**Supervillain Show-Offs**



SuperCity consists of a single line of *n* buildings, where each building *i* is *heighti* units tall; however, SuperCity is under attack by two supervillains looking to show off their superpowers! The supervillains are standing on opposite ends of the city, unleashing their powers in an attempt to destroy all *n* buildings. In a single move, a supervillain unleashes their power and destroys the nearest contiguous block of buildings in which the respective heights of each building are nondecreasing. In other words:

* If a supervillain is standing on the *left* end of the city and the nearest intact building is building *i*, then performing a move will destroy each consecutive building *i, i + 1, i + 2, …* satisfying *heighti ≤ heighti+1 ≤ heighti+2 ≤ …* until there are either no more buildings in their path or there is some building *j* satisfying *heightj > heightj+1*.
* If a supervillain is standing on the *right* end of the city and the nearest intact building is building *i*, then performing a move will destroy each consecutive building *i, i − 1, i − 2, …* satisfying *heighti ≤ heighti-1 ≤ heighti-2 ≤ …* until there are either no more buildings in their path or there is some building *j* satisfying *heightj > heightj-1*.

Once a supervillain destroys a building, the building's height becomes *0*.

Complete the function in the editor below. It has one parameter: an array of integers, *height*, where each *heighti* denotes the height of building *i*. The function must return an integer denoting the minimum number of total moves needed for two supervillains standing on opposite ends of the city (as described by the array of building heights) to destroy all *n* buildings.

Input Format

The first line contains an integer, *n*, denoting the number of elements in *height*.

Each line *i* of the *n* subsequent lines contains an integer describing *heighti*.

Constraints

* *1≤ n ≤ 105*
* *1 ≤ heighti ≤ 105*, where *0 ≤ i < n*.

Output Format

Return an integer denoting the minimum number of total moves needed for two supervillains on opposite ends of the array to destroy all *n* buildings.

Sample Input 0

8

1

2

3

4

8

7

6

5

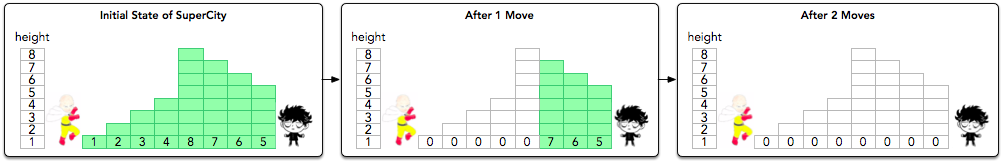
Sample Output 0

2

Explanation 0

The respective heights of each building are given as *height = [1, 2, 3, 4, 8, 7, 6, 5]*. The supervillains can perform the following minimal sequence of moves:

The diagram above depicts the changes to SuperCity's skyline after each move by a supervillain.



1. In the first move, the supervillain on the left destroys buildings *0* through *4*, because *height0 ≤ height1 ≤ height2 ≤ height3 ≤ height4*; note that the destruction stops at this point, as *height4 > height5*.
2. In the second move, the supervillain on the right destroys buildings *7* through *5*, because *height7 ≤ height6 ≤ height5*.

As it took a minimal two moves for the supervillains to level all the buildings, the function returns *2*.

Sample Input 1

6

1

2

1

2

10

9

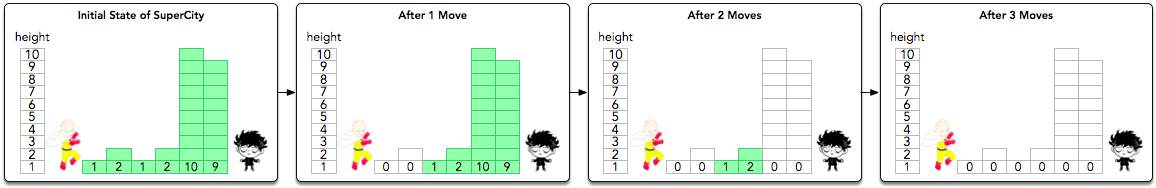
Sample Output 1

3

Explanation 1

The respective heights of each building are given as *height = [1, 2, 1, 2, 10, 9]*. The supervillains can perform the following minimal sequence of moves:

The diagram above depicts the changes to SuperCity's skyline after each move by a supervillain.



1. In the first move, the supervillain on the left destroys buildings *0* through *1*, because *height0 ≤ height1*.
2. In the second move, the supervillain on the right destroys buildings *5* through *4*, because *height5 ≤ height4*.
3. In the third move, the supervillain on the left destroys buildings *2* through *3*, because *height2 ≤ height3*.

As it took a minimal three moves for the supervillains to level all the buildings, the function returns *3*.