Amazon Transport Delivery Agent

Good news, You have a chance to help **Amazon**. Applications are required to help agent delivery goods between the cities. You need to answer the queries by building network of road sites for Amazon Transport Agent.

Given N cities from 0 to N-1 and distance between any two cities as d, if Agent wants to go from city A to another city C via city B as early as possible, where time is directly proportional to distance.

What you have to do is to print the total distance and path to be covered by cities if it exists , otherwise print **"No Path Found."**. If a **road connects city A to B with distance d** than it means it will also **connects city B to city A with same distance d**.

Input

1. First line contains two integers N and k , where N is the number of cities 1 to N and k is the number of road lines which connects cities.
2. Following k lines each contain three numbers A , B and D which represents city A connects city B and distance between them is D.
3. From the next line, queries start. Queries can be (Graph, DirectedPaths and ViaPaths).
    a. If the given query is **“Graph”**, print the Graph object (Adjacency list representation).
    b. If the given query is **“DirectedPaths”**, print the distance from the given source to destination. (Source and Destination are two integers as input.)
    c. If the given query is **“ViaPaths”**, print the distance from source to destination by passing through the via. In the next line print the path from source to destination(including source and destination). (Source, Via and Destination are three integers as input.)

Output

If there exists no paths as described in problem print **"No Path Found."** otherwise in first line print the total distance and in second line path to be covered .

Note : While building the graph, parallel edges are allowed. But, self loops are not allowed.

Graph building test cases: (2 marks)

Input000.txt
Input001.txt

Shortest path between Source and Destination test cases: (2 marks)

Input002.txt
Input003.txt

Shortest path between Source, Via and Destination test cases: (2 marks)

Input004.txt
Input005.txt