☆ Angry Animals

Pi's father, Danny, runs the Hackerville Zoo. He is moving to Rookieville and wants to take all of the zoo animals with him via ship. He is confused about how to arrange them because a few of the species cannot be kept together in the same cabin.

There are *n* animals placed in a straight line. Each animal is identified by a unique number from *1* to *n* in order. There are *m* pairs *(a[i], b[i])* which imply that animals *a[i]* and *b[i]* are enemies and should not be kept in the same cabin. Pi is good at solving problems and he came up with following challenge: count the number of different groups that do not contain any pair such that they are enemies. A group is defined as an interval (x, y) such that all animals in the range from x to y form a group. Determine the number of groups that can be formed according to the Pi's challenge.

For example, given *n = 3* animals and *m = 3* pairs of enemies, *a = [1, 2, 3]* and *b = [3, 3, 1]*, animal *1* is the enemy of animal *3*, and animal *3* is the enemy of animals *1* and *2*. Because *3* is an enemy of both *1* and *2*, it must be in its own cabin. Animals *1* and *2* can be roomed together or separately. There are four possible groupings meeting the constraints: *{1, 2}* , *{1}, {2}, {3}*. Note that the intervals are along the original line of animals numbered consecutively from *1* to *n*, i.e. *[1, 2, 3]* in this case. The animals cannot be reordered and animals cannot be skipped, e.g. *{2, 1}* and *{1, 3}* are invalid intervals.

Function Description

Complete the function *angryAnimals* in the editor below. The function must return the number of groups that can be formed according to Pi's challenge.

angryAnimals has the following parameters:

n: an integer that denotes the number of unique animals
a[a[0],...a[m-1]]: an array of integers
b[b[0],...b[m-1]]: an array of integers

Constraints

- $1 \le n \le 10^5$
- $1 \le m \le 10^6$
- $1 \le a[i], b[i] \le n$

Input Format For Custom Testing

The first line contains an integer, *n*.
The second line contains an integer, *m*, that denotes the number of elements in *a*.

Each line *i* of the *m* subsequent lines (where $0 \le i < m$) contains an integer that describes *a[i]*.
The next line again contains an integer, *m*, that denotes the number of elements in *b*.

Each line *i* of the *m* subsequent lines (where $0 \le i < m$) contains an integer that describes *b[i]*.

Sample Case 0

Sample Input For Custom Testing

```
4
2
1
2
2
3
4
```

Sample Output

```
7
```

Explanation

(1), (1,2), (2), (2,3), (3), (3,4), (4) are the groups that be formed according to Pi's challenge.

```
5
2
1
2
2
3
5
```

Sample Output

```
11
```

Explanation

(1), (1,2), (2), (2,3), (2,3,4), (3), (3,4), (3,4,5), (4), (4,5), (5) are the groups that can be formed according to Pi's challenge.

YOUR ANSWER

We recommend you take a quick tour of our editor before you proceed. The timer will pause up to 90 seconds for the tour. Start tour

View Code Diff | Java 8

```
1   import java.io.*;…
14
15  class Result {
16
17      /*
18       * Complete the 'angryAnimals' function below.
19       *
20       * The function is expected to return a LONG_INTEGER.
21       * The function accepts following parameters:
22       *  1. INTEGER n
23       *  2. INTEGER_ARRAY a
24       *  3. INTEGER_ARRAY b
25       */
26
27      public static long angryAnimals(int n, List<Integer> a, List<Integer> b) {
28      // Write your code here
29
30      }
31
32  }
33
34  public class Solution {…
```

Line: 14 Col: 1

Test against custom input

Run Code | Submit code & Continue

(You can submit any number of times)

Download sample test cases *The input/output files have Unix line endings. Do not use Notepad to edit them on windows.*