**Problem: Leader of an Array**

Write a program to print all the **leaders** in the array. An element is a leader if it is strictly greater than all the elements to its right side. And the rightmost element is always a leader.

**Input Format**

- The first line contains N, the number of elements in an array.
- The second line contains N integers, denoting the elements in the array.

**Output Format**

In a single line output all the leaders in the given array.

**Sample 1:**

Input

6

16 17 4 3 5 2

Output

17 5 2

## Task 1:

Complete the following code:

```
#include <bits/stdc++.h>
using namespace std;

int main() {

    int n;
    cin>>n;
    int a[n];
    for(int i=0;i<n;i++)
    {
        cin>>a[i];
    }
    // --- your code goes here
    cout<<"\n";
}
```

**Maximum Subarray Sum**

Given an integer array nums*nums*, find the **subarray** with the largest sum, and print its sum.

**Note**: A subarray is a contiguous non-empty sequence of elements within an array.

**Input Format**

- The first line contains T, the number of test cases.
- The first line in each test case contains N, the number of elements in an array.
- The second line in each test case contains N integers, denoting the elements in the array.

**Output Format**

For each test case, output the maximum subarray sum of each array.

**Constraints**

- $1 \leq T \leq 100$
- $1 \leq N \leq 100$
- $-10^9 \leq A_i \leq 10^9$

Input

3

9

-2 1 -3 4 -1 2 1 -5 4

1

1

5

5 4 -1 7 8

Output

6

1

23

## Task 2:

**Approach:**

We will check for every possible subarray:

- Iterate over all possible starting indices of the subarray.
- For each starting index, iterate over all possible ending indices (greater than or equal to the starting index).
- Calculate the sum of elements between the starting and ending index (inclusive).
- Keep track of the maximum sum encountered during this process.

Complete the following code:

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int t;
    cin>>t;
    while(t--){
        int n;
        cin>>n;
        int arr[n];
        for(int i=0; i<n; i++){
            cin>>arr[i];
        }
        int maxsum = INT_MIN;
        //your code goes here
        cout<<maxsum<<endl;
    }
}
```