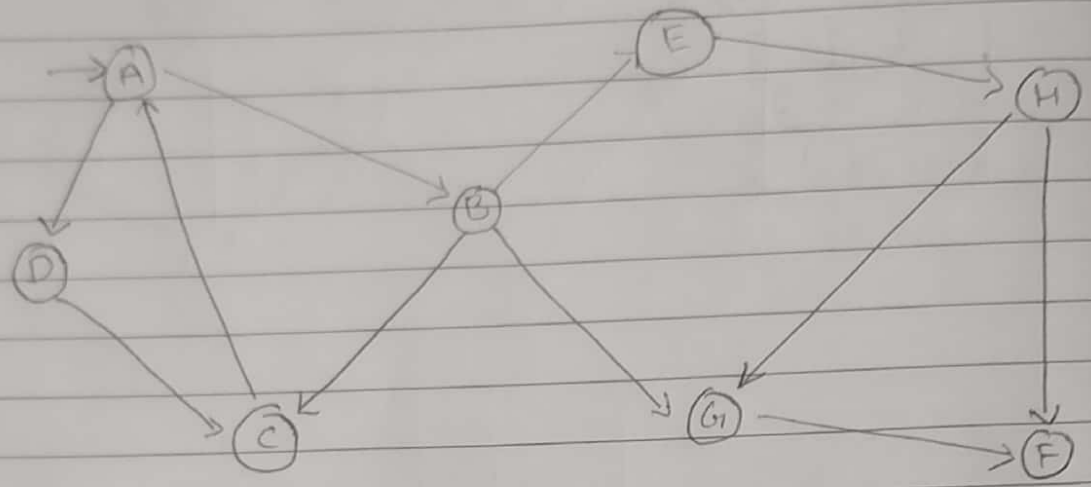
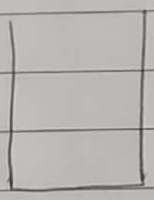
Q. Consider the following Graph.
Node A is the root node.
Update visting time & Finish time for each node.

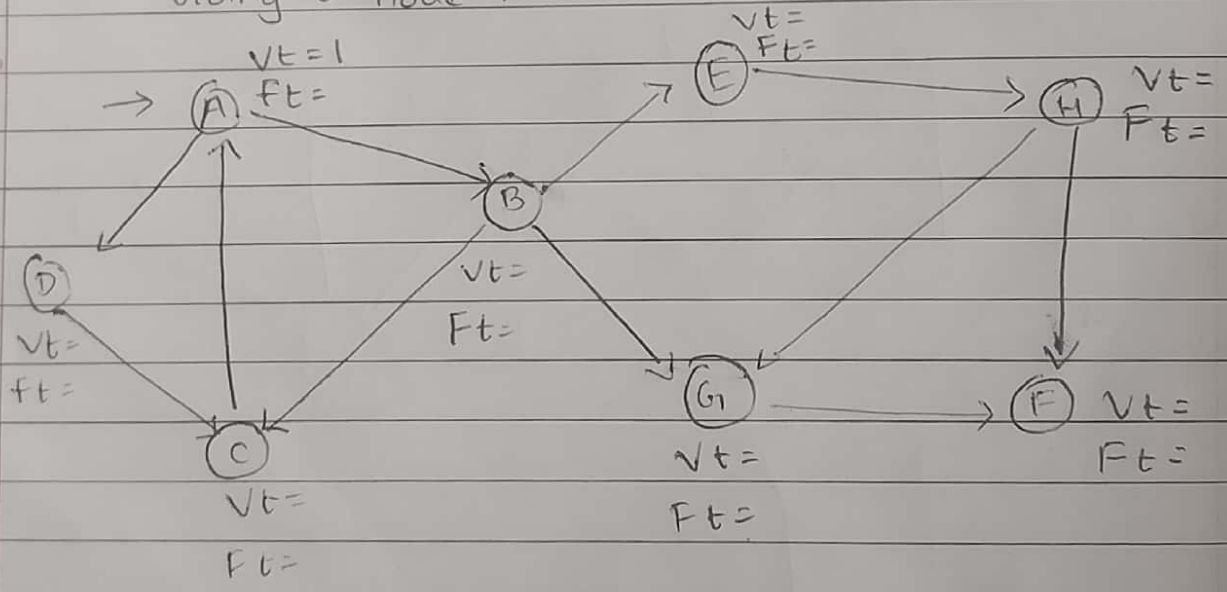


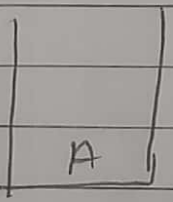
Stack  is empty

Vt → visiting time
Ft → Finishing time.
Black arrow for diag
Blue arrow for path
(Tree Edge)

