

E. The Classic Problem

time limit per test: 5 seconds
memory limit per test: 768 megabytes
input: standard input
output: standard output

You are given a weighted undirected graph on n vertices and m edges. Find the shortest path from vertex s to vertex t or else state that such path doesn't exist.

Input

The first line of the input contains two space-separated integers — n and m ($1 \leq n \leq 10^5$; $0 \leq m \leq 10^5$).

Next m lines contain the description of the graph edges. The i -th line contains three space-separated integers — u_i, v_i, x_i ($1 \leq u_i, v_i \leq n$; $0 \leq x_i \leq 10^5$). **That means that vertices with numbers u_i and v_i are connected by edge of length 2^{x_i} (2 to the power of x_i).**

The last line contains two space-separated integers — the numbers of vertices s and t .

The vertices are numbered from 1 to n . The graph contains no multiple edges and self-loops.

Output

In the first line print the remainder after dividing the length of the shortest path by 1000000007 ($10^9 + 7$) if the path exists, and -1 if the path doesn't exist.

If the path exists print in the second line integer k — the number of vertices in the shortest path from vertex s to vertex t ; in the third line print k space-separated integers — the vertices of the shortest path in the visiting order. The first vertex should be vertex s , the last vertex should be vertex t . If there are multiple shortest paths, print any of them.

Examples

input	Copy
4 4 1 4 2 1 2 0 2 3 0 3 4 0 1 4	
output	Copy
3 4 1 2 3 4	
input	Copy
4 3 1 2 4 2 3 5 3 4 6 1 4	
output	Copy

```
112
4
1 2 3 4
```

input

Copy

```
4 2
1 2 0
3 4 1
1 4
```

output

Copy

```
-1
```

Note

A path from vertex s to vertex t is a sequence v_0, \dots, v_k , such that $v_0 = s$, $v_k = t$, and for any i from 0 to $k - 1$ vertices v_i and v_{i+1} are connected by an edge.

The length of the path is the sum of weights of edges between v_i and v_{i+1} for all i from 0 to $k - 1$.

The shortest path from s to t is the path which length is minimum among all possible paths from s to t .