

## Problem D

### Discrete Speed

Consider car trips in a country where there is no friction. Cars in this country do not have engines. Once a car started to move at a speed, it keeps moving at the same speed. There are acceleration devices on some points on the road, where a car can increase or decrease its speed by 1. It can also keep its speed there. Your job in this problem is to write a program which determines the route with the shortest time to travel from a starting city to a goal city.

There are several cities in the country, and a road network connecting them. Each city has an acceleration device. As mentioned above, if a car arrives at a city at a speed  $v$ , it leaves the city at one of  $v - 1$ ,  $v$ , or  $v + 1$ . The first road leaving the starting city must be run at the speed 1. Similarly, the last road arriving at the goal city must be run at the speed 1.

The starting city and the goal city are given. The problem is to find the best route which leads to the goal city going through several cities on the road network. When the car arrives at a city, it cannot immediately go back the road it used to reach the city. No U-turns are allowed. Except this constraint, one can choose any route on the road network. It is allowed to visit the same city or use the same road multiple times. The starting city and the goal city may be visited during the trip.

For each road on the network, its distance and speed limit are given. A car must run a road at a speed less than or equal to its speed limit. The time needed to run a road is the distance divided by the speed. The time needed within cities including that for acceleration or deceleration should be ignored.

### Input

The input consists of multiple datasets, each in the following format.

```
 $n$   $m$ 
 $s$   $g$ 
 $x_1$   $y_1$   $d_1$   $c_1$ 
...
 $x_m$   $y_m$   $d_m$   $c_m$ 
```

Every input item in a dataset is a non-negative integer. Input items in the same line are separated by a space.

The first line gives the size of the road network.  $n$  is the number of cities in the network. You can assume that the number of cities is between 2 and 30, inclusive.  $m$  is the number of roads between cities, which may be zero.

The second line gives the trip.  $s$  is the city index of the starting city.  $g$  is the city index of the goal city.  $s$  is not equal to  $g$ . You can assume that all city indices in a dataset (including the above two) are between 1 and  $n$ , inclusive.

The following  $m$  lines give the details of roads between cities. The  $i$ -th road connects two cities with city indices  $x_i$  and  $y_i$ , and has a distance  $d_i$  ( $1 \leq i \leq m$ ). You can assume that the distance is between 1 and 100, inclusive. The speed limit of the road is specified by  $c_i$ . You can assume that the speed limit is between 1 and 30, inclusive.

No two roads connect the same pair of cities. A road never connects a city with itself. Each road can be traveled in both directions.

The last dataset is followed by a line containing two zeros (separated by a space).

## Output

For each dataset in the input, one line should be output as specified below. An output line should not contain extra characters such as spaces.

If one can travel from the starting city to the goal city, the time needed for the best route (a route with the shortest time) should be printed.

**You need to output 5 (please use `%.5lf`, or other function with same effect) of digits after the decimal point.**

If it is impossible to reach the goal city, the string "unreachable" should be printed. Note that all the letters of "unreachable" are in lowercase.

## Sample Input

```
2 0
1 2
5 4
1 5
1 2 1 1
2 3 2 2
3 4 2 2
4 5 1 1
6 6
1 6
1 2 2 1
2 3 2 1
3 6 2 1
1 4 2 30
4 5 3 30
5 6 2 30
```

```
6 7
1 6
1 2 1 30
2 3 1 30
3 1 1 30
3 4 100 30
4 5 1 30
5 6 1 30
6 4 1 30
0 0
```

### Output for the Sample Input

```
unreachable
4.00000
5.50000
11.25664
```