DEPTH FIRST METHOD USING A STACK

Slides based on :

VINAY R U - DEPT OF COMPUTER SCIENCE- JSS COLLEGE MYSURU

Documents are here:

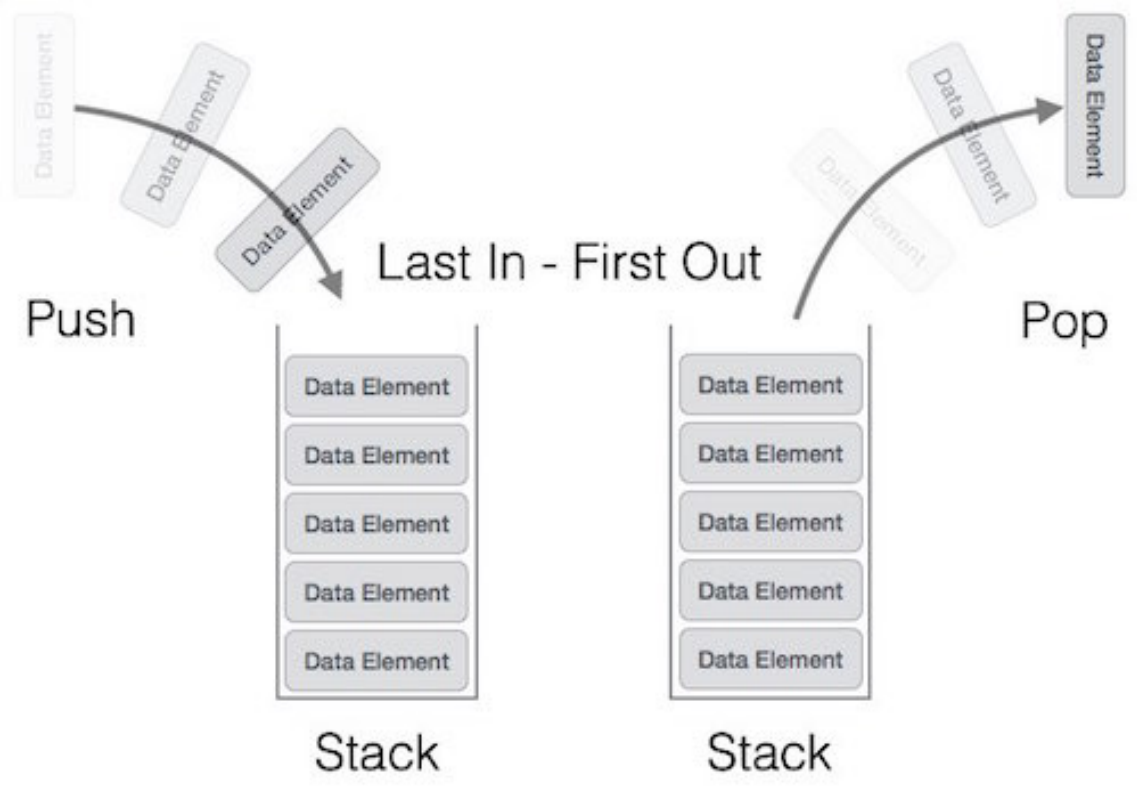
https://www-l2ti.univ-paris13.fr/~viennet/ens/2024-USTH-Graphs

Depth-First Search (DFS) (reminder)

