Показан ви е граф с N върха и M ребра, свързващи по два върха. Гарантирано е, че съществува път от всеки връх до всеки друг връх, използвайки ребрата. Също е известно, че всяка двойка върхове е свързана от най-много едно ребро, както и, че няма ребра свързващи връх със себе си.

Собственикът на графа планира да премахне **точно едно** ребро от графа и затова се интересува между колко двойки върхове винаги ще съществува път, независимо кое ребро се премахне.

Напишете програма отговаряща на въпроса му.

**Вход**

От първия ред на файла **connection.in** се въвеждат N и M - броят върхове и броят ребра. Всеки от следващите M редове съдържа 2 числа а и b, описващи, че има ребро между върховете a и b.

**Изход**

На единствения ред във файла **connection.out** отпечатайте 1 число: търсения брой двойки.

**Ограничения**

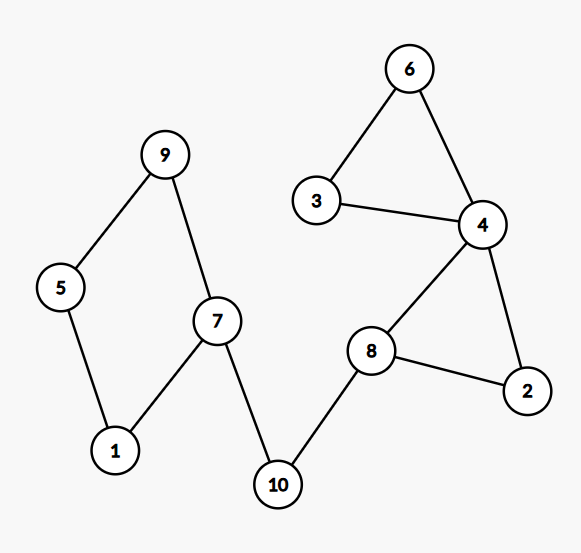
**Ограничение по време: 1 sec.**

**Ограничение по памет: 256 MB.**

**Примерен Тест**

|  |  |
| --- | --- |
| **Вход (connection.in)** | **Изход (connection.out)** |
| 10 12  10 8  4 3  2 8  4 6  5 9  5 1  1 7  10 7  2 4  6 3  4 8  7 9 | 16 |

**Обяснение на тестовия пример**

Пример за някои двойки, които не отговарят на условието:

* 4 и 9: при премахване на реброто 10-7 или 10-8 няма да има път между тях
* 10 и 7: при премахване на реброто 10-7 няма да има път между тях.

Пример за някои двойки, които изпълняват условието:

* 3 и 4
* 7 и 9