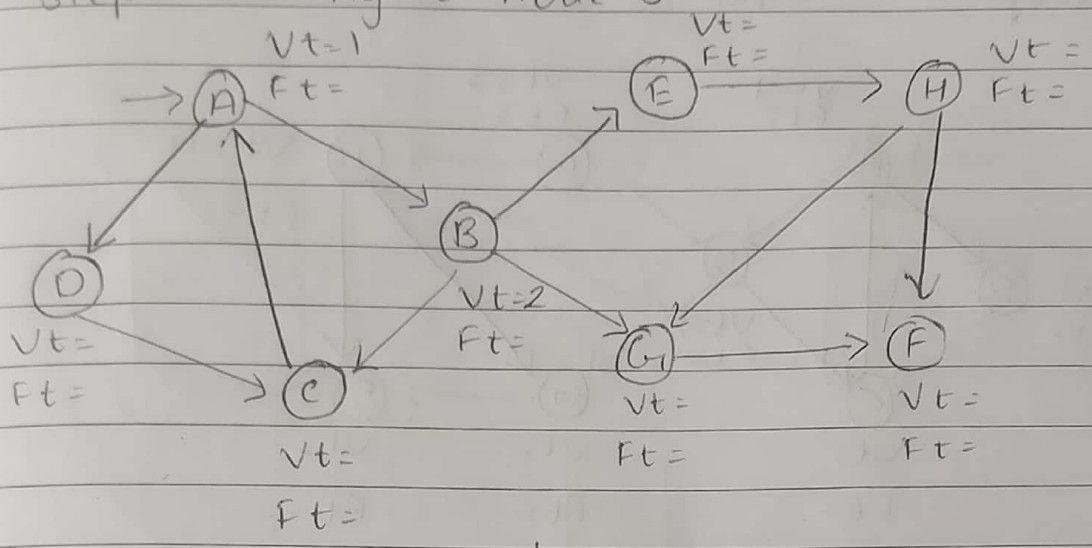
Step 1

Going to node A.



Stack 

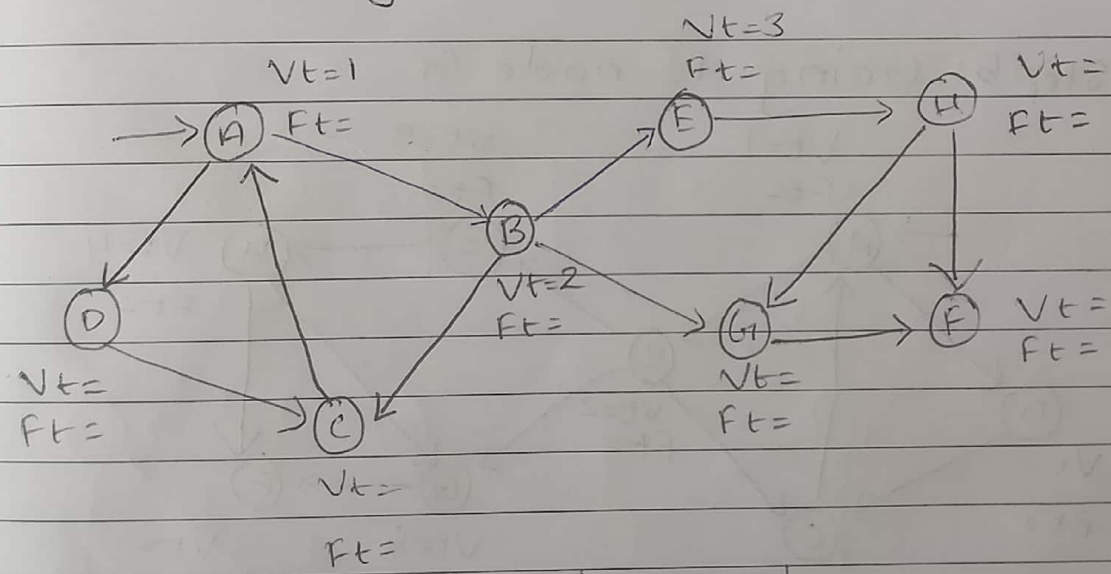
Step 2: Going to node B



Stack:

