Graf

Дан граф, состоящий из N вершин, пронумерованных от 1 до N, где некоторые пары вершин соединены ребрами. Изначально граф содержит M ребер, каждое из которых представлено парой (a_i, b_i), указывающей на то, что должно существовать ребро, соединяющее вершину a_i и вершину b_i.

Будем говорить, что граф обладает структурной стабильностью, если для любых трех вершин $a, b, c,$ таких что $a < b < c$, если существует ребро между вершиной a и вершиной b, и другое ребро между вершиной a и вершиной c, тогда также должно существовать ребро между вершиной b и вершиной c.

Задача. Разработайте программу, которая, имея исходный граф, определяет минимальное количество ребер, которое необходимо добавить, чтобы обеспечить структурную стабильность графа.

Входные данные. Первая строка стандартного ввода содержит два целых числа N и M, разделенных пробелом. Следующие M строк содержат по два целых числа, разделенных пробелом. Строка i содержит a_i и b_i ($1 \le a_i < b_i \le n$). Все M пар (a_i, b_i) различны.

Выходные данные. Стандартный вывод должен содержать одно целое число, представляющее минимальное общее количество ребер после добавления ребер для достижения структурной стабильности графа.

Ограничения. $1 \le N, M \le 10^5$. Ограничения по времени выполнения и объему используемой памяти даны в общем описании задач, предложенных для решения. Исходный файл должен иметь название `graf.pas, graf.c` или `graf.cpp`.

Пример 1.

Вход

```
6 4
1 2
1 4
4 6
4 5
```

Выход

```
6
```

Пояснение: У нас есть 4 ребра. Добавляем ребро (2, 4), потому что у нас есть ребро (1, 2) и (1, 4). Добавляем ребро (5, 6), потому что у нас есть ребро (4, 6) и (4, 5). Таким образом, мы получаем 6 ребер.

Пример 2.

Вход

```
7 6
2 3
2 6
2 7
1 3
1 4
1 5
```

Выход

```
16
```

Оценка: Тесты будут организованы следующим образом:

- Для 17% тестов, $N \le 100$.
- Для других 25% тестов, $N \le 5000$.
- Для других 18% тестов, для всех $1 \le j \le N$, существует не более одной пары (ai, bi), такой что bi = j.
- Оставшиеся 30% тестов без дополнительных ограничений.