

GeeksforGeeks

A computer science portal for geeks

Practice

IDE

Q&A

GeeksQuiz

Leaders in an array

Write a program to print all the LEADERS in the array. An element is leader if it is greater than all the elements to its right side. And the rightmost element is always a leader. For example in the array {16, 17, 4, 3, 5, 2}, leaders are 17, 5 and 2.

Let the input array be `arr[]` and size of the array be `size`.

Method 1 (Simple)

Use two loops. The outer loop runs from 0 to `size - 1` and one by one picks all elements from left to right. The inner loop compares the picked element to all the elements to its right side. If the picked element is greater than all the elements to its right side, then the picked element is the leader.

C++

```
#include<iostream>
using namespace std;

/*C++ Function to print leaders in an array */
void printLeaders(int arr[], int size)
{
    for (int i = 0; i < size; i++)
    {
        int j;
        for (j = i+1; j < size; j++)
        {
            if (arr[i] <= arr[j])
                break;
        }
        if (j == size) // the loop didn't break
            cout << arr[i] << " ";
    }
}

/* Driver program to test above function */
int main()
{
    int arr[] = {16, 17, 4, 3, 5, 2};
    int n = sizeof(arr)/sizeof(arr[0]);
```

```
    printLeaders(arr, n);  
    return 0;  
}
```

Python

```
# Python Function to print leaders in array  
  
def printLeaders(arr,size):  
  
    for i in range(0, size):  
        for j in range(i+1, size):  
            if arr[i]<=arr[j]:  
                break  
        if j == size-1: # If loop didn't break  
            print arr[i],  
  
# Driver function  
arr=[16, 17, 4, 3, 5, 2]  
printLeaders(arr, len(arr))  
  
# This code is contributed by _Devesh Agrawal__
```

Output:

```
17 5 2
```

Time Complexity: $O(n^2)$

Method 2 (Scan from right)

Scan all the elements from right to left in array and keep track of maximum till now. When maximum changes it's value, print it.

C++

```
#include <iostream>  
using namespace std;  
  
/* C++ Function to print leaders in an array */  
void printLeaders(int arr[], int size)  
{  
    int max_from_right = arr[size-1];
```

```
/* Rightmost element is always leader */
cout << max_from_right << " ";

for (int i = size-2; i >= 0; i--)
{
    if (max_from_right < arr[i])
    {
        max_from_right = arr[i];
        cout << max_from_right << " ";
    }
}

/* Driver program to test above function*/
int main()
{
    int arr[] = {16, 17, 4, 3, 5, 2};
    int n = sizeof(arr)/sizeof(arr[0]);
    printLeaders(arr, n);
    return 0;
}
```

Python

```
# Python function to print leaders in array
def printLeaders(arr, size):

    max_from_right = arr[size-1]
    print max_from_right,
    for i in range( size-2, 0, -1):
        if max_from_right < arr[i]:
            print arr[i],
            max_from_right = arr[i]

# Driver function
arr = [16, 17, 4, 3, 5, 2]
printLeaders(arr, len(arr))

# This code contributed by _Devesh Agrawal__
```

Output

```
2 5 17
```

Time Complexity: $O(n)$

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.



52 Comments Category: [Arrays](#) Tags: [array](#)

Related Posts:

- [Longest Span with same Sum in two Binary arrays](#)
- [Count Inversions of size three in a give array](#)
- [Find the subarray with least average](#)
- [Count triplets with sum smaller than a given value](#)
- [Find zeroes to be flipped so that number of consecutive 1's is maximized](#)
- [Reorder an array according to given indexes](#)
- [Find maximum value of Sum\(i*arr\[i\]\) with only rotations on given array allowed](#)
- [Find maximum average subarray of k length](#)

([Login](#) to Rate and Mark)

2.1

Average Difficulty : 2.1/5.0
Based on 12 vote(s)

☐

Add to TODO List

☐

Mark as DONE

[Like](#) [Share](#) 2 people like this. Be the first of your friends.

Writing code in comment? Please use code.geeksforgeeks.org, generate link and share the link here.