Poisonous Plants ★

Problem   Submissions   Leaderboard   Editorial 🔒

There are a number of plants in a garden. Each of the plants has been treated with some amount of pesticide. After each day, if any plant has more pesticide than the plant on its left, being weaker than the left one, it dies.

You are given the initial values of the pesticide in each of the plants. Determine the number of days after which no plant dies, i.e. the time after which there is no plant with more pesticide content than the plant to its left.

Example

$p = [3, 6, 2, 7, 5]$ // pesticide levels

Use a 1-indexed array. On day 1, plants 2 and 4 die leaving $p' = [3, 2, 5]$. On day 2, plant 3 in p' dies leaving $p'' = [3, 2]$. There is no plant with a higher concentration of pesticide than the one to its left, so plants stop dying after day 2.

Function Description

Complete the function poisonousPlants in the editor below.

poisonousPlants has the following parameter(s):

- int p[n]: the pesticide levels in each plant

Returns

- int: the number of days until plants no longer die from pesticide

Input Format

The first line contains an integer n, the size of the array p.
The next line contains n space-separated integers $p[i]$.

Constraints

$1 \le n \le 10^5$
$0 \le p[i] \le 10^9$

Sample Input

```
7
6 5 8 4 7 10 9
```

Sample Output

```
2
```

Explanation

Initially all plants are alive.

Plants = {(6,1), (5,2), (8,3), (4,4), (7,5), (10,6), (9,7)}

Plants[k] = (i,j) => jth plant has pesticide amount = i.

After the 1st day, 4 plants remain as plants 3, 5, and 6 die.

Plants = {(6,1), (5,2), (4,4), (9,7)}

After the 2nd day, 3 plants survive as plant 7 dies.

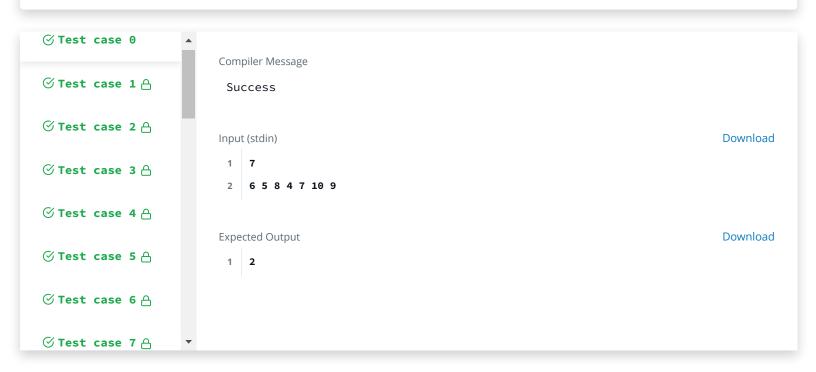Plants = {(6,1), (5,2), (4,4)}

Plants stop dying after the 2nd day.

Change Theme   Language C#

```
29            int max = 0;
30            Stack<int> stack = new Stack<int>();
31            stack.Push(0);
32
33            for (int i = 1; i < p.Count; i++)
34            {
35                if(p[i] > p[i-1])
36                {
37                    //Console.WriteLine($"{p[i]} is greater than {p[i - 1]}. So adding 1
    to the day array");
38                    days[i] = 1;
39                }
40
41                //Console.WriteLine($"check current element {p[i]} is less than min {min}
    . If less then change min to current element ");
42                min = Math.Min(min, p[i]);// min < p[i] ? min : p[i];
43                //Console.WriteLine($" While to condition is top of the stack is {p
    [stack.Peek()]} >= to the current element {p[i]}");
44                while (stack.Count > 0 && p[stack.Peek()] >= p[i])
45                {
46
47                    if(p[i] > min)
48                    {
49                        //Console.WriteLine($"If current element {p[i]} is greater than
    min - {min} then updates days array {days[i]} by getting max of " +
50                        //  $"current element at days array  {days[i]} or index at the
    stack top element plus 1 {days[stack.Peek()]+1}");
```

Line: 29 Col: 23

⬆ Upload Code as File   ☐ Test against custom input   Run Code   Submit Code

---

You have earned 70.00 points!

These points will also count towards your progress in the Problem Solving Badge.

18%   1090/2200

Problem Solving
★★★★★

Congratulations   Proceed

You solved this challenge. Would you like to challenge your friends?

Compiler Message

Success

Input (stdin)   Download

1   7
2   6 5 8 4 7 10 9

Expected Output   Download

1   2