B
A

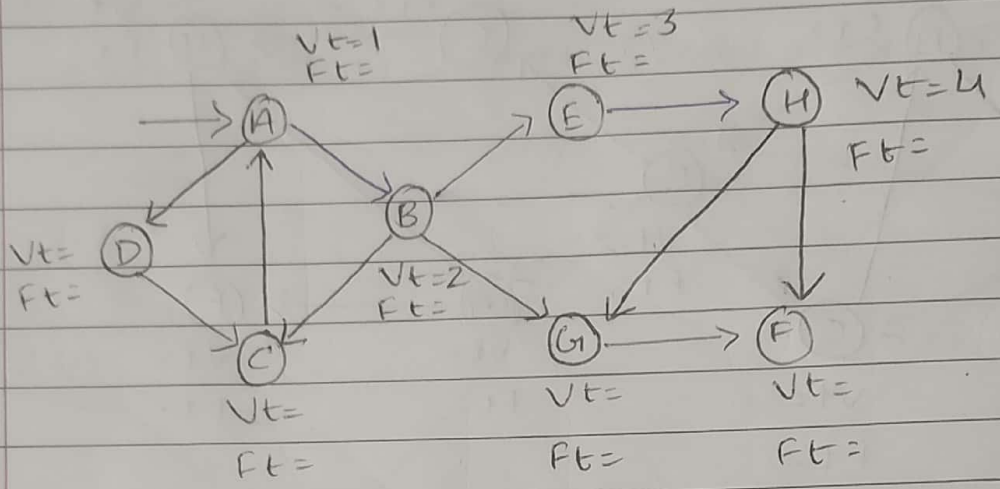
Step 3: Going to node E



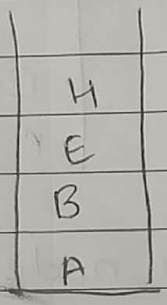
Stack

E
B
A

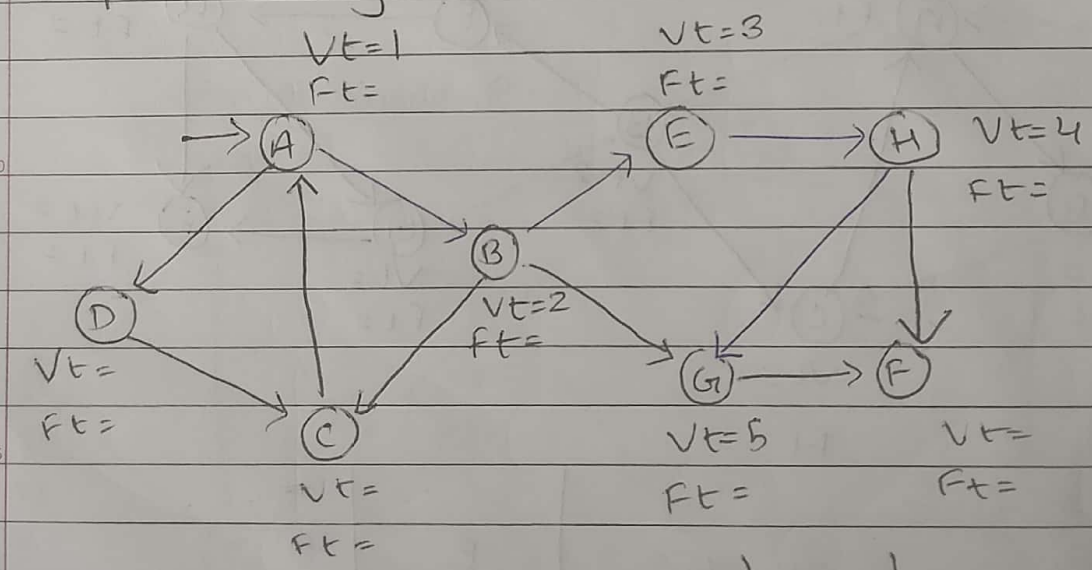
Step 4: Going to Node H



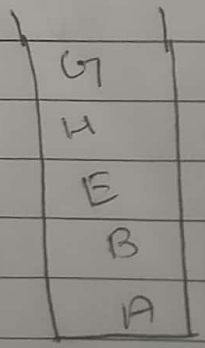
Stack.



