You are given a directed graph $G(V, E)$ with a set of vertices and edges. Each edge (i, j) that connects some vertex i to vertex j has an integer cost associated with that edge.

Define the operation $Halum(v, d)$ to operate on a vertex v using an integer d as follows: subtract d from the cost of all edges that enter v and add d to the cost of every edge that leaves v.

As an example of that operation, consider graph G that has three vertices named (1, 2, 3) and two edges. Edge (1, 2) has cost -1, and edge (2,3) has cost 1. The operation $Halum(2, -3)$ operates on edges entering and leaving vertex 2. Thus, edge (1, 2) gets cost -1-(-3)=2 and the edge (2, 3) gets cost 1 + (-3) = -2.

Your goal is to apply the Halum function to a graph, potentially repeatedly, until every edge in the graph has at least a certain cost that is greater than zero. You have to maximize this cost.

Input

Two space-separated integers per case: V ($V \leq 500$) and E ($E \leq 2700$). E lines follow. Each line represents a directed edge using three space-separated integers (u, v, d). Absolute value of cost can be at most 10000.

Output

If the problem is solvable, then print the maximum possible value. If there is no such solution print '`No Solution`'. If the value can be arbitrary large print '`Infinite`'

Sample Input

```
2 1
1 2 10
2 1
1 2 -10
3 3
1 2 4
2 3 2
3 1 5
4 5
2 3 4
4 2 5
3 4 2
3 1 0
1 2 -1
```

Sample Output

```
Infinite
Infinite
3
1
```