

**DSA LAB**  
**Lab Assignment number 15**

**Name:** Aamir Ansari

**Batch:** A

**Roll no:** 01

**Aim:** Implementation of BFS and DFS on a directed graph using adjacency matrix.

**A) BFS Algorithm:**

BFS is Breadth First Search

Algorithm for breadth-first search in a graph G beginning at a starting node A

Step 1: SET STATUS = 1 (ready state) for each node in G.

Step 2: Enqueue the starting node A and set its STATUS = 2 (waiting state)

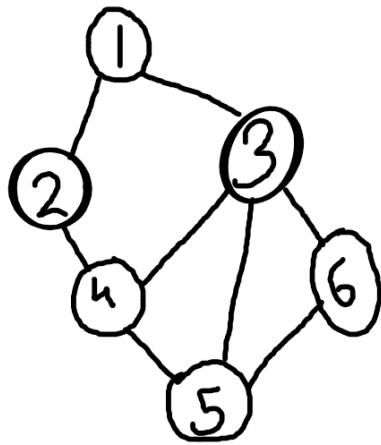
Step 3: Repeat Steps 4 and 5 until QUEUE is empty

Step 4: Dequeue a node N. Process it and set its STATUS = 3 (processed state).

Step 5: Enqueue all the neighbors of N that are in the ready state (whose STATUS = 1) and set their STATUS = 2 (waiting state)

[END OF LOOP

Step 6: EXIT



Consider the following graph, it's BFS traversal would be

1, 2, 3, 4, 5, 6

## B) DFS Algorithm

DFS is Depth First Search tree

Algorithm for depth-first search in a graph G beginning at a starting node A

Step 1: SET STATUS = 1 (ready state) for each node in G.

Step 2: Push the starting node A on the stack and set its STATUS = 2 (waiting state)

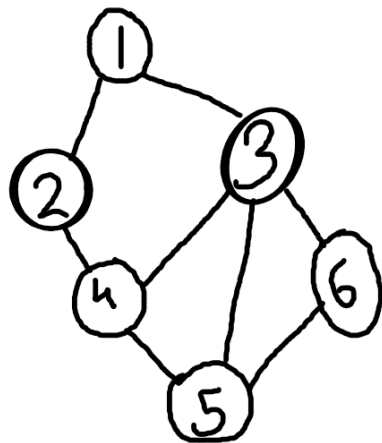
Step 3: Repeat Steps 4 and 5 until STACK is empty

Step 4: Pop the top node N. Process it and set its STATUS = 3 (processed state).

Step 5: Push on to the stack all the neighbours of N that are in the ready state (whose STATUS = 1) and set their STATUS = 2 (waiting state)

[END OF LOOP]

Step 6: EXIT



Consider the above graph, it's DFS traversal would be:

1, 2, 4, 3, 6, 5