### Stock span problem

The stock span problem is a financial problem where we have a series of **n** daily price quotes for a stock and we need to calculate the span of stocks price for all **n** days.   
The span **Si** of the stocks price on a given day **i** is defined as the maximum number of consecutive days just before the given day, for which the price of the stock on the current day is less than or equal to its price on the given day.  
For example, if an array of 7 days prices is given as {100, 80, 60, 70, 60, 75, 85}, then the span values for corresponding 7 days are {1, 1, 1, 2, 1, 4, 6}.

**Example 1:**

**Input**:

N = 7, price[] = [100 80 60 70 60 75 85]

**Output**:

1 1 1 2 1 4 6

**Explanation**:

Traversing the given input span for 100

will be 1, 80 is smaller than 100 so the

span is 1, 60 is smaller than 80 so the

span is 1, 70 is greater than 60 so the

span is 2 and so on. Hence the output will

be 1 1 1 2 1 4 6.

**Example 2:**

**Input**:

N = 6, price[] = [10 4 5 90 120 80]

**Output**:

1 1 2 4 5 1

**Explanation**:

Traversing the given input span for 10

will be 1, 4 is smaller than 10 so the

span will be 1, 5 is greater than 4 so

the span will be 2 and so on. Hence, the

output will be 1 1 2 4 5 1.

**Company Tags**

[**Flipkart**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Flipkart) [**Accolite**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Accolite) [**Amazon**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Amazon) [**Microsoft**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Microsoft) [**Samsung**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Samsung) [**Adobe**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Adobe)

**Topic Tags**

[**Arrays**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Arrays) [**Stack**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Stack) [**Data Structures**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Data%20Structures)

Java code

//{ Driver Code Starts

import java.util.\*;

import java.io.\*;

import java.lang.\*;

class CodingMaxima

{

public static void main(String args[])

{

Scanner sc = new Scanner(System.in);

int t = sc.nextInt();

while(t-- > 0)

{

int n =sc.nextInt();

int a[] = new int[n];

int i = 0;

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

a[i] = sc.nextInt();

int[] s = new Solution().calculateSpan(a, n);

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

{

System.out.print(s[i] + " ");

}

System.out.println();

}

}

}

// } Driver Code Ends

class Solution

{

//Function to calculate the span of stockâ€™s price for all n days.

public static int[] calculateSpan(int price[], int n)

{

// Your code here

int[] arr=new int[n];

Stack<Integer> stack = new Stack<Integer>();

arr[0]=1;

stack.push(0);

for(int i=1;i<n;i++)

{

while(!stack.isEmpty() && price[i]>=price[stack.peek()]){

stack.pop();

}

arr[i]=stack.isEmpty()?i+1:i-stack.peek();

stack.push(i);

}

return arr;

}

}