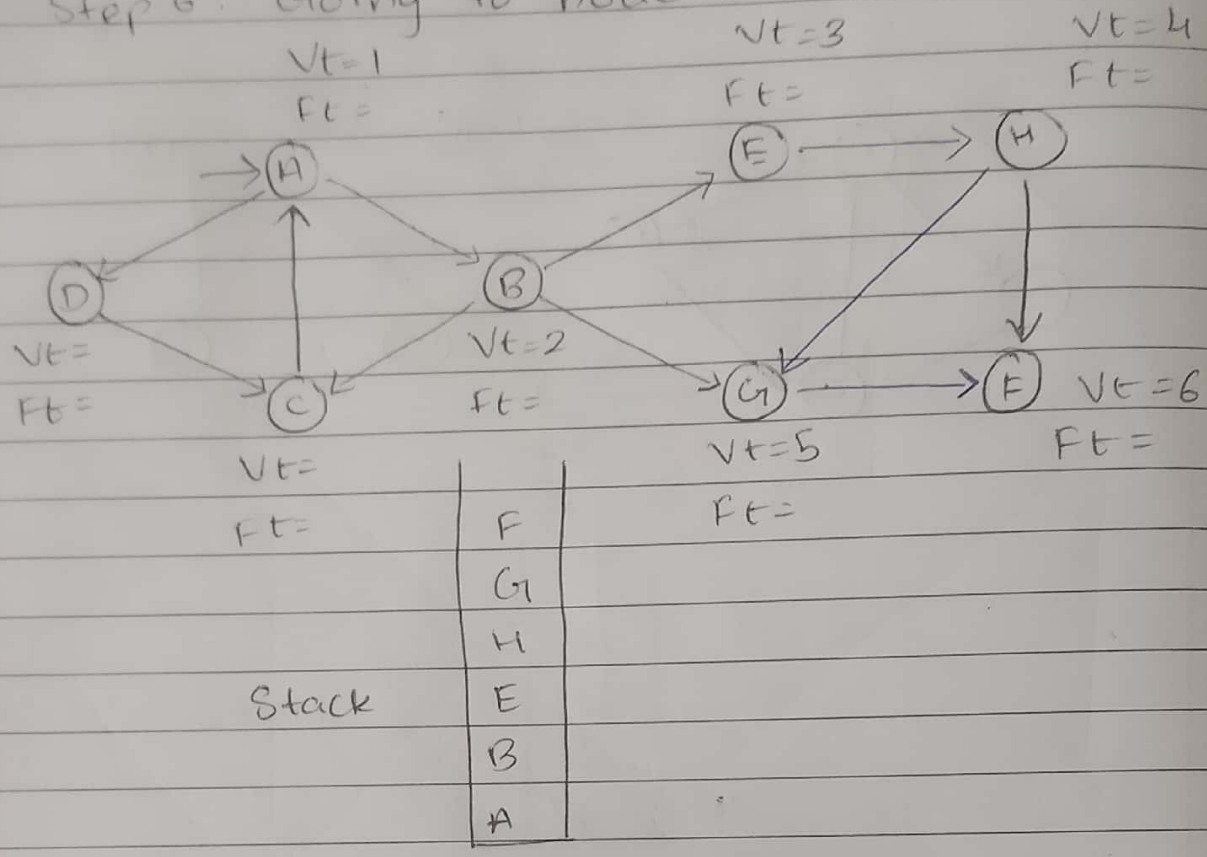
Step 5: Going to node G



