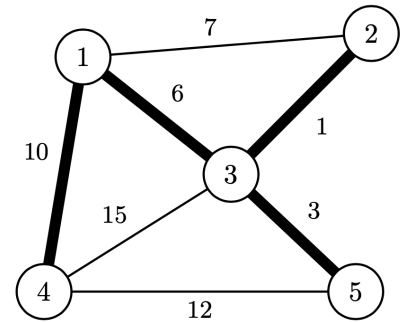


Ejercicio 2. Grafos y estructuras de partición (3.5 puntos)

Los residentes de Barro City son demasiado tacaños para pavimentar las calles de la ciudad; después de todo, a nadie le gusta pagar impuestos. Sin embargo, tras varios meses de lluvias intensas empiezan a estar cansados de enfangarse los pies cada vez que salen a la calle.

Debido a su gran tacañería, en vez de pavimentar todas las calles de la ciudad, quieren pavimentar solamente las suficientes para poder ir de una intersección a otra cualquiera de la ciudad siguiendo una ruta pavimentada y, además, quieren gastarse tan poco dinero como sea posible en la realización de esta obra. Y es que a los residentes de Barro City no les importa caminar mucho, si ello les permite ahorrar algún dinero. El alcalde tiene interés en saber cuál es el mínimo presupuesto que tiene que reservar de las arcas públicas para pavimentar la ciudad.

Por ejemplo, en una ciudad como la del dibujo con 5 intersecciones y 7 calles, donde el número que aparece al lado de cada calle representa lo que costaría pavimentarla, convendría pavimentar las calles que aparecen más gruesas.



Entrada

La entrada está compuesta por diversos casos de prueba, ocupando cada uno de ellos varias líneas.

En la primera línea aparece el número N (entre 1 y 10.000) de intersecciones en la ciudad, y en la segunda línea aparece el número C (entre 0 y 100.000) de calles (entre intersecciones). A continuación, aparece una línea por cada calle con tres enteros, que indican los números de las intersecciones que une la calle (números entre 1 y N) y lo que costaría pavimentarla (un valor entre 1 y 100.000). Nunca hay una calle que conecte una intersección consigo misma, ni dos calles que conecten el mismo par de intersecciones.

Salida

Para cada caso de prueba se escribirá una línea con el coste mínimo necesario para pavimentar la ciudad de tal forma que se pueda viajar de cualquier intersección a cualquier otra por calles pavimentadas. Si tal pavimentación no es posible, se escribirá **Imposible**.

Entrada de ejemplo

```
5
7
1 2 7
1 3 6
3 2 1
1 4 10
3 4 15
3 5 3
4 5 12
4
3
1 3 2
1 4 3
3 4 5
```

Salida de ejemplo

```
20
Imposible
```