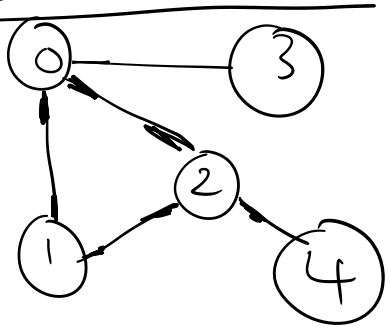
29 September 2023 19:54

Undirected Graph



Task is to visit all the nodes using DFS. Starting node is 0.

DFS Algorithm :-

Step 1 :- [ ] ← Queue

[ ] ← stack.

[0] ← Q & [1, 2, 3] ← Stack.

Step 2 :- Add the latest element from Stack to queue.

Q = [0, 1] & [2, 3] ← Stack.

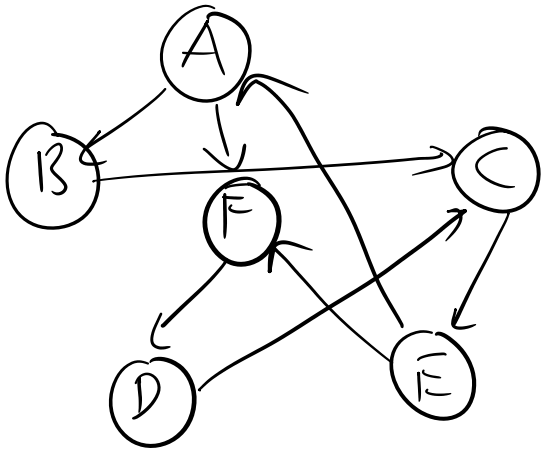
Step 3 :- Add the latest element from Stack to queue

Q = [0, 1, 2] & [3, 4]

Step 4 :- Add all the remaining element from Stack to queue as the node 3 & node 4 do not have any new neighbors.

Q = [0, 1, 2, 3, 4] & [ ] ← Stack

Directed Graph :-



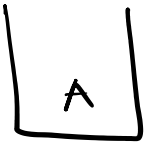
Task :- to start at node A & visit all the nodes.

Step 1 :- Create an adjacency list for the above graph

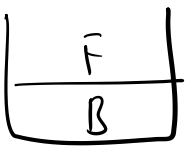
A → B, F
B → C
C → E
D → C
E → A, F
F → D

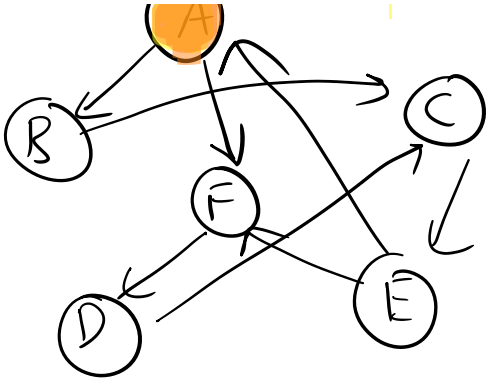
Step 2 :- Create an empty stack. ⌊⌋

Step 3 :- Push 'A' into the stack

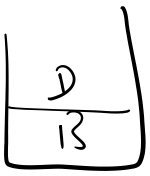


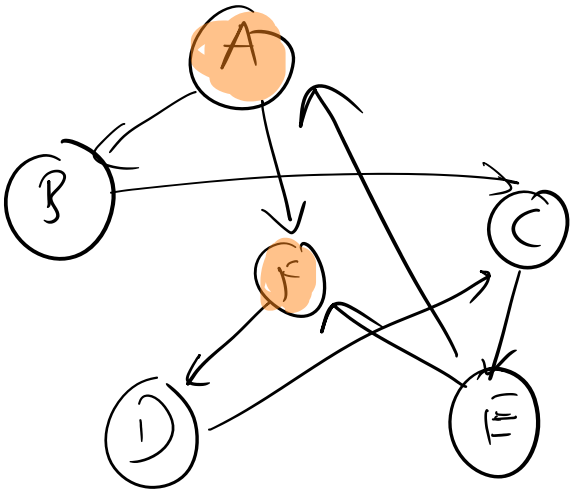
Step 4 :- Pop A & push A's neighbors i.e. B, F into the stack & mark A as visited.



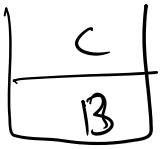


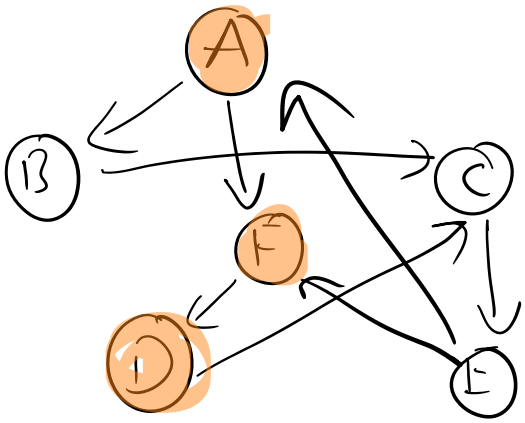
Step 5 :- Pop F & push F's neighbour (D) into stack. Mark F as visited.



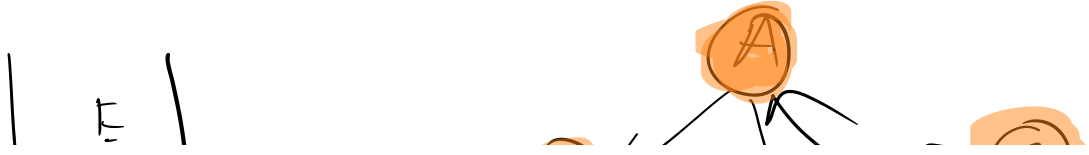


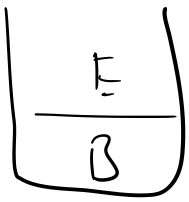
Step 6 :- Pop D & push D's neighbour (C). Mark D as visited

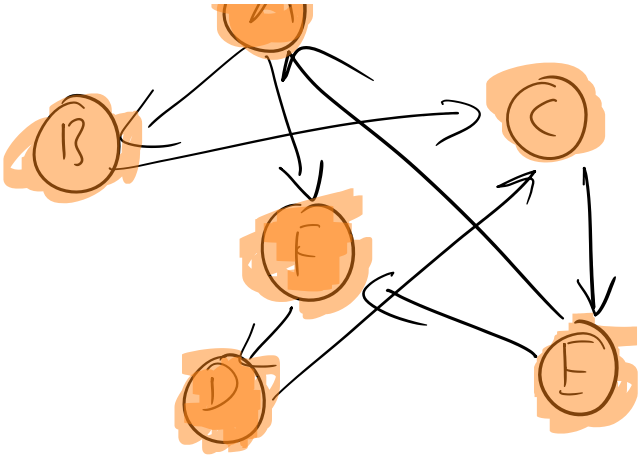




Step 7 :- Pop C & push E. Mark C as visited.







Step 8 :- Pop E. Mark E as visited. [B]

Step 9 :- Pop B. Mark B as visited. [ ]

As we see that the stack is empty indicating all nodes are already visited & no more nodes are left.