

[Đồ thị]. Bài 37. Quần đảo chìm

Time limit: 1.0s **Memory limit:** 256M

Thành phố **Vice** được xây dựng trên một nhóm các hòn đảo, với những cây cầu nối giữa chúng.

Như bất kỳ ai ở **Vice City** đều biết, nỗi sợ hãi lớn nhất của các phó thành phố là một ngày nào đó các hòn đảo sẽ bị nhấn chìm.

Vấn đề lớn của việc này là một khi các hòn đảo bị nhấn chìm, một số hòn đảo khác có thể bị ngắt kết nối.

Bạn đã được thị trưởng thành phố **Vice** thuê để nói cho ông ta biết có bao nhiêu hòn đảo, khi bị nhấn chìm, sẽ ngắt kết nối các bộ phận của thành phố Vice.

Bạn nên biết rằng ban đầu tất cả các hòn đảo của thành phố đều được kết nối với nhau.

Đầu vào

Dòng đầu tiên là **N** và **M**, tương ứng với số lượng đảo và cầu

M dòng tiếp theo của mỗi test case mô tả cầu nối giữa 2 hòn đảo **u, v**

Giới hạn

$$1 \leq T \leq 100$$

$$1 \leq N \leq 1000$$

$$1 \leq M \leq N \cdot (N - 1) / 2$$

$$1 \leq u, v \leq N$$

Đầu ra

In ra số lượng hòn đảo khi bị ngập sẽ làm thành phố **Vice** bị ngắt kết nối.

Ví dụ :

Input 01

```
6 8
1 3
6 1
6 3
4 1
6 4
5 2
3 2
3 5
```

Output 01

```
1
```

[Đồ thị]. Bài 41. Sửa đường

Time limit: 1.0s **Memory limit:** 256M

Ở **28techland** có **n** thành phố và **m** đường giữa chúng. Thật không may, tình trạng của những con đường quá kém nên chúng không thể được sử dụng. Nhiệm vụ của bạn là sửa chữa một số con đường để có một con đường tốt giữa hai thành phố bất kỳ.

Đối với mỗi con đường, bạn biết chi phí sửa chữa của nó và bạn nên tìm giải pháp sao cho tổng chi phí càng nhỏ càng tốt.

Đầu vào

Dòng đầu tiên có hai số nguyên **n** và **m**: số thành phố và đường. Các thành phố được đánh số $1, 2, \dots, n$.

Sau đó, có **m** dòng mô tả các con đường. Mỗi dòng có ba số nguyên **a**, **b** và **c**: có một con đường giữa các thành phố **a** và **b**, và chi phí sửa chữa của nó là **c**. Tất cả các con đường đều là đường hai chiều.

Mọi con đường đều nằm giữa hai thành phố khác nhau và có nhiều nhất một con đường giữa hai thành phố.

Giới hạn

$$1 \leq n \leq 10^5$$

$$1 \leq m \leq 2 \cdot 10^5$$

$$1 \leq a, b \leq n$$

$$1 \leq c \leq 10^9$$

Đầu ra

In một số nguyên: tổng chi phí sửa chữa tối thiểu. Nếu không có giải pháp nào, hãy in "**IMPOSSIBLE**".

Ví dụ :

Input 01

```
5 6
1 2 3
2 3 5
2 4 2
3 4 8
5 1 7
5 4 4
```

Output 01

```
14
```

[Đồ thị]. Bài 49. Neverland

Time limit: 1.0s **Memory limit:** 256M

Nhân dịp nghỉ lễ ngày 2/9, **Tèo** muốn đi du lịch tại nước **Neverland**, từ thành phố Hồ Chí Minh muốn tới **Neverland** có tất cả **M** chuyến bay 1 chiều giữa các thành phố, các thành phố **Tèo** có thể đi qua được đánh số từ 1 tới **N**, trong đó thành phố Hồ Chí Minh được đánh số là 1 và **Neverland** được đánh số là **N**.

