Problem A.   Dijkstra

Input filename:	`dijkstra.in`
Output filename:	`dijkstra.out`
Time limit:	2 seconds
Memory limit:	256 Mb

You are given a directed weighted graph. Find the shortest path from one vertex to another.

Input file format

First line of the input file contains three numbers: the number $1 \leq n \leq 2000$ of vertices, and the indices $1 \leq s, d \leq n$ of the source and destination vertices respectively.

The following n lines contain the adjacency matrix of a graph, with -1 denoting the absence of an edge, and non-negative numbers denoting the distance between the pair of vertices. It is guaranteed that the main diagonal contains all zeros.

Output file format

Output the requested distance, or -1 if there is no path from source to destination.

Sample tests

dijkstra.in	dijkstra.out
3 1 2 0 -1 2 3 0 -1 -1 4 0	6

Problem B. Distance Between Vertices

Input filename:	distance.in
Output filename:	distance.out
Time limit:	2 seconds
Memory limit:	256 Mb

You are given an undirected weighted graph. Find the path of the minimal weight from one vertex to another.

Input file format

First line of the input file contains four positive integers: the numbers $1 \leq n \leq 70000$ of vertices and $1 \leq m \leq 200000$ edges, as well as the indices $1 \leq s \neq d \leq n$ of the source and destination vertices respectively.

The following m lines contain the triples of positive integers $1 \leq u_i, v_i \leq n$ and $0 \leq w_i \leq 100000$, denoting the indices of the endpoints of an undirected edge, together with the weight of this edge.

Output file format

First string must contain one positive integer — the requested weight. The second string must contain the path itself.

If there is no path from source to destination output a single line containing -1.

Sample tests

distance.in	distance.out
4 4 1 3 1 2 1 3 4 5 3 2 2 4 1 4	3 1 2 3

Problem C.   Shortest Path

Input filename:	`path.in`
Output filename:	`path.out`
Time limit:	2 seconds
Memory limit:	256 Mb

You are given a directed weighted graph and a vertex s. Find the distances of the shortest paths from s to all other vertices.

Input file format

First line of the input file contains three positive integers: the numbers $1 \leq n \leq 2000$ of vertices and $1 \leq m \leq 5000$ edges, as well as the index $1 \leq s \leq n$ of the source vertex.

The following m lines contain the triples of positive integers $1 \leq b_i, e_i \leq n$ and $-10^{15} \leq w_i \leq 10^{15}$, denoting the indices of beginning and the end vertices of an edge, together with the weight of the edge.

Output file format

Output n lines, with the distance from s to the corresponding vertex. Output * if there is no path, and - if there is no shortest path.

Sample tests

path.in	path.out
6 7 1 1 2 10 2 3 5 1 3 100 3 5 7 5 4 10 4 3 -18 6 1 -1	0 10 - - - *