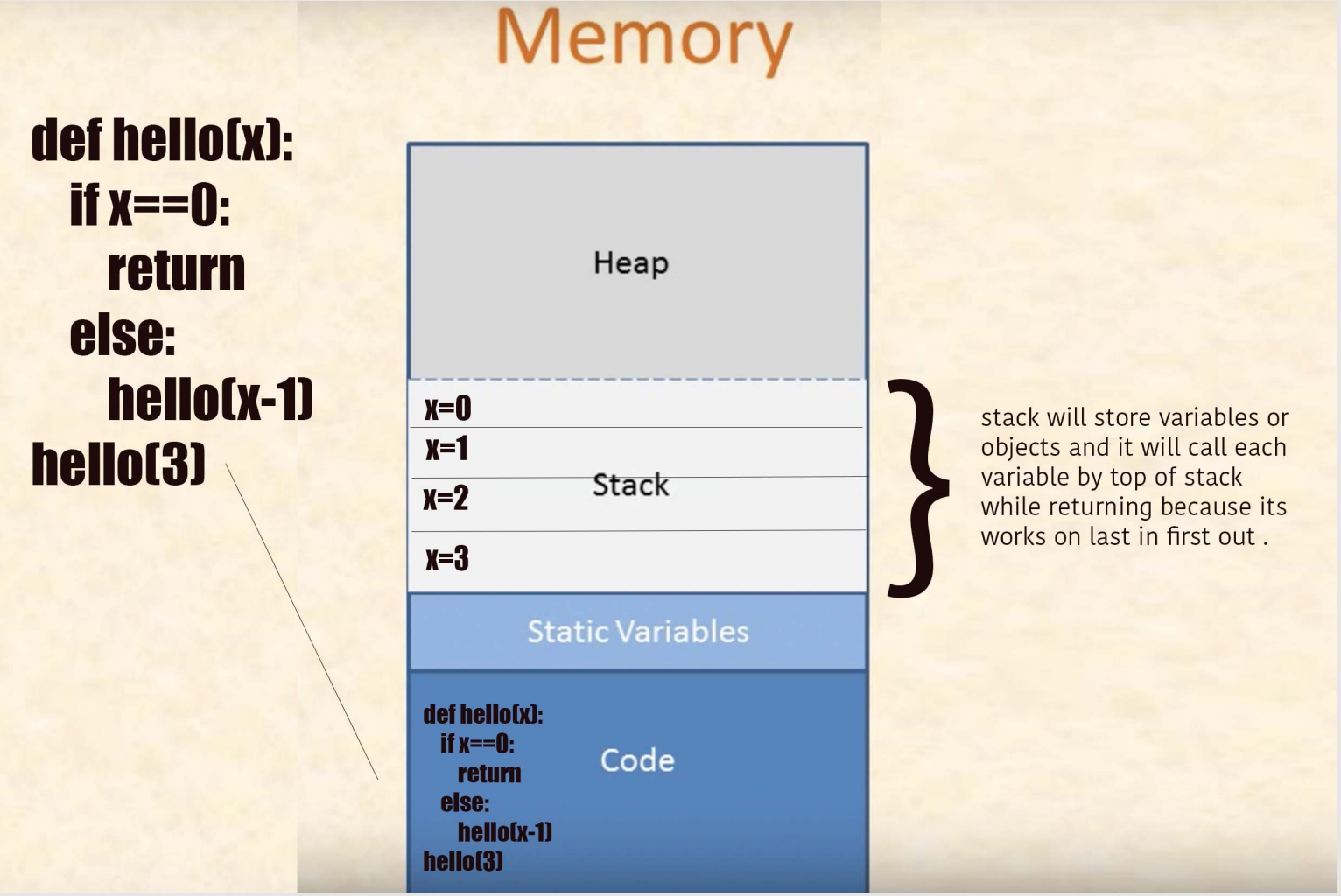
DFS algorithm: recursive formulation

```
def DFS(u):
    mark u as "explored"
    for each edge (u, v) incident to u:
        if v is not marked as "explored":
            DFS(v)
```

Stacks (reminder)



Recursion and stack



DEPTH FIRST SEARCH USING STACK

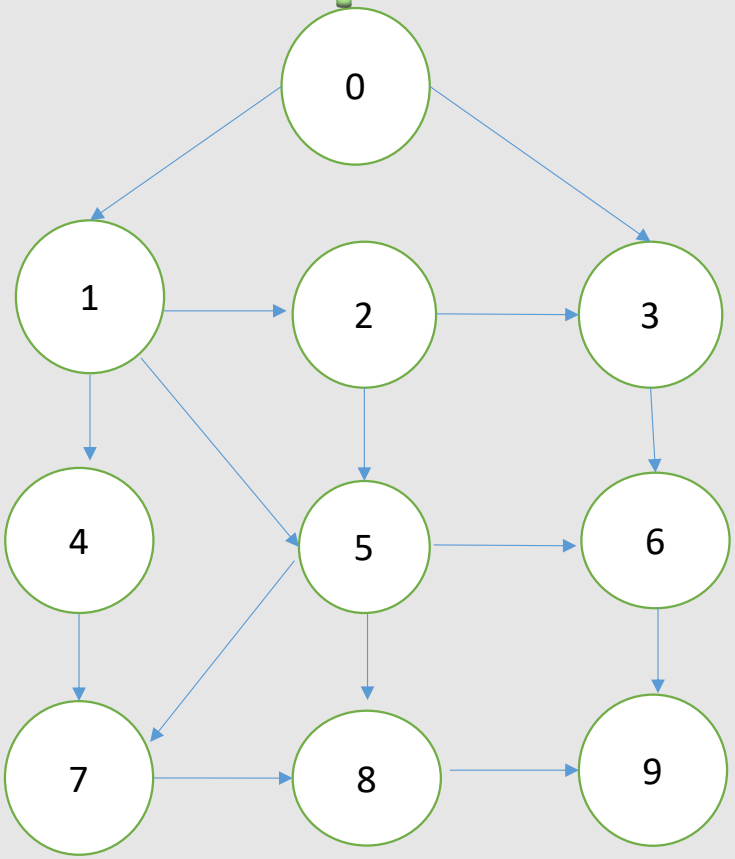
At the start of the algorithm all vertex will be in *initial* state.

