

No such index exists.

Constraints:

0 <= nums.length <= 100000

0 <= nums[i] <= 10000

Topic Tags

- **Basics**
- **Arrays**

My code

```
// in java

import java.util.*;
import java.lang.*;
import java.io.*;

public class Main
{
    public static void main (String[] args) throws java.lang.Exception
    {
        //your code here
        Scanner s=new Scanner(System.in);
        int n=s.nextInt();
        int arr[]=new int[n];
        int arrb[]=new int[n];
        for(int i=0;i<n;i++)
            arr[i]=s.nextInt();

        for(int i=0;i<n;i++)
            arrb[i]=-1;

        int max=arr[0];
        arrb[0]=arr[0];
          for(int i=1;i<n;i++)
              {
                  if(arr[i]>max)
                  {
                      arrb[i]=arr[i];
```

```
                                          max=arr[i];
                                   }
                           }

                int min=arr[n-1];
                  for(int i=n-1;i>=0;i--)
                           {
                                   if(arr[i]>min)
                                   {
                                           arrb[i]=-1;

                                   }
                                   else
                                   {
                                             min=arr[i];
                                   }
                           }
             int f=1;
                     for(int i=0;i<n;i++)
                               if(arrb[i]!=-1)
                               {
                                       System.out.print(i+" ");
                                //return;
                                       f=0;
                              }
                if(f==1)
         System.out.print("-1");
        }

}
*/
```

```
import java.util.*;
import java.lang.*;
import java.io.*;

public class Main
{
    public static void main (String[] args) throws java.lang.Exception
    {
        Scanner sc  = new Scanner (System.in);
        int n = sc.nextInt();
        int [] arr= new int [n];
        for (int i=0;i<n;i++){
            arr[i] = sc.nextInt();
        }
        System.out.print(isHinged(arr,n));

    }
    public static int isHinged(int[] arr,int n){
        if(n==0 || n==1)
            return 0;

        int[] prefix = new int[n];
        prefix[0]=arr[0];
        for(int i=1;i<n;i++){
            prefix[i]=Math.max(arr[i],prefix[i-1]);
        }

        int[] sufix = new int[n];
        sufix[n-1]=arr[n-1];
        for(int i=n-2;i>=0;i--){
            sufix[i]=Math.min(arr[i],sufix[i+1]);
```

```
                }

                int ans = -1;
                for(int i=0;i<n;i++){
                        if(i==0)
                        {
                                if(arr[i]<sufix[i+1])
                                        ans = i;
                        }
                        else if(i==n-1)
                        {
                                if(arr[i]>prefix[i-1])
                                        ans = i;
                        }
                          else if(prefix[i-1]<arr[i] && arr[i]<sufix[i+1])
                                  ans=i;



                        }

                return ans;
        }
}
```