Algorithm Design Laboratory with Applications

Prof. Stefano Leucci

Problem: *Metro.*

The city of Algoland has two distinct subway systems, built from different companies at different depths. Both companies have subway stations in the same n locations of the city, indexed from 0 to $n-1$.

When these subway system were constructed, both companies knew whether a tunnel connecting any two locations a and b could be built, and what its length $\ell(a,b) \in \mathbb{N}$ in kilometers would have been.

Nevertheless, since the two networks were built at different depths and with different construction techniques, the estimated cost of building such a tunnel was $c_1(a,b) \in \mathbb{N}$ for the first company and $c_2(a,b) \in \mathbb{N}$ for the second company. Each company built its network by digging the cheapest set of tunnels that allows to connect all stations. For both companies such a set of tunnels turned out to be unique.

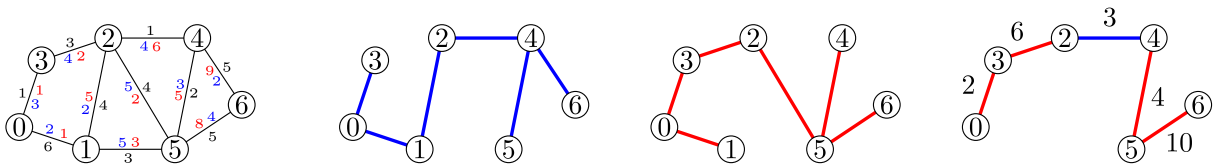
The trains of the first company take t_1 seconds to travel a distance of 1 Km, while the trains of the second company take t_2 seconds to travel the same distance.

You are in location 1 and need to reach location n as quickly as possible. Design an algorithm that, given t_1, t_2 and all values $\ell(a,b)$, $c_1(a,b)$, $c_2(a,b)$, computes the shortest possible time to complete your trip. You can assume that waiting times and transfer times are negligible.

Input. The input consists of a set of instances, or *test-cases*, of the previous problem. The first line contains the number T of test-cases. The first line of each test-case contains n, t_1, t_2, and the number of m of potential tunnels that were considered by the two companies when the subway networks were built. Each of the next m lines describes one potential tunnel between a and b, and contains the five integers a, b, $\ell(a,b)$, $c_1(a,b)$, and $c_2(a,b)$.

Output. The output consists of T lines. The i-th line is the answer to the i-th test-case and contains a single number that represents the number of seconds needed to travel from location 0 to location $n-1$ using any number of trains from the two subway networks.

Assumptions. $1 \le T \le 10$; $1 \le n \le 2^{14}$; $1 \le m \le 2^{17}$; $\max\{t_1, t_2\} \le 2^8$.
The maximum cost of a tunnel is at most 2^7. The maximum length of a tunnel is less than 2^9.



(a) (b) (c) (d)

Figure 1: (a) An example of all the potential tunnels that could have been built by the two companies. Tunnel lengths are in black while tunnel costs for the first and second company are in blue and red, respectively. (b) The subway network built by the first company. (c) The subway network built by the second company. (d) The quickest route from location 0 to location $n-1$ when $t_1 = 3$ and $t_2 = 2$. Black numbers refer to travel times.

Example.

Input (corresponding to Figure 1):

```
1
7 3 2 10
0 1 6 2 1
0 3 1 3 1
1 2 4 2 5
1 5 3 5 3
2 3 3 4 2
2 4 1 4 6
2 5 4 5 2
4 5 2 3 5
4 6 5 2 9
5 6 5 4 8
```

Output:

```
25
```

Requirements. Your algorithm should require time $O(m \log n)$ (with reasonable hidden constants), where m is the number potential tunnels that could have been built by the two companies.

Notes. A reasonable implementation should not require more than 1 second for each input file.