Msc in Data Science

Foundations of computer science

Instructor: Georgios Zois

Programming Exercise - Submission date: 29/09/2022, 11:59EET,
Examination date: 30/09/2022

Racing and refueling

A speed race is taking place among N different cities which are connected via M bidirectional roads. In fact, each road i connects 2 different cities and has length l_i kilometers. The racing route is fixed and passes through K different cities, $C_1, C_2, \ldots, C_{K-1}, C_K$ in this certain order (assuming that there exists a road between any two consecutive cities of the above ones). Each driver must strictly follow the above route and should refuel his vehicle at least L times during the race. The refueling can be done in each of the L different cities among the cities $C_2, \ldots, C_{K-1}$ (the refueling at the starting and the finishing cities is meaningless).

For the refueling operation, gas stations are located at B different cities, let $S_1, \ldots, S_B$, and at each one there are partners for every driver ready to move towards him and refuel his vehicle. More specifically, in order to refuel his vehicle, a driver should stop at some city $C_i, 2 \leq i \leq K-1$, along the racing route and then, his partners start from the closest gas station and move towards the city C_i for the refueling. Thus, the driver has to wait at city C_i a time equal to the required time that his partners need to reach him, starting from the closest gas station (assuming that the refueling time is zero). After the refueling operation, the driver continues his route to the next city C_{i+1}.

The Greek team asks you to write them a program in order to compute the minimum time needed to complete the race. For your computations assume that the speed of all vehicles (both driver's and his partners') is 1 kilometer per second, partners' vehicles can use any road they want (even the route roads), and partners always start from the closest gas station (which can be the same for two or more refueling points).

Input data: Your program will read from standard input, the first line, 5 natural numbers that correspond to the number N of the cities, the number M of roads, the number K of the cities of the racing route, the number L of required refueling, and the number B of gas stations. The cities are numbered from 1 to N. At the i-th of the next M lines there will be 3 natural numbers u_i, v_i, l_i which specify that the i-th road connects the cities u_i and v_i and has length l_i. In the next K lines there will be K different natural numbers $C_1, C_2, \ldots, C_{K-1}, C_K$ that specify the cities of the racing route. Finally, in the next B lines there will be B natural numbers, $S_1, \ldots, S_B$ that specify the cities where the gas stations are located.

Output data: Your program should type, in the standard output, an integer which corresponds to the minimum time, in seconds, that the Greek team needs to complete the race. Note that for large values of N and M, the minimum time (and also some intermediate results needed for its computation) may be greater than 2^{32}.

Restrictions:

$3 \leq N \leq 60000$
$3 \leq M \leq 10^6$
$1 \leq u_i, v_i, C_i, S_i \leq N$
$3 \leq K \leq N$
$1 \leq B \leq N$
$1 \leq L \leq K - 2$
$1 \leq l_i \leq 20000$

Input Example:

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

Output Example:

```
19
```