

Problem O. Journey to the Moon

OS Linux

The member states of the UN are planning to send **2** people to the moon. They want them to be from different countries. You will be given a list of pairs of astronaut ID's. Each pair is made of astronauts from the same country. Determine how many pairs of astronauts from different countries they can choose from.

Example

$n = 4$

$astronaut = [1, 2], [2, 3]$

There are **4** astronauts numbered **0** through **3**. Astronauts grouped by country are **[0]** and **[1, 2, 3]**. There are **3** pairs to choose from: **[0, 1]**, **[0, 2]** and **[0, 3]**.

Function Description

Complete the *journeyToMoon* function in the editor below.

journeyToMoon has the following parameter(s):

- *int n*: the number of astronauts
- *int astronaut[p][2]*: each element *astronaut[i]* is a **2** element array that represents the ID's of two astronauts from the same country

Returns

- *int*: the number of valid pairs

Input Format

The first line contains two integers *n* and *p*, the number of astronauts and the number of pairs.

Each of the next *p* lines contains **2** space-separated integers denoting astronaut ID's of two who share the same nationality.

Constraints

- $1 \leq n \leq 10^5$
- $1 \leq p \leq 10^4$

Input	Output
5 3 0 1 2 3 0 4	6

Explanation 0

Persons numbered $[0, 1, 4]$ belong to one country, and those numbered $[2, 3]$ belong to another. The UN has **6** ways of choosing a pair:

$[0, 2], [0, 3], [1, 2], [1, 3], [4, 2], [4, 3]$

Input	Output
4 1 0 2	5

Explanation 1

Persons numbered $[0, 2]$ belong to the same country, but persons **1** and **3** don't share countries with anyone else. The UN has **5** ways of choosing a pair:

$[0, 1], [0, 3], [1, 2], [1, 3], [2, 3]$