Tèo có một phiếu giảm giá ngày Quốc khánh có thể áp dụng để giảm 50% giá vé cho 1 chuyến bay duy nhất trong tất cả các chuyến bay trong hành trình tới **Neverland**. Nếu giá vé chuyến bay là số lẻ thì bạn sẽ được làm tròn xuống. Bạn hãy giúp **Tèo** sử dụng phiếu giảm giá này một cách hợp lý để có thể tối thiểu chi phí bay của **Tèo**. Dữ liệu đảm bảo **Tèo** luôn tìm được đường đi từ thành phố HCM tới **Neverland**.

Đầu vào

- Dòng 1 bắt đầu bởi hai số nguyên **N** và **M**.
 - **N** dòng tiếp theo, mỗi dòng gồm thông tin **x**, **y**, **z** trong đó **z** là chi phí bay từ thành phố **x** tới thành phố **y**, đây là chuyến bay 1 chiều.
-

Giới hạn

- $2 \leq N \leq 10^5$
 - $1 \leq M \leq 2 \cdot 10^5$
 - $1 \leq x, y \leq N$
 - $1 \leq z \leq 2 \cdot 10^9$
-

Đầu ra

In ra kết quả của bài toán

Ví dụ :

Input 01

```
10 12
1 2 4
5 7 1
4 6 4
7 9 3
9 10 1
7 8 5
1 3 1
6 7 4
4 5 1
3 4 2
8 10 5
2 4 3
```

Output 01

```
7
```

[Đồ thị]. Bài 52. Xây dựng thư viện

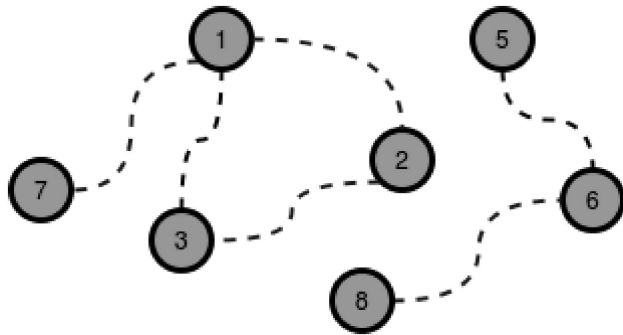
Time limit: 1.0s **Memory limit:** 256M

Tại **28Land** có **N** thành phố được đánh số từ 1 tới **N**, **28Tech** muốn tất cả các cư dân trong thành phố đều có thể đọc sách. Ban đầu **N** thành phố này chưa có đường đi (có thể di chuyển 2 chiều) vì đã bị hỏng hết, tuy nhiên **28Tech** có thể sửa lại bất kỳ đường đi nào giữa 2 thành phố với chi phí đều nhau là **cost**.

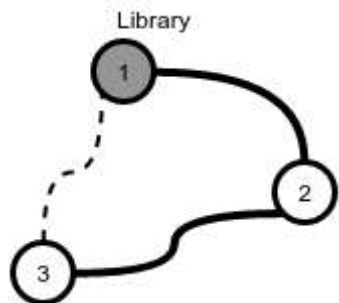
Bây giờ cần phải xây dựng các thư viện để mọi người có thể đọc sách, chi phí xây dựng 1 thư viện tại 1 thành phố bất kỳ là **build**. Các thành phố mà giữa chúng có đường đi có thể sử dụng chung 1 thư viện. Hay nói cách khác các cụm thành phố (2 thành phố bất kỳ trong 1 cụm luôn có đường đi tới nhau) sẽ sử dụng chung 1 thư viện được đặt tại 1 thành phố nào đó trong cụm để tiết kiệm chi phí.

Bây giờ bạn hãy giúp **28Tech** tìm ra chi phí nhỏ nhất để xây dựng các thư viện và sửa các con đường giữa các thành phố nhé.

