Depth First Search (DFS)

→ In DFS, we start with a node and start exploring its connected nodes keeping on suspending the exploration of previous nodes.

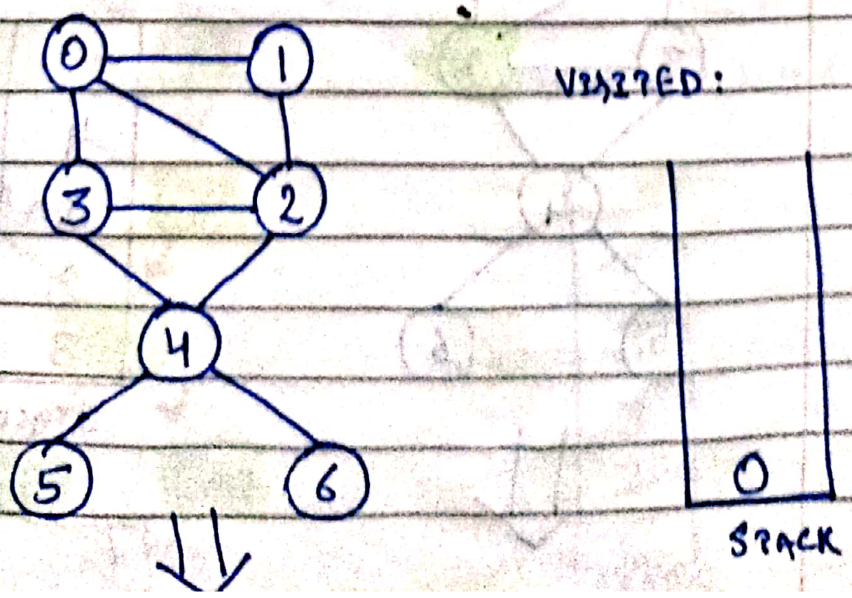
→ DFS Procedure :

→ Let's define a stack named which would hold the nodes we'll be suspending to explore one by one later. And stack, is a data structure in which the element you push the last comes out the first. Choose any node as the source node and push it into the top of the stack we created. We would maintain another array holding the status and mark it visited or add it to the visited list.

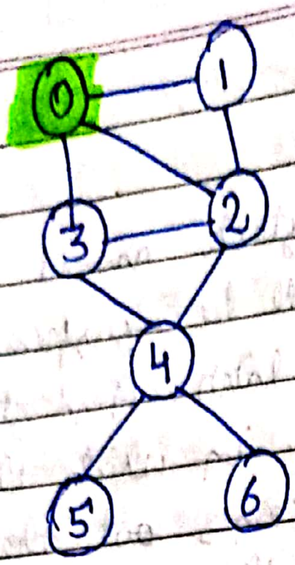
→ Take the top item of the stack and mark it visited or add it to the visited list.

→ Create a list of the nodes directly connected to the vertex we visited. Push the ones which are still not visited into the stack.

→ Repeat steps 2 and 3 until the stack is empty.



