

Прототип 27 задания сложный уровень

Бузин Игорь Тимурович vk.com/buz_igor

Условие:

Лера приехала в Санкт-Петербург к другу, в этом городе есть N домов и какое-то количество дорог. В Питере есть две компании которые предоставляют услуги такси. Первая компания имеет в своем распоряжении M дорог и за проезд по i -ой дороге требует C_{1i} рублей. Вторая компания имеет в своем распоряжении K дорог и за проезд по j -ой дороге требует C_{2j} рублей. Лера стоит у дома под номером S и хочет добраться к другу на такси до дома под номером F . По ходу движения от дома S к дому F Лера может не однократно остановиться у какого-то иного дома и бесплатно пересесть на такси другой компании. Количество бесплатных пересадок с такси одной компании на такси другой не ограничено. В ответе укажите минимальную стоимость, за которую Лера может добраться из пункта F в пункт S .

Входные данные:

Даны два входных файла (файл А и файл В), в них все числа целые. Каждый файл имеет следующий формат:

Первая строка содержит число N ($1 \leq N \leq 100$) – количество домов, число M ($1 \leq M \leq 10000$) – количество дорог между домами в распоряжении первой компании и число K ($1 \leq K \leq 10000$) – количество дорог между домами в распоряжении второй компании. Вторая строка содержит два числа S ($1 \leq S \leq N$) и F ($1 \leq F \leq N$), дом, у которого стоит Лера, и дом, до которого нужно добраться Лере, соответственно.

Следующие M строк содержат информацию о дорогах между домами, которые в распоряжении первой компании вида:

$U_1 V_1 C_{11}$

....

$U_i V_i C_{1i}$

....

$$U_M V_M C_{1M}$$

Где каждая пара чисел обозначает что таксист первой компании может проехать по двухсторонней дороге между домами $U_i (1 \leq U_i \leq N)$ и $V_i (1 \leq V_i \leq N)$ и за это потребует $C_{1i} (1 \leq C_{1i} \leq 10000)$ рубль.

Следующие K строк содержат информацию о дорогах между домами, которые в распоряжении второй компании вида:

$$U_1 V_1 C_{21}$$

....

$$U_i V_i C_{2j}$$

....

$$U_K V_K C_{2K}$$

Где каждая пара чисел обозначает что таксист второй компании может проехать по двухсторонней дороге между домами $U_j (1 \leq U_j \leq N)$ и $V_j (1 \leq V_j \leq N)$ и за это потребует $C_{2i} (1 \leq C_{2i} \leq 10000)$ рублей.

Гарантируется что путь между S и F существует, но не гарантируется, что от какого-то дома можно дойти до любого другого.

Пример входного файла:

