https://course.acciojob.com/idle?question=3f172f2c-d823-4c47-93c0-30c9dccfde82

- EASY
- Max Score: 30 Points

Local Maximum.

Given an array 'A' with 'N' positive integers. The array follows a special property:

For some i with 0 < i < N - 1 :

A[0] < A[1] < ... < A[i - 1] < A[i] and A[i] > A[i + 1] > ... > A[N - 1].

Your task is to find the index i in 0-based indexing.

Note: There always exists such an index.

Input Format

The first line contains an integer 'N' denoting the number of elements.

The second line contains N-space-separated integers denoting the elements of the array 'A'.

Output Format

Print a single integer in a new line, denoting the required index.

Example 1

Input:

```
5
0 3 5 2 1
```

Output:

```
2
```

Explanation:

Since A[0] < A[1] < A[2] and A[2] > A[3] > A[4] ,the answer is 2.

Example 2

Input:

```
4
1 3 1 0
```

Output:

```
1
```

Explanation:

Since A[0] < A[1] and A[1] > A[2] > A[3] ,the answer is 1.

Constraints

3 <= N <= 10^4

0 <= A[i] <= 10^6

Topic Tags

- **Binary Search**
- **Arrays**

My code

```
// in java
import java.util.*;
```

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



//test case issuse




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];
		 for(int i=0;i<n;i++)
			arr[i]=s.nextInt();


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

```
                }
            System.out.println(ans);
           /* Scanner s=new Scanner(System.in);

                    int n=s.nextInt();
                    int arr[]=new int[n];
                    for(int i=0;i<n;i++)
                        arr[i]=s.nextInt();
                    int lp=0,rp=n-1,ans=-1;
                    while(lp<=rp)
                        {
                            int mid=(lp+rp)/2;
                        if( mid>0 && mid <n && arr[mid] >arr[mid-1]
&& arr[mid] >arr[mid+1]   )
                                ans=mid;

                         if( arr[mid] >arr[mid+1])
                            rp=mid-1;
                      else  lp=mid+1;
                }
            System.out.println(ans);*/

      }
}
```