

CS1020 Take-home Lab #4

Exercise #1: Swinging Monkeys

(http://www.comp.nus.edu.sg/~cs1020/3_ca/labs.html)

Objective

The objective of this exercise is to test the understanding of **stack** and its application.

Task Statement

Monkeys like to swing from tree to tree. They can swing from one tree to another directly as long as there is no tree in between that is taller than or have the same height as either one of the two trees. For example, if there are 5 trees with heights 19m, 17m, 20m, 20m and 20m lining up in that order, then the monkey will be able to swing from one tree to the other as shown below:

1. from first tree to second tree
2. from first tree to third tree
3. from second tree to third tree
4. from third tree to fourth tree
5. from fourth tree to fifth tree

Tarzan, the king of jungle who is able to communicate with the monkeys, wants to test the monkeys to see if they know how to count the total number of pairs of trees that they can swing directly from one to the other. But he himself is not very good in counting. So he turns to you, the best Java programmer in the country, to write a program for getting the correct count for the trees in different parts of the jungle.

Input

The first line contains **N**, the number of trees in the path. The next line contains **N** integers $a_1 a_2 a_3 \dots a_N$, where a_i represents the height of the i -th tree in the path, $0 < a_i \leq 2^{31}$ and $2 \leq N \leq 500,000$.

Note that short symbol **N** is used above for convenience. In your program, you are expected to give it a descriptive name.

Output

The total number of pairs of trees which the monkeys can swing directly from one to the other with the given list of tree heights.

Sample Input 1

```
4
3 4 1 2
```

Sample Output 1

```
4
```

Sample Input 2

5
19 17 20 20 20

Sample Output 2

5

Sample Input 3

4
12 21 21 12

Sample Output 3

3

Sample Explanation

Sample 1:

- (3, 4): from 1st tree, it can only swing to the 2nd tree directly but not all the other tree as the 2nd tree blocks it.
- (4, 1) and (4, 2): from the 2nd tree, it can swing directly to the 3rd and 4th tree directly.
- (1, 2): It can swing from the 3rd tree to the 4th tree.

Sample 2:

- (19, 17) and (19, 20): from 1st tree, it can swing directly to 2nd and 3rd tree but not the rest.
- (17, 20): from the 2nd tree, it can only swing to the 3rd tree but not the rest.
- (20, 20): from the 3rd tree, it can only swing to the 4th tree but not the 5th.
- (20, 20): from the 4th tree, it can swing to the 5th tree.

Sample 3:

- (12, 21): from 1st tree, it can only swing to the 2nd tree but not the rest.
- (21, 21): from the 2nd tree, it can swing to the 3rd tree but not the 4th.
- (21, 12): obvious.

Notes:

- An efficient program is required. You may try some given test cases to see if your program is fast enough.
- For example, if you check all the pairs to see if there is any tree in between them that will block the swing, it might be too slow to pass some of the test cases.
- You can improve the efficiency by using a stack.