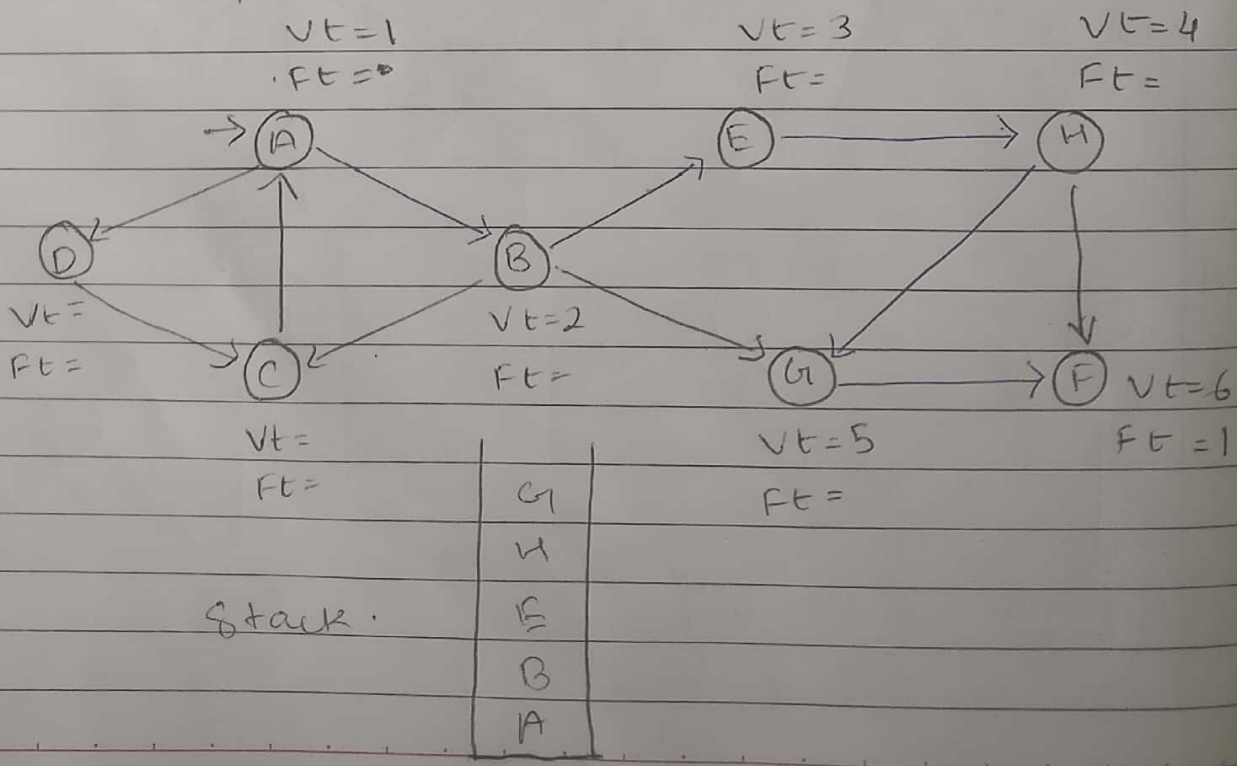
Stack.



