**LAB-6**

**AIM:**

Write a program to implement BFS and DFS for the given graph and check whether the graph is strongly connected or not. Also print the pre and post visited times.

**EXPERIMENT:**

Breadth first Search:

BFS is a traversing algorithm where you should start traversing from a selected node (source or starting node) and traverse the graph layer wise thus exploring the neighbor nodes (nodes which are directly connected to source node) and then move towards the next-level neighbor nodes. We must traverse the graph breadthwise as follows:

* First move horizontally and visit all the nodes of the current layer
* Move to the next layer

Depth First Search:

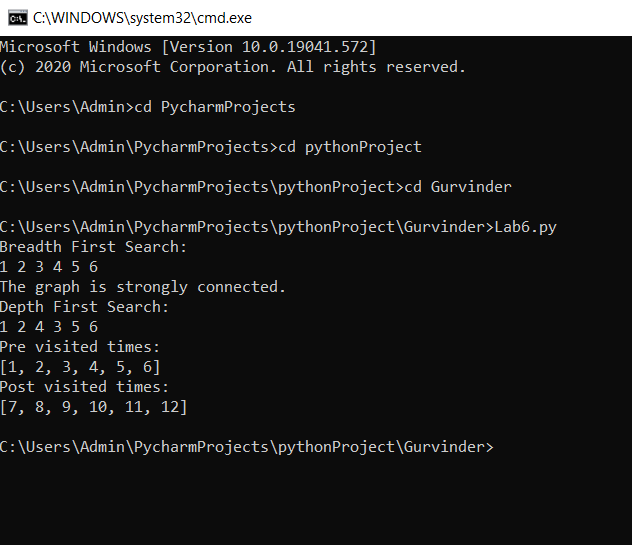
In DFS we traverse the graph depth wise. The algorithm starts at the selected node and explores as far as possible along each branch before backtracking. It continues till it has visited all the nodes in a branch before backtracking and then moves on to the next branch.

Checking whether a graph is strongly connected or not:

In case of an undirected graph (given graph) we should perform BFS/DFS starting from any vertex. If BFS/DFS visits all the nodes then it is said to be strongly connected.

The given graph is strongly connected since all the nodes were visited.

**OUTPUT:**



**CONCLUSION:**

BFS and DFS were implemented on the given graph and it was also checked whether the graph is strongly connected or not. Also, the pre and post visited times was printed.