**04. The Secret Network**

**Условие:**

Шпионска организация поддържа тайна комуникационна мрежа между своите агенти, разпределени в различни градове. Всеки агент може да изпраща съобщения до друг агент само ако между техните градове съществува пряка връзка. Всяка връзка има различна цена за поддържане, а някои връзки са толкова скъпи, че ръководството на организацията иска да ги избегне.

Вашата задача е да помогнете на организацията да създаде мрежа, която свързва всички агенти с минимални разходи за поддържане на връзките. Важно е да се избере оптималната комбинация от връзки, така че всеки агент да бъде свързан с всеки друг (директно или индиректно), но с минимална обща цена.

**Вход:**

1. На първия ред се задават две цели числа **N** и **M** (1 ≤ **N** ≤ 1000, 0 ≤ **M** ≤ 10000), представляващи броя на градовете и броя на връзките между тях.
2. Следват **M** реда, всеки от които съдържа три цели числа **A**, **B** и **C**, които означават, че има връзка между градовете **A** и **B** с цена **C**.
3. Градовете са номерирани от 1 до **N**.

**Изход:**

Изведете едно цяло число, което представлява минималната обща цена за свързване на всички агенти. Ако няма начин всички агенти да бъдат свързани, изведете **-1**.

**Допълнителни условия:**

* Всяка връзка е двупосочна, т.е. ако има връзка между градовете **A** и **B**, тя може да се използва в двете посоки.
* Ако мрежата вече е свързана, не се изисква добавяне на нови връзки, а само оптимизация на цената.
* Някои агенти може да са в изолирани градове, което означава, че няма връзки към техния град.

**Пример:**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 4 5  1 2 3  1 3 4  4 2 2  3 4 5  1 4 10 | 9 |

**Обяснение:**

Най-евтината мрежа за свързване на всички агенти е:

* Връзка между градове 1 и 2 с цена 3.
* Връзка между градове 4 и 2 с цена 2.
* Връзка между градове 1 и 3 с цена 4.  
  Общата цена е 9.

**Указания за решаване:**

* Представете мрежата като неориентиран граф с тегла на ръбовете.
* Използвайте **алгоритъма на Крускал** или **алгоритъма на Прим** за намиране на минималното покриващо дърво (MST) на графа.
* Поддържайте структура за обединяване на компоненти (Union-Find) за бързо обединение и проверка на свързаност.
* Ако след изпълнението на алгоритъма не всички градове са свързани, изведете **-1**.