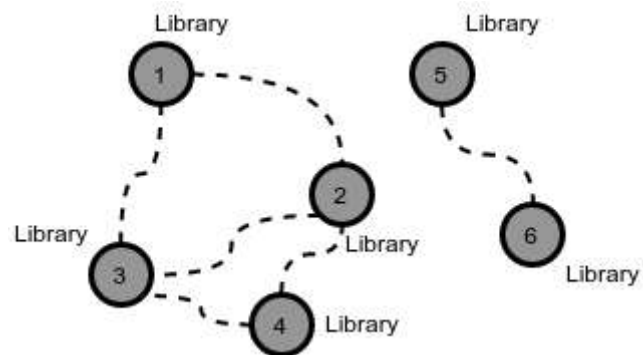
Ví dụ 1 : Sơ đồ thành phố được cho dưới đây mô tả có tất cả 7 thành phố, với các con đường có thể được sửa lần lượt là (1, 2) , (1, 3), (1, 7), (2, 3), (5, 6), (6, 8) (ví dụ này mình bỏ qua thành phố thứ 4 ko xét nhưng trong test case sẽ đủ mọi thành phố từ 1 tới N). Chi phí $\text{cost} = 2$, $\text{build} = 3$, ta có thể sửa 3 con đường (1, 2), (1, 7), (2, 3) để kết nối 4 thành phố (1, 2, 3, 7), chi phí sẽ là $3 * 2 = 6$, cộng thêm xây dựng 1 thư viện tại thành phố 1 mất 3 nữa. Cụm thành phố thứ 2 sẽ sửa 2 con đường (5, 6) và (6, 8) chi phí sẽ là $2 * 2$, cộng thêm xây dựng 1 thư viện tại thành phố 5 mất 3 nữa. Vậy chi phí nhỏ nhất trong ví dụ này là : 16



Ví dụ 2 : $\text{build} = 2$, $\text{cost} = 1$, chi phí nhỏ nhất = 4



Ví dụ 3 : build = 2, cost = 7, chi phí nhỏ nhất là 12



Đầu vào

- Dòng 1 bắt đầu bởi số nguyên **T** số bộ test
- Mỗi bộ test gồm dòng đầu tiên là 4 số : **N, M, build, cost**.
- **M** dòng tiếp theo của bộ test mô tả **M** con đường có thể sửa chữa giữa 2 thành phố.

Giới hạn

- $1 \leq T \leq 100$
- $1 \leq N \leq 10^5$

- $0 \leq \mathbf{M} \leq n * (n - 1) / 2$
 - $1 \leq \mathbf{cost, build} \leq 10^5$
-

Đầu ra

- In ra chi phí nhỏ nhất cho mỗi test case
-

Ví dụ :

Input 01

```
5
9 2 91 84
8 2
2 9
5 9 92 23
2 1
5 3
5 1
3 4
3 1
5 4
4 1
5 2
4 2
8 3 10 55
6 4
3 2
7 1
1 0 5 3
2 0 102 1
```

Output 01

```
805
184
80
5
204
```

[Đồ thị]. Bài 53. Đường lên sao hỏa

Time limit: 1.0s **Memory limit:** 256M

Yilong Ma muốn chọn người để đưa lên sao hỏa, mỗi tàu vũ trụ **Falcon X** chỉ mang được 2 người nhưng phải mang 2 người ở 2 quốc gia khác nhau. Có **N** người tham gia đăng ký lên sao hỏa free, được đánh số thứ tự từ 0 tới **N - 1**, có **M** tấm thẻ, trong tấm thẻ này sẽ chứa 2 con số mô tả số thứ tự của 2 người mà thuộc về cùng một quốc gia.

Dựa vào **M** tấm thẻ này, bạn hãy giúp **Yilong Ma** tính xem có bao nhiêu cách lựa chọn 2 người lên sao hỏa free, lưu ý rằng bạn không được chọn 2 người ở cùng 1 quốc gia nhé.

Đầu vào

- Dòng 1 bắt đầu bởi số 2 nguyên **N** và **M**
 - M** dòng tiếp theo mô tả thông tin của **M** tấm thẻ.
-

Giới hạn

- $1 \leq N \leq 10^5$
 - $1 \leq M \leq 10^5$
-

Đầu ra

- In ra đáp án của bài toán
-

Ví dụ :

Input 01

```
6 3
0 1
1 2
3 4
```

Output 01

```
11
```

Giải thích test :

3 người (0, 1, 2), (3, 4), (5) thuộc cùng 1 quốc gia khi đó bạn có thể tạo ra 11 cặp : (5, 0), (5, 1), (5, 2), (5, 3), (5, 4), (0, 3), (0, 4), (1, 3), (1, 4), (2, 3), (2, 4).