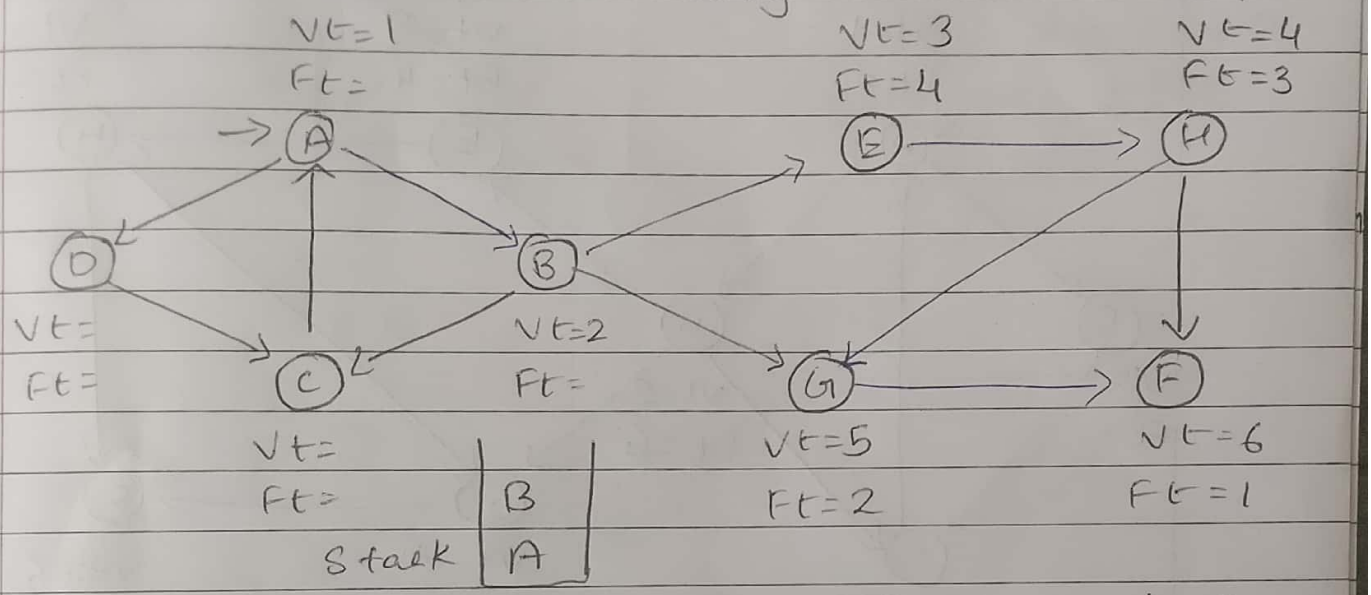
Step 6: Going to node F



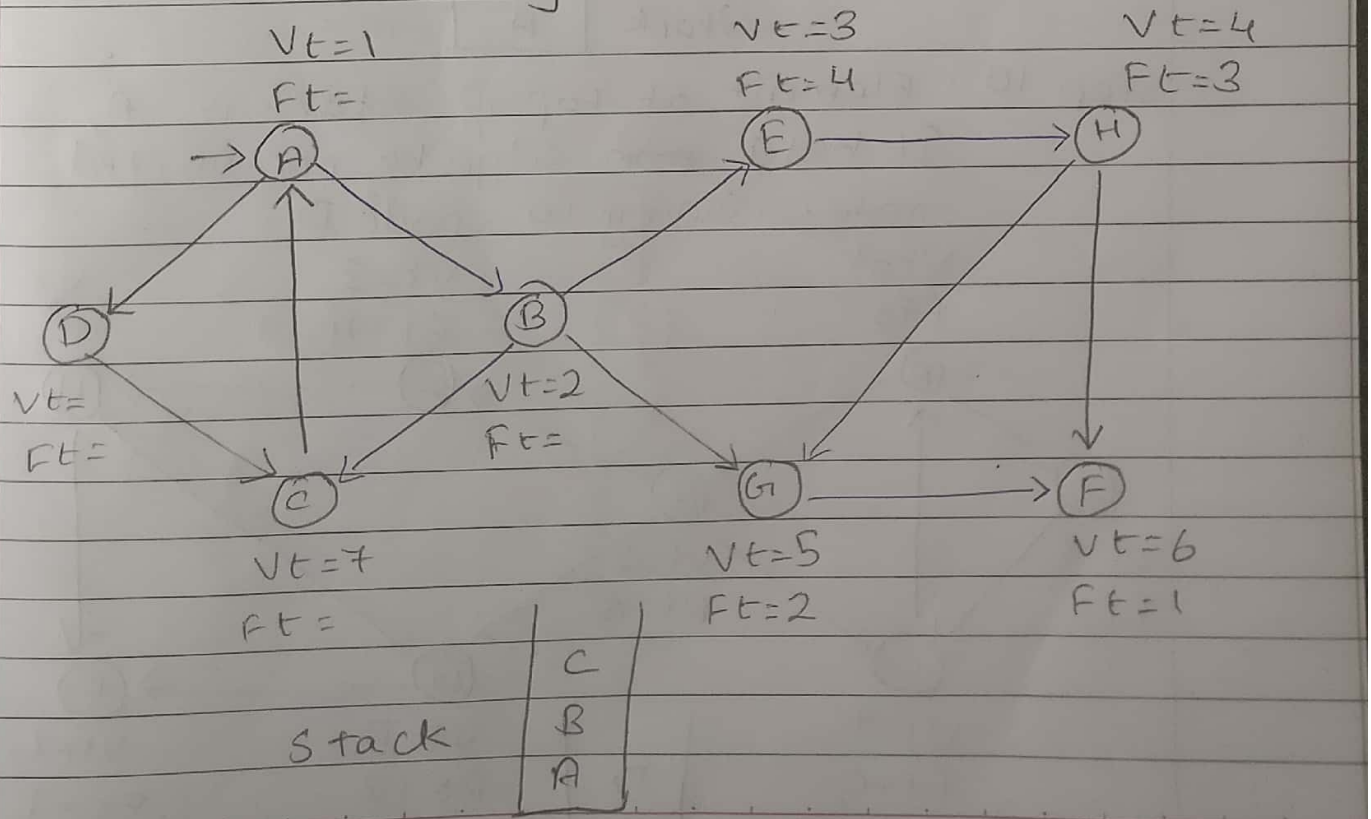
Step 7: Node F does not have any node to visit. So pop F from the ~~queue~~ stack, back ~~track~~ to the next element and update Finishing Time (Ft)



