Next input line will obtain source and destination vertex number.
Last input line will obtain value k.

Output Format:
Output will be the weigth of shortest path from source to destination having exactly k edges.
If no path is available then print "**no path of length k is available**".

Sample I/O Problem III:

Input:	**Output:**
4 0 10 3 2 0 0 0 7 0 0 0 6 0 0 0 0 1 4 2	Weight of shortest path from (1,4) with 2 edges : 9

Week 8:

Note: Input, output format along with sample input output for problem I and II is same and is provided at the end of problem II.

I. Assume that a project of road construction to connect some cities is given to your friend. Map of these cities and roads which will connect them (after construction) is provided to him in the form of a graph. Certain amount of rupees is associated with construction of each road. Your friend has to calculate the minimum budget required for this project. The budget should be designed in such a way that the cost of connecting the cities should be minimum and number of roads required to connect all the cities should be minimum (if there are N cities then only N-1 roads need to be constructed). He asks you for help. Now, you have to help your friend by designing an algorithm which will find minimum cost required to connect these cities. (use Prim's algorithm)

II. Implement the previous problem using Kruskal's algorithm.

Input Format:
The first line of input takes number of vertices in the graph.
Input will be the graph in the form of adjacency matrix or adjacency list.

Output Format:
Output will be minimum spanning weight

Sample I/O Problem I and II:

Input:	**Output:**
7 0 0 7 5 0 0 0 0 0 8 5 0 0 0 7 8 0 9 7 0 0 5 0 9 0 15 6 0 0 5 7 15 0 8 9 0 0 0 6 8 0 11 0 0 0 0 9 11 0	Minimum Spanning Weight: 39

III. Assume that same road construction project is given to another person. The amount he will earn from this project is directly proportional to the budget of the project. This person is greedy, so he decided to maximize the budget by constructing those roads who have highest construction cost. Design an algorithm and implement it using a program to find the maximum budget required for the project.

Input Format:
The first line of input takes number of vertices in the graph.
Input will be the graph in the form of adjacency matrix or adjacency list.

Output Format:
Out will be maximum spanning weight.

Sample I/O Problem III:

Input:	**Output:**
7 0 0 7 5 0 0 0 0 0 8 5 0 0 0 7 8 0 9 7 0 0 5 0 9 0 15 6 0 0 5 7 15 0 8 9 0 0 0 6 8 0 11 0 0 0 0 9 11 0	Maximum Spanning Weight: 59

Week 9:

I. Given a graph, Design an algorithm and implement it using a program to implement Floyd-Warshall all pair shortest path algorithm.

Input Format:
The first line of input takes number of vertices in the graph.
Input will be the graph in the form of adjacency matrix or adjacency list. If a direct edge is not present between any pair of vertex (u,v), then this entry is shown as AdjM[u,v] = INF.

Output Format:
Output will be shortest distance matrix in the form of V X V matrix, where each entry (u,v) represents shortest distance between vertex u and vertex v.

Sample I/O Problem I:

Input:	**Output:**
5 0 10 5 5 INF INF 0 5 5 5 INF INF 0 INF 10 INF INF INF 0 20 INF INF INF 5 0	Shortest Distance Matrix: 0 10 15 5 15 INF 0 5 5 5 INF INF 0 15 10 INF INF INF 0 20 INF INF INF 5 0

II. Given a knapsack of maximum capacity w. N items are provided, each having its own value and weight. You have to Design an algorithm and implement it using a program to find the list of the selected items such that the final selected content has weight w and has maximum value. You can take fractions of items,i.e. the items can be broken into smaller pieces so that you have to carry