Implement Undirected Graph API.
AdjacencyList implements the graph API using the adjacency-lists representation. AdjacencyMatrix uses the same API using the adjacency-matrix representation.

Input Format :

1. First line of input contains String “List” or “Matrix”, which represents Graph using Adjacency List or Adjacency Matrix.
2. Second line of input contains number of Vertices.
3. Third line of input contains number of Edges.
4. Fourth line contains the key names which are separated by comma.
5. From fifth line onwards, each line contains two vertices (Integers) separated by space, that indicates the keys are to be connected. (The number of edges indicates the number of lines.)

Output Format:
For Adjacency List Representation :

1. First line contains count of vertices, count of Edges.
2. From second line, print the key and its adjacent keys separated by colon. Add space to separate adjacent cities.

For Adjacency Matrix Representation :

1. First line contains count of vertices, count of Edges.
2. From the next line, Print the V * V binary matrix, where it contains 0’s and 1’s. If there is a path, print 1 otherwise 0.

Sample Input:

```
List
13
13
Andaman and Nicobar,Andhra Pradesh,Arunachal Pradesh,Assam,Bihar,Chandigarh,Chhattisgarh,Dadra and Nagar Haveli,Daman and Diu,Delhi,Goa,Gujarat,Haryana
0 5
4 3
0 1
9 12
6 4
5 4
0 2
11 12
9 10
0 6
7 8
9 11
5 3
```

Sample Output for List Representation:

```
13 vertices, 13 edges
Andaman and Nicobar: Chhattisgarh Arunachal Pradesh Andhra Pradesh Chandigarh
Andhra Pradesh: Andaman and Nicobar
Arunachal Pradesh: Andaman and Nicobar
Assam: Chandigarh Bihar
Bihar: Chandigarh Chhattisgarh Assam
Chandigarh: Assam Bihar Andaman and Nicobar
Chhattisgarh: Andaman and Nicobar Bihar
Dadra and Nagar Haveli: Daman and Diu
Daman and Diu: Dadra and Nagar Haveli
Delhi: Gujarat Goa Haryana
Goa: Delhi
Gujarat: Delhi Haryana
Haryana: Gujarat Delhi
```

Sample Output for Matrix Representation:

13 vertices, 13 edges
0 1 1 0 0 1 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 0 0 0 0 0 0 0
0 0 0 1 0 1 1 0 0 0 0 0 0
1 0 0 1 1 0 0 0 0 0 0 0 0
1 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 1
0 0 0 0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 1
0 0 0 0 0 0 0 0 0 1 0 1 0

Note:

1. While adding edges in a graph, don't add self loop and parallel edges.
2. While printing the output, Use Bag to store adjacent vertices, to print them in output order.
3. For case of Adjacency List, print the keys and it adjacency keys.
4. For case of Adjacency Matrix, print the V*V binary matrix, which contains 0 or 1. If there is an edge print 1 else 0.
5. For a Undirected Graph, if edges are 0, print "No edges".