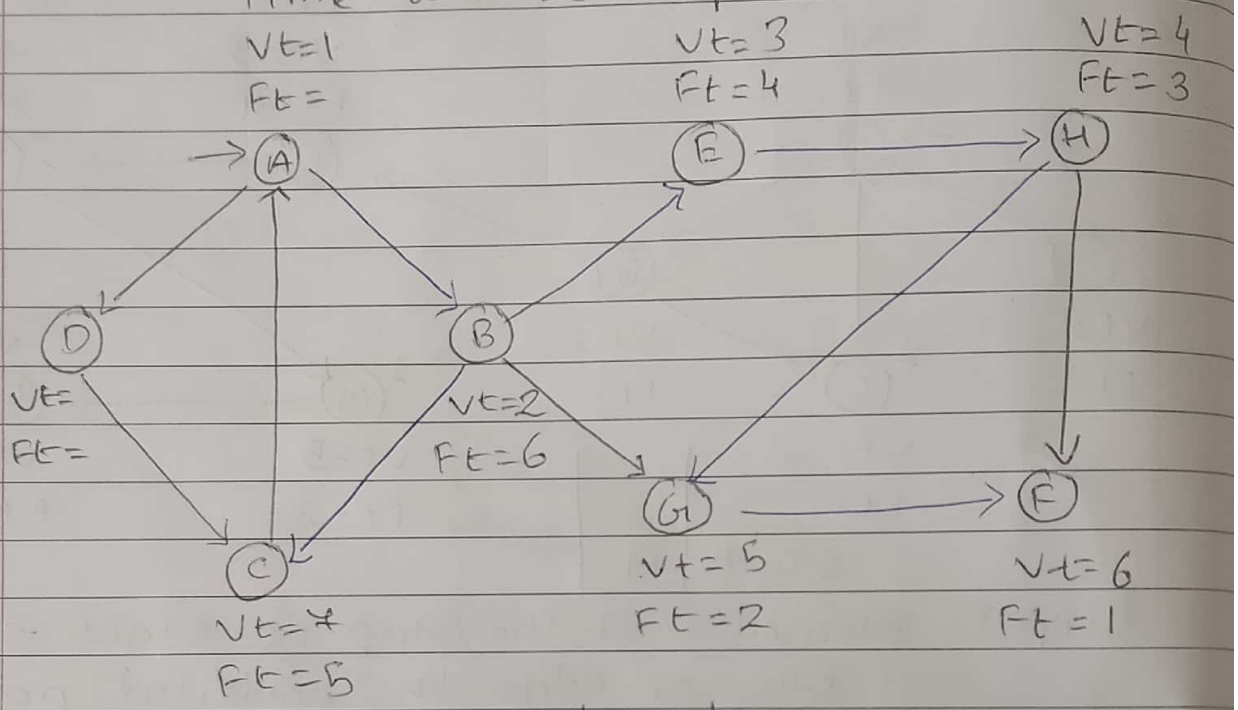
Step 8: Element at the top of stak is G, followed by H and E. All these nodes do not have an edge that goes to any unvisited node. So they will be popped from the stack one after the other and Finishing Time will be updated



Step 9: Element at the top of Stack is B, it has an edge to unvisited node C, So visiting Node C.

