

# CSE 221: Algorithms (Lab)

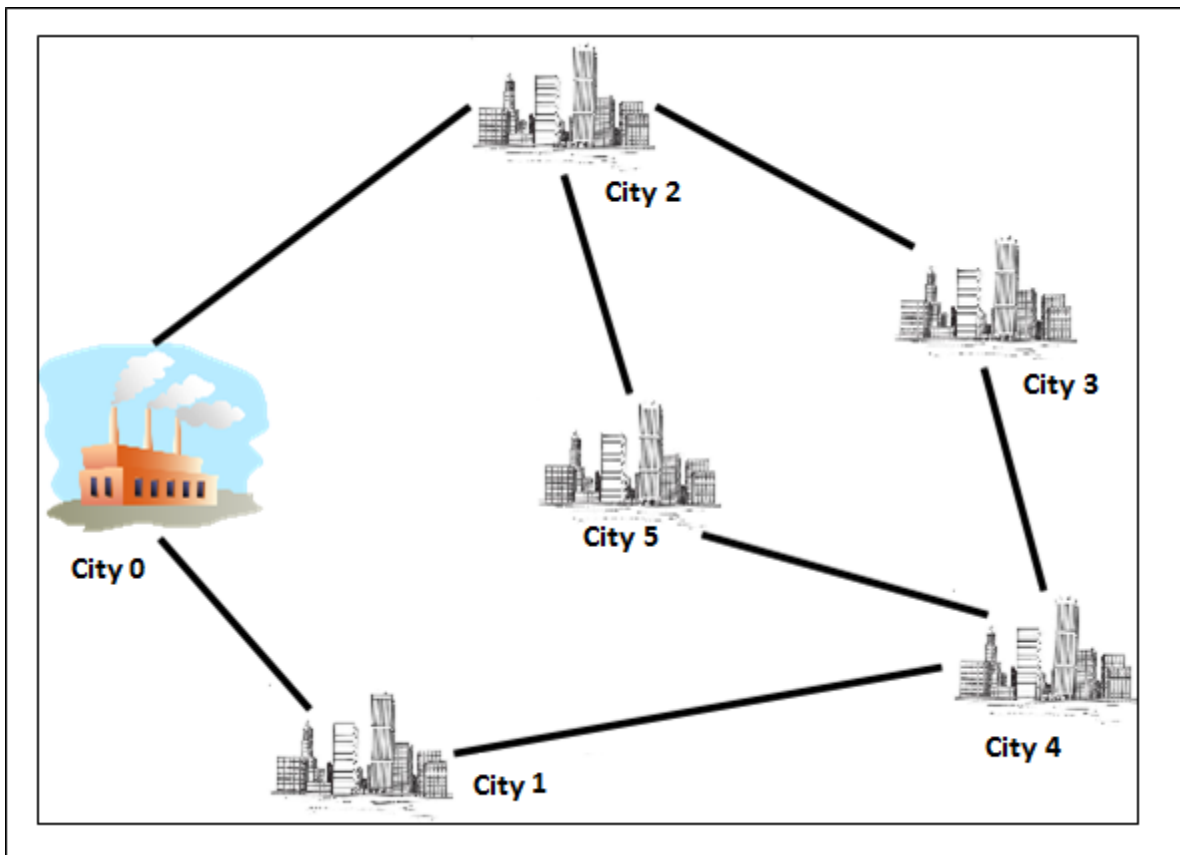
## Lab 1



### Abstract: Adjacency Matrix and Adjacency List

Fruit Co. is a prominent fruit supplier of Wander-land, supplying almost all kinds of fruits. Its factory is located at "City 0". Its operation is guided by Mr. Owner who is also the proud owner of that company.

Here is how Fruit Co. operates its business. From the single factory, it supplies fruits to other cities through highways. It has a graph showing cities as vertices and edges indicating whether two cities are connected by highways.

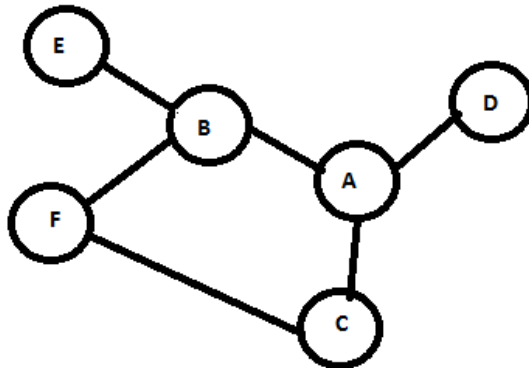


Mr. Owner recently learned about graphs and tried use them in business. (He loves computer science but unfortunately, has no background on that subject !). Following that Mr. Owner tried to implement concepts of graph himself and **failed!!** So he recruited you to do the job for him.

**Details:**

In this lab, you are required to help Mr. Owner perform path searching on the above given graph. Design a program that takes city names (i.e. vertices) and bidirectional connectivity between cities (i.e. edge) as inputs. For the above graph, a user will give no.of nodes and the edges. Using 2 dimensional array form an adjacency matrix and also using concept of linked list make an adjacency list of the graph above .

For further clarification an example, for a particular starting node, is given below:



**Example:**

User Input:

Node: E,F,B,C,A,D

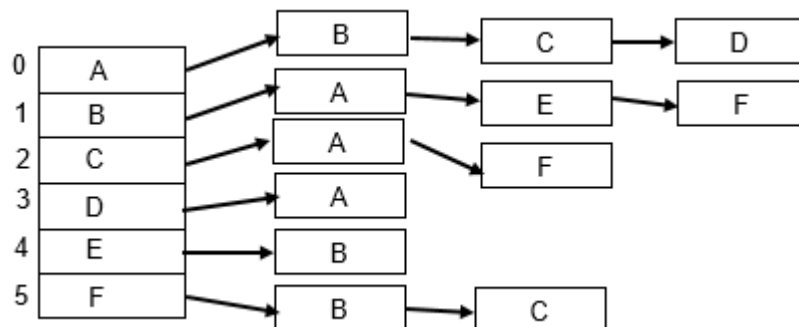
Edges: (A,B), (C,A),(E,B),  
(D,A),(F,B), (C,F)

**Adjacency Matrix:**

	A	B	C	D	E	F
A	0	1	1	1	0	0
B	1	0	0	0	1	1
C	1	0	0	0	0	1
D	1	0	0	0	0	0
E	0	1	0	0	0	0
F	0	1	1	0	0	0

Or

**Adjacency List:**



**Input and Output Format:**

The first line of the input will contain 2 integers. They are number of vertices and number of edges. These values will be user given. Following that there will be as many lines as there are edges. Each line will contain the two associated vertex of the corresponding edge, which will also be user initiated.

If **n** is given as input for number of vertices then the labels of the vertices will be from **0** to **n-1**. For example, in the above graph n is 6 and vertices are labeled with 0 to 5. This will help in indexing the array.