Sightseeing trip

There is a travel agency in Adelton town on Zanzibar island. It has decided to offer its clients, besides many other attractions, sightseeing the town. To earn as much as possible from this attraction, the agency has accepted a shrewd decision: it is necessary to find the shortest route which begins and ends at the same place. Your task is to write a program which finds such a route.

In the town there are N crossing points numbered from 1 to N and M two-way roads numbered from 1 to M. Two crossing points can be connected by multiple roads, but no road connects a crossing point with itself. Each sightseeing route is a sequence of road numbers $y_1, \ldots, y_k$, $k > 2$. The road y_i ($1 \leq i \leq k-1$) connects crossing points x_i and x_{i+1}, the road y_k connects crossing points x_k and x_1. All the numbers $x_1, \ldots, x_k$ should be different. The length of the sightseeing route is the sum of the lengths of all roads on the sightseeing route, i.e. $L(y_1) + L(y_2) + \ldots + L(y_k)$ where $L(y_i)$ is the length of the road y_i ($1 \leq i \leq k$). Your program has to find such a sightseeing route, the length of which is minimal, or to specify that it is not possible, because there is no sightseeing route in the town.

Input: The first line of input file `TRIP.IN` contains two positive integers: the number of crossing points $N \leq 100$ and the number of roads $M \leq 10\,000$. Each of the next M lines describes one road. It contains 3 positive integers: the number of its first crossing point, the number of the second one, and the length of the road (a positive integer less than 500).

Output: There is only one line in output file `TRIP.OUT`. It contains either a string '`No solution.`' in case there isn't any sightseeing route, or it contains the numbers of all crossing points on the shortest sightseeing route in the order how to pass them (i.e. the numbers x_1 to x_k from our definition of a sightseeing route), separated by single spaces. If there are multiple sightseeing routes of the minimal length, you can output any one of them.

See the next page for examples...

Example 1:

`TRIP.IN`

```
5 7
1 4 1
1 3 300
3 1 10
1 2 16
2 3 100
2 5 15
5 3 20
```

`TRIP.OUT` (one of correct answers)

```
1 3 5 2
```

Example 2:

`TRIP.IN`

```
4 3
1 2 10
1 3 20
1 4 30
```

`TRIP.OUT` (the only correct answer)

```
No solution.
```