Flights

This question is graded for 1.5%!

Statement

Fluffy the Hamster is operating an airline. His airline operates flights in n different cities numbered from 0 to $n-1$. There are m different direct flights operated by Fluffy's airline, where each direct flight connects two different cities u_i, v_i in both directions. It is also possible to reach any city from any other city via one or more flights operated by Fluffy's airline.

The holiday season is fast approaching, and Fluffy predicts an increase in passenger volume travelling from city a to city b on his flights. To increase passenger satisfaction, Fluffy hopes to minimize the number of layovers (flight changes) required. To do so, he will add **one** additional direct flight between two cities which do not have a direct flight connecting them yet.

Help Fluffy compute the number of pairs of cities such that the minimum number of layovers will be decreased if Fluffy adds a direct flight between those two cities.

Constraints

- $2 \leq n \leq 10^3$
- $1 \leq m \leq 5 \cdot 10^3$
- $0 \leq u_i, v_i, a, b \leq n-1$
- $u_i \neq v_i$

Input

The first line of input will contain four integers n, m, a, b.

The next m lines of input will each contain two integers, u_i, v_i.

Output

Print a single integer, the number of pairs of cities such that if Fluffy adds a direct flight between the two cities, the minimum number of layovers required to get from a to b is decreased.

Examples

Sample Input	Expected Output
4 3 0 3 0 1 1 2 2 3	3

Notes

1. A skeleton file has been given to help you. You should not create a new file or rename the file provided. You should develop your program using this skeleton file.
2. You are free to define your own helper methods and classes (or remove existing ones) if it is suitable but you must put all the new classes, if any, in the same skeleton file provided.

Skeleton File

You are given the skeleton file `Flights.java`. You should see the following contents when you open the file:

```
/**
 * Name          :
 * Matric. No    :
*/

import java.util.*;

public class Flights {
    private void run() {
    }

    public static void main(String args[]) {
        Flights runner = new Flights();
        runner.run();
    }
}
```