VISITED: 0

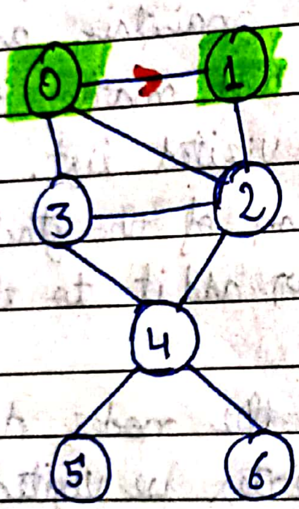


1
2
3

STACK

⇓

VISITED: 0 1

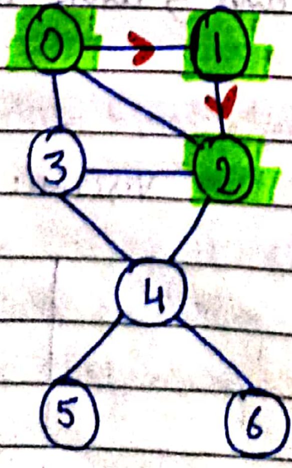


2
2
3

STACK

⇓

VISITED: 0 1 2

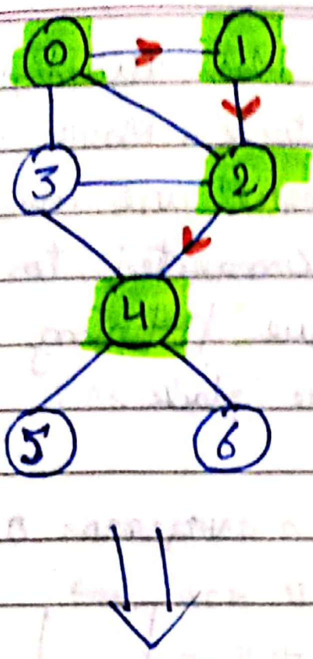


4
3
2
3

STACK

⇓

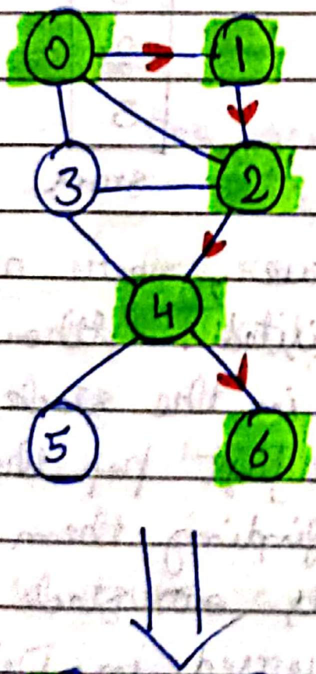
Page No.
Date



VISITED: 0 1 2 4

6
5
3
3
3
2
3

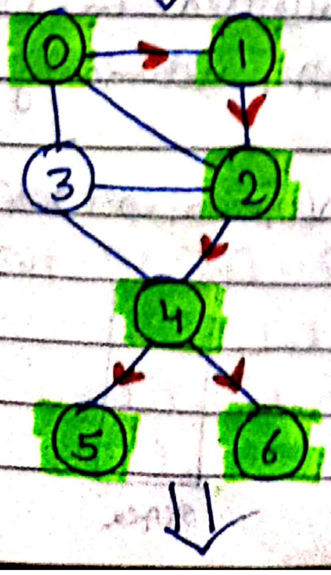
STACK



VISITED: 0 1 2 4 6

5
3
3
3
2
3

STACK



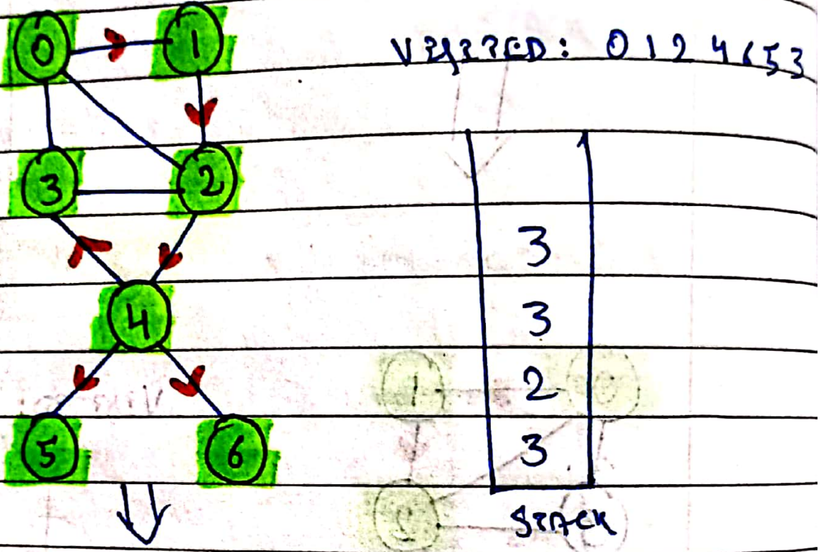
VISITED: 0 1 2 4 6 5

3
3
3
2
3

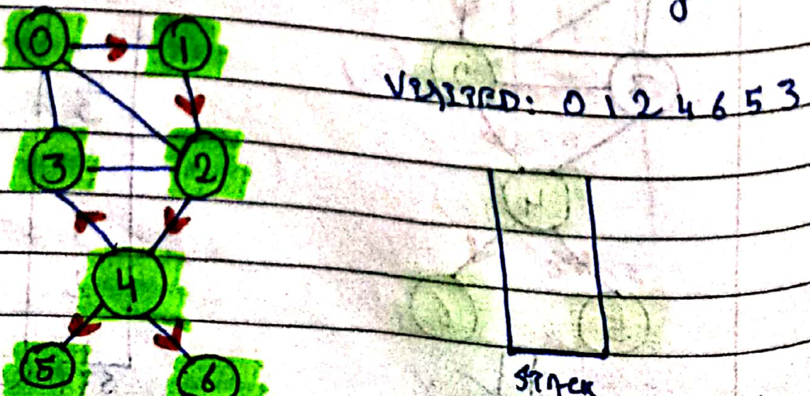
STACK

Page No.
Date

Node 3 comes next to be visited, being on the top in the stack. Mark node 3 visited and again there are no nodes left unvisited and connected to node 3. So, we just continue popping out elements from the stack.



Now if we could observe, there are no nodes left to be visited. Although there are elements in the stack to be explored. So, we just pop them one by one and ignore finding them already visited. And this gets our stack emptied and every node traversed in Depth First Search manner, ultimately.



Scanned with CamScanner

Page No. ______
Date

→ PSEUDOCODE

→ INPUT: A GRAPH G = (V, E) and source code s in V
→ Algorithm:

```
DFS (G, u)

U.VISITED = TRUE
FOR EACH V ∈ G.ADJ (u)
    IF V.VISITED == FALSE
        DFS (G, V)

INIT ( )
    FOR EACH U ∈ G
        U.VISITED = FALSE
    FOR EACH U ∈ G
        DFS (G, u)
```

Scanned with CamScanner