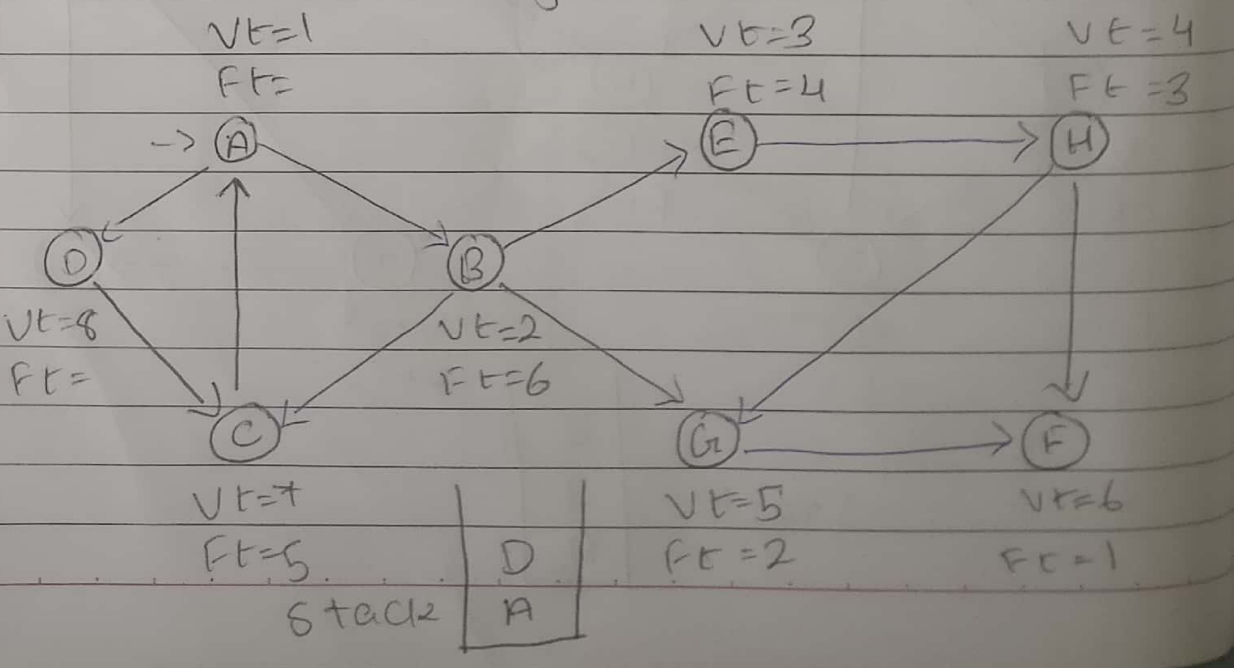


Step 9: Element at top of stack is C, followed by B. These two element do not have any edge to unvisited node. So they will be popped one after the other and Finishing Time will be updated.

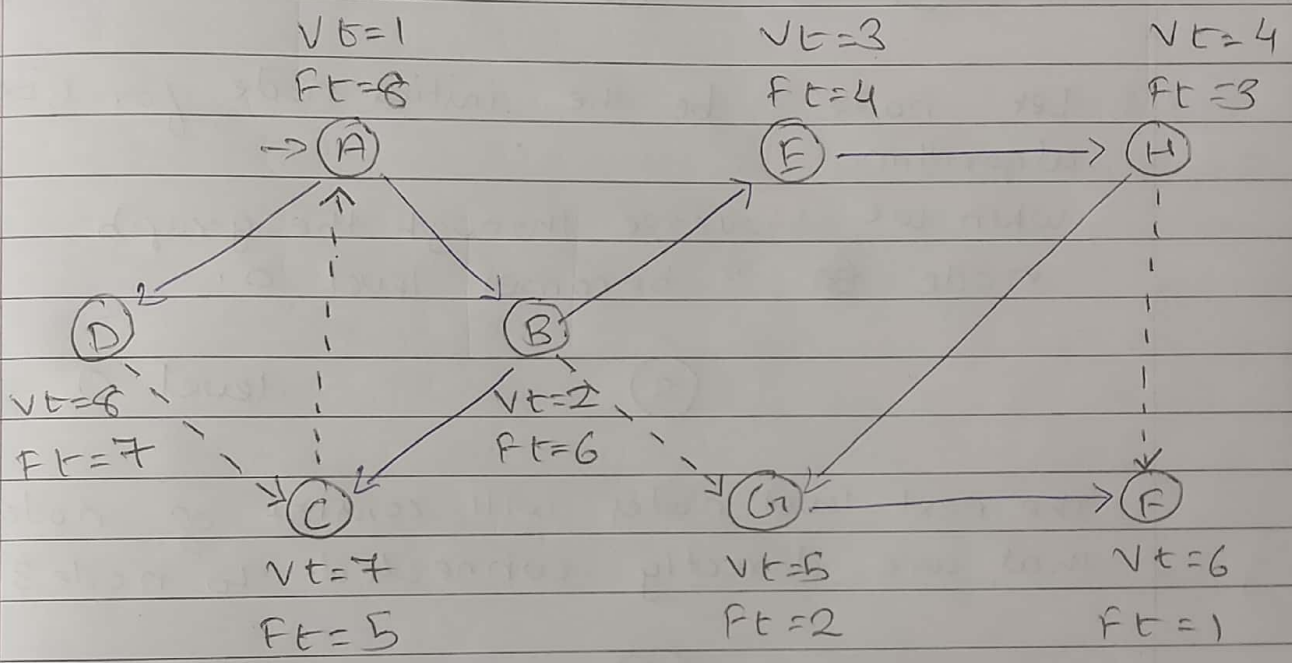


Stack | A

Step 10:- Element at top of stack is A, It has 1 ~~non~~ edge to non-visited node. Going to node D.



Step 11: Element at top of stack is D, followed by A. They do not have an edge to any unvisited node. So, elements will be popped and Finishing time will be updated.
In the following final ~~dig~~ diagram Back edge will be drawn from dotted lines



Stack.