

Roman Patrollers

In ancient times, patrollers were used to ensure that all the cities of the Roman Empire were under control. A patroller's job consisted in continuously visiting the cities of the empire, trying to minimise the interval between two visits to each city. The Military Society (MS) wants to simulate the behavior of one such patroller to see how effective they were.

Each cycle of the simulation corresponds to one time unit. The instantaneous city idleness (ICI) for a city X after T cycles of the simulation is the number of cycles elapsed since the last visit of the patroller to the city X (i.e. the number of time units the city X remained unvisited). All of the cities have initial instantaneous city idleness equal to zero at the start of the simulation. The instantaneous empire idleness (IEI) after each given cycle is the sum of the instantaneous city idleness of all cities after that given cycle. Finally, the empire idleness (EI) for an N -cycle simulation is the sum of the instantaneous empire idleness after each of the N cycles of simulation.

After visiting a city X , the patroller always chooses to visit the neighbour city Y with the highest instantaneous city idleness (if more than one city has the highest idleness, the one with the lowest identifier is chosen). Cities X and Y are neighbour if there is a road linking the two cities directly, without going through any intermediate city. In the beginning of the simulation, the patroller is located in one of the cities, and is given a map of the Roman Empire containing a description of all the roads in the empire, indicating the length (in kilometers) and which two cities each road connects. A road between cities X and Y can be used both to go from X to Y and from Y to X .

Assuming that a patroller travels one kilometer in one time unit (one simulation cycle) and that the time to visit a city is negligible (equal to zero), MS asks you to determine the empire idleness after an N -cycle simulation.

For clarity, consider the example of an empire which contains 3 cities (1, 2 and 3) and two roads of length 1 km. The first road connects cities 1 and 2, while the second road connects cities 2 and 3. Below you find a trace of a 3-cycle simulation for such a scenario, considering that the patroller starts at city 1.

Start of the simulation

Patroller at: 1

$ICI_1 = 0, ICI_2 = 0, ICI_3 = 0$

$IEI = 0$

$EI = 0$

After cycle 1

Patroller at: 2

$ICI_1 = 1, ICI_2 = 0, ICI_3 = 1$

$IEI = 2$

$EI = 2$

After cycle 2

Patroller at: 1
ICI1 = 0, ICI2 = 1, ICI3 = 2
IEI = 3
EI = 5

After cycle 3 Patroller at: 2
ICI1 = 1, ICI2 = 0, ICI3 = 3
IEI = 4
EI = 9

Therefore, for such a scenario, after 3 simulation cycles the empire idleness is 9.

Input

The input consists of several test cases. The first line of a test case contains four integers $C, R, N,$ and S , indicating respectively the quantity of cities in the empire ($2 \cdot C \cdot 1000$), the number of roads ($1 \cdot R \cdot C(C - 1)/2$), the number of cycles to be simulated ($1 \cdot N \cdot 1000$) and the identifier of the starting city of the patroller ($1 \cdot S \cdot C$). Each city is identified by a distinct integer from 1 to C . Each of the following R lines contains three integers X, Y and D describing a road; X and Y represent cities ($1 \cdot X \leq Y \cdot C$) and D represents the distance ($1 \cdot D \cdot 1000$), in kilometers, of the road that connects X and Y directly, without passing through any other city. Each pair of cities X and Y will appear at most once in a road description. You can assume that it is always possible to travel from any city to any other city in the empire using the roads available. The end of input is indicated by $C = R = N = S = 0$.

Output

For each test case in the input, your program must produce one line containing the empire idleness after the N -cycle simulation.

Example

Input:

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

Output:

```
2
4
8
10
9
```