**Leaders in an array**

[array](http://www.practice.geeksforgeeks.org/tag-page.php?tag=array&isCmp=0)

[Payu](http://www.practice.geeksforgeeks.org/tag-page.php?tag=Payu&isCmp=1)

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. The rightmost element is always a leader.

**Input:**  
The first line of input contains an integer T denoting the number of test cases. The description of T test cases follows.  
The first line of each test case contains a single integer N denoting the size of array.  
The second line contains N space-separated integers A1, A2, ..., AN denoting the elements of the array.

**Output:**  
Print all the leaders.

**Constraints:**  
1 <= T <= 100  
1 <= N <= 100  
0 <= A[i]<=100

**Example:**  
Input:  
2  
6  
16 17 4 3 5 2  
5  
1 2 3 4 0  
Output:  
17 5 2  
4 0

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=623>

#include <iostream>

#include <stdio.h>

#include <map>

#include <vector>

#include <algorithm>

#define ll long long int

using namespace std;

int main() {

int t;

scanf("%d", &t);

while(t--) {

int n;

scanf("%d", &n);

int arr[n];

for(int i =0; i<n; i++) {

scanf("%d", &arr[i]);

}

for(int i =0; i<n; i++) {

int j;

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

if(arr[j] >= arr[i]) {

break;

}

}

if(j >= n) {

cout << arr[i] << " ";

}

}

printf("\n");

}

return 0;

}

**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.

#include <iostream>

#include <stdio.h>

#include <map>

#include <vector>

#include <algorithm>

#define ll long long int

using namespace std;

int main() {

int t;

scanf("%d", &t);

while(t--) {

int n;

scanf("%d", &n);

int arr[n];

for(int i =0; i<n; i++) {

scanf("%d", &arr[i]);

}

std::vector<int> rev;

//printf("%d ", arr[n-1]);

rev.push\_back(arr[n-1]);

int max = arr[n-1];

for(int i = n-2; i >=0; i--) {

if(arr[i] > max) {

//cout << arr[i] << " ";

rev.push\_back(arr[i]);

max = arr[i];

}

}

std::reverse(rev.begin() , rev.end());

for(int i =0; i<rev.size(); i++) {

cout << rev[i] << " ";

}

printf("\n");

}

return 0;

}