**Count increasing Subarrays**

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

Given an array of N integers A1, A2, ….AN, count number of subarrays of A which are strictly increasing.   
A subarray, A[i, j], where 1<=i<=j<=N is a sequence of integers of Ai, Ai+1, ….Aj.  
A subarray A[i, j] is strictly increasing if Ai < Ai+1 < Ai+2< …….

**Input :**  
The first line contains an integer ‘T’ denoting the total number of test cases.  
In each test case, the first line contain an integer 'N' denoting number of elements in an array.  
Second line contains ‘N’ integers A1, A2,…AN denoting the elements of the array.

**Output :**  
For each test case, output number of strictly increasing subarrays.

**Constraints:**  
1<=T<=30  
1<=N<=1000  
1<=Ai<=100000

**Example:**  
Input:  
2  
6  
1 3 3 2 3 5  
2  
1 5

Output:  
4  
1

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

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=405>

#include <iostream>

#include <stdio.h>

#include <math.h>

#include <vector>

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]);

       }

       int cont =0;

       for(int i =0; i < N - 1; i++) {

           for(int j=i+1; j<N; j++) {

                bool incrementa = true;

                for(int k =i+1; k<=j; k++) {

                    if(arr[k] <= arr[k-1]){

                        incrementa= false;

                          break;

                    }

                }

                if(incrementa) {

                    cont ++;

                }

            }

        }

       printf("%d**\n**", cont);

    }

  system("pause");

 return 0;

}