782

085

W1823C5E085 K11823C5E085 K11825C5E085 K11825C5E085 K11825C5E085 K11825C5E085 K11825



## STUDENT REPORT

**DETAILS** 

Name

MOHAMMED FURKHAN

**Roll Number** 

KUB23CSE085

**Title** 

PEAK ELEMENT, FINDER

Description: You are given an N- dimensional array arr[]. A peak element in the array is defined as an element whose value is greater than or equal to its neighboring elements (if they exist). Your task is to find the index of any peak element in the given array

Note: use 0-based indexing

Input:

An integer representing the number of elements in the array. N space-separated integers, denoting the elements of the array.

N space-separated integers ,denoting the elements of the array arr[]

Sample Input:

5

1 3 20 4 1

**Sample Output:** 

KU823C5E085 KU823C5E085 KU82

2

LOR (1823) CSE085 KUB23C

11ed3c7 WB23C5E085 KUB23C5E085 KUB25C5E085 KUB25C5E085 KUB25C5E085 KUB25C5E085 KUB25C5E085 KUB25C5 https://practice.reinprep.com/student/get-report/6e3c2d8e-7d68-11ef-ae9a-0e411ed3c76b

KUB23C5E085 KUB23C5E085
KUB23C5E085
KUB23C5E085

KNB23C5E085 KNB23C5E085 KNB23C5E085

KUR23CSE085 KUR23CSE085 KUR23CSE085 KUR23CSE085

TIB.

```
def find_peak(arr):
      n = len(arr)
      # Check the first element
      if n == 1 or arr[0] >= arr[1]:
          return 0
     # Check the last element
      if arr[n - 1] >= arr[n - 2]:
          return n - 1
     # Check for peak in the middle elements
     for i in range(1, n - 1):
          if arr[i] >= arr[i - 1] and arr[i] >= arr[i + 1]:
              return i
      return -1 # Return -1 if no peak is found, though the problem guarantees a peak
 # Example usage
 if __name__ == "__main__":
     n = int(input())
      arr = list(map(int, input().split()))
      result = find_peak(arr)
      print(result)
5 / 5 Test Cases Passed | 100 %
```

https://practice.reinprep.com/student/get-report/6e3c2d8e-7d68-11ef-ae9a-0e411ed3c76b