A. Shortest Path I

Given an undirected weighted graph find shortest path of each node from the node 0.

Input
First Line: N (O<N<=100000). number of nodes.
Second line: M (0<N<=300000), number of edges.
Next M lines, each: U V W(0<=U, V<N, 1<=W<=10^6), defines an edge between U and V with weight W.

Output
Shortest path of each node. If any node is unreachable, print -1. See sample for clarification.

Sample

Input	Output
7 6 1 3 2 1 2 5 0 1 3 3 2 1 3 4 50 2 5 10	0: 0 1: 3 2: 6 3: 5 4: 55 5: 16 6: -1

B. Shortest Path II

Given an undirected weighted graph print the shortest path of node 1 from the node 0. If there are multiple shortest paths, print the lexicographically smallest one.

Input
First line: N (O<N<=100000), number of nodes.
Second line: M (0<N<=300000), number of edges.
Next M lines, each: U V W(0<=U, V<N, 1<=W<=10^6), defines an edge between U and V with weight W.

Output
Shortest path of node 1. If node is not reachable, print “NOT REACHABLE”. See sample for clarification.

Sample

Input	Output
7 6 4 3 2 4 2 5 0 4 3 3 2 1 3 6 50 2 1 10	0 4 2 1

C. Shortest Path III

Given an directed weighted graph find its minimum spanning tree. The graph is guaranteed to be connected.

Input
First line: N (0<N<=150), number of nodes.
Second line: M (0<N<=5000), number of edges.
Next M lines, each: U V W(0<=U, V<N, -1000<=W<=1000), defines an edge from U to V with weight W.
Next line: Q (1<=Q<=1000), number of queries.
Next Q lines: X Y (0<=X, Y<N), a query containing a pair of vertices.

Output
For each query, find the shortest distance between the pair of vertices. In case of no path or -Infinity, print IMPOSSIBLE. See sample for clarification.

Sample

Input	Output
4 3 0 1 2 1 2 2 3 3 1 4 0 2 1 2 3 0 3 3	4 2 Impossible 0
2 1 0 1 100 q 0 1 1 0	100 Impossible