ACM International Collegiate Programming Contest
Japan Domestic, 2005-07-01

Problem D
Traveling by Stagecoach

Once upon a time, there was a traveler.

He plans to travel using stagecoaches (horse wagons). His starting point and destination are fixed, but he cannot determine his route. Your job in this problem is to write a program which determines the route for him.

There are several cities in the country, and a road network connecting them. If there is a road between two cities, one can travel by a stagecoach from one of them to the other. A coach ticket is needed for a coach ride. The number of horses is specified in each of the tickets. Of course, with more horses, the coach runs faster.

At the starting point, the traveler has a number of coach tickets. By considering these tickets and the information on the road network, you should find the best possible route that takes him to the destination in the shortest time. The usage of coach tickets should be taken into account.

The following conditions are assumed.

- A coach ride takes the traveler from one city to another directly connected by a road. In other words, on each arrival to a city, he must change the coach.
- Only one ticket can be used for a coach ride between two cities directly connected by a road.
- Each ticket can be used only once.
- The time needed for a coach ride is the distance between two cities divided by the number of horses.
- The time needed for the coach change should be ignored.

Input

The input consists of multiple datasets, each in the following format. The last dataset is followed by a line containing five zeros (separated by a space).

$n\ m\ p\ a\ b$
$t_1\ t_2\ \ldots\ t_n$
$x_1\ y_1\ z_1$
$x_2\ y_2\ z_2$
...
$x_p\ y_p\ z_p$

Every input item in a dataset is a non-negative integer. If a line contains two or more input items, they are separated by a space.

n is the number of coach tickets. You can assume that the number of tickets is between 1 and 8. m is the number of cities in the network. You can assume that the number of cities is between 2 and 30. p is the number of roads between cities, which may be zero.

a is the city index of the starting city. *b* is the city index of the destination city. *a* is not equal to *b*. You can assume that all city indices in a dataset (including the above two) are between 1 and *m*.

The second line of a dataset gives the details of coach tickets. t_i is the number of horses specified in the *i*-th coach ticket (1<=*i*<=*n*). You can assume that the number of horses is between 1 and 10.

The following *p* 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 z_i (1<=*i*<=*p*). You can assume that the distance is between 1 and 100.

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.

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 the traveler can reach the destination, the time needed for the best route (a route with the shortest time) should be printed. The answer should not have an error greater than 0.001. You may output any number of digits after the decimal point, provided that the above accuracy condition is satisfied.

If the traveler cannot reach the destination, the string "`Impossible`" should be printed. One cannot reach the destination either when there are no routes leading to the destination, or when the number of tickets is not sufficient. Note that the first letter of "`Impossible`" is in uppercase, while the other letters are in lowercase.

Sample Input

```
3 4 3 1 4
3 1 2
1 2 10
2 3 30
3 4 20
2 4 4 2 1
3 1
2 3 3
1 3 3
4 1 2
4 2 5
2 4 3 4 1
5 5
1 2 10
2 3 10
3 4 10
1 2 0 1 2
1
8 5 10 1 5
2 7 1 8 4 5 6 3
1 2 5
2 3 4
3 4 7
4 5 3
1 3 25
2 4 23
3 5 22
1 4 45
```

```
2 5 51
1 5 99
0 0 0 0 0
```

Output for the Sample Input

```
30.000
3.667
Impossible
Impossible
2.856
```

Since the number of digits after the decimal point is not specified, the above result is not the only solution. For example, the following result is also acceptable.

```
30.0
3.66667
Impossible
Impossible
2.85595
```

First Input Data

Here is your first input.

The ACM ICPC