8 5 6

1 8

1 2 10

2 6 2

3 7 5

8 4 1

5 6 3

1 2 9

2 3 4

5 8 3

6 8 1

6 7 2

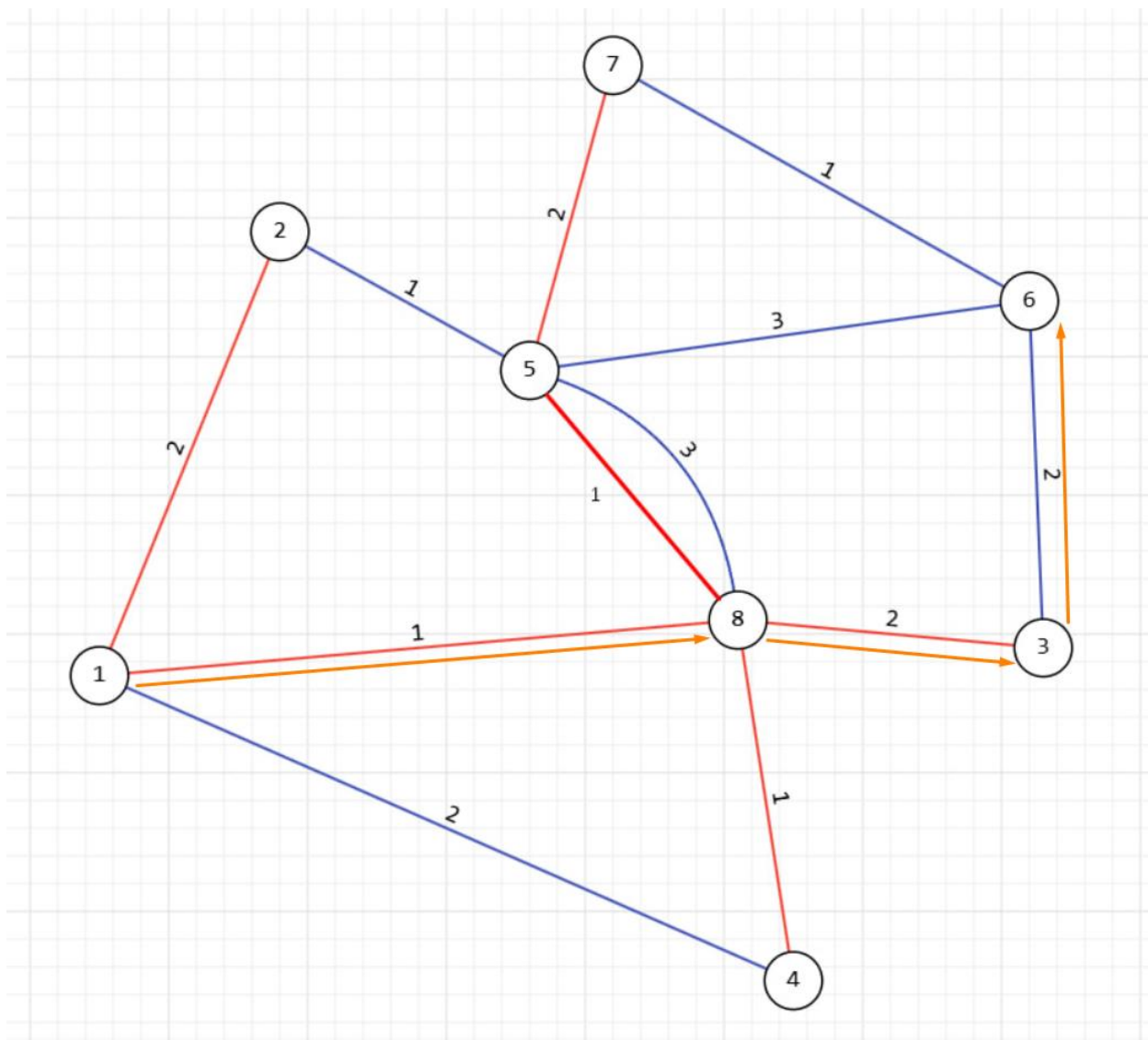
7 4 1

При таких исходных данных Лера стоит у первого дома и может вызвать такси до дома под номером два как от первой, так и от второй компании. Но так как доехать на такси второй компании стоит дешевле – она поедет на нем и заплатит 9 рублей. У дома под номером два она пересекает на такси первой компании и доедет до дома под номером шесть за 2 рубля. Сделав еще одну пересадку, она доедет до дома 8 за 1 рубль. Минимальная стоимость, за которую Лера может добраться до нужного места – 12 рублей.

В ответе укажите искомую величину для файла А, затем искомое значение для файла В.

Решение задачи:

Для файла А ученик может нарисовать граф и понять какой будет ответ в этом пункте, в данном случае ответ 5.



Для файла В ученику предлагается догадаться, что если пересадка от одного таксиста к другому осуществляется бесплатно, то можно объединить граф дорог первой и второй компании и найти самый дешевый путь от S к F с использованием алгоритма Флойда. Ответ 224.

```
#include <bits/stdc++.h>

using namespace std;
using ll = long long;

constexpr ll INF = 1e18;

void solve() {
    ll n, m, k; cin >> n >> m >> k;
    ll s, f; cin >> s >> f;
    vector<vector<ll>>>d(n, vector<ll>(n, INF));
    for (int i = 0; i < m + k; i++) {
        ll a, b, c; cin >> a >> b >> c; a--; b--;
        d[a][b] = min(d[a][b], c);
        d[b][a] = d[a][b];
    }
    for (int to = 0; to < n; to++) {
        for (int u = 0; u < n; u++) {
            for (int v = 0; v < n; v++) {
                d[u][v] = min(d[u][v], d[u][to] + d[to][v]);
            }
        }
    }
    cout << d[--s][--f];
}

signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    freopen("B.txt", "r", stdin);
    solve();
    return 0;
}
```

Пояснение по поводу сложности задачи:

Эта задача сложного уровня так как помимо поиска кратчайшего пути в графе с помощью алгоритма Флойда нужно еще придумать идею, чтобы понять, как его применить.