Initially **Stack** is Empty

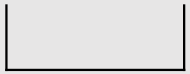
ALGORITHM

1) PUSH starting vertex into the **stack**
2) POP a vertex from the **stack**
3) If poped vertex is in initial state, **visit it** and change the state from *initial* to *visited* state, then push all unvisited vertices adjacent to poped vertex.
4) Repeat Step 2 and 3 until stack is empty

Example



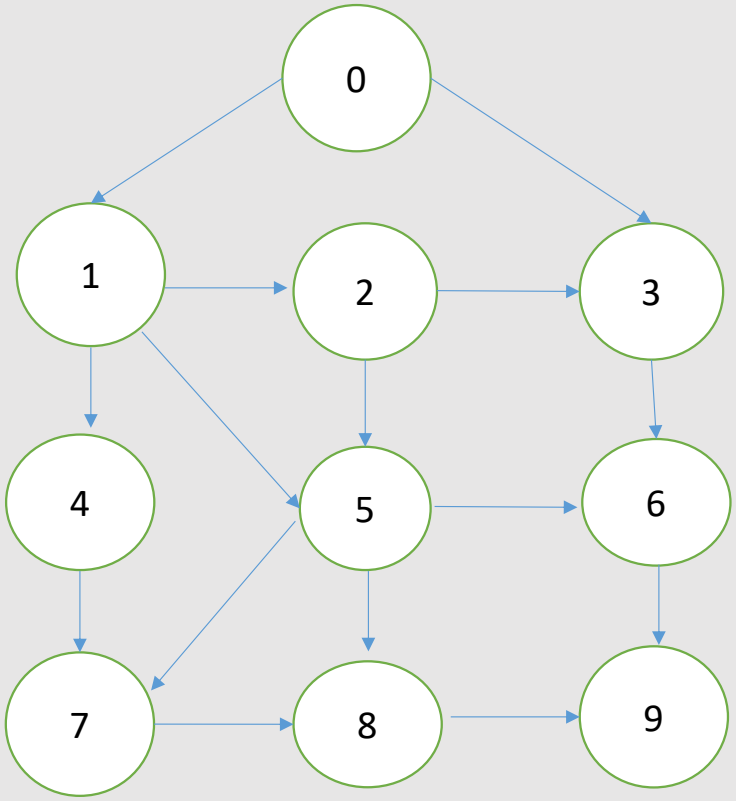
- Initially Stack is Empty
- All vertices are in *initial* state



Stack

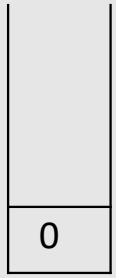
Initial state

0	1	2	3	4	5	6	7	8	9

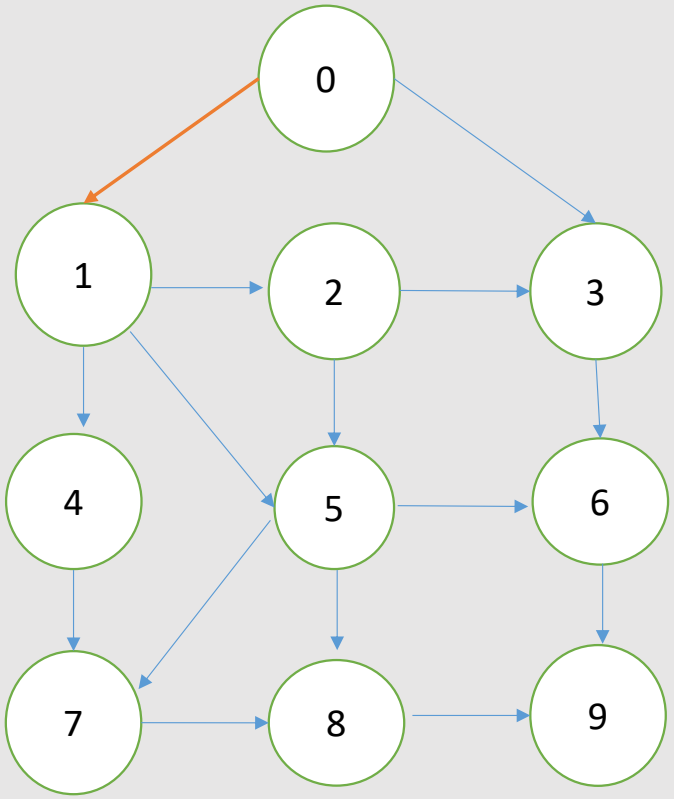


❖ Push a starting vertex into the stack

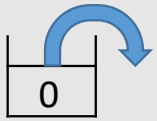
In this example we selected 0 as a starting vertex because in-degree of 0 is zero



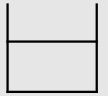
Stack



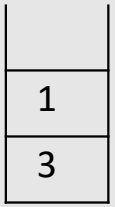
- Pop a vertex from the stack



0 is poped



- If Poped vertex is in initial state visit it and change the state from initial state to visited state. Here 0 is in initial state so visit 0 and change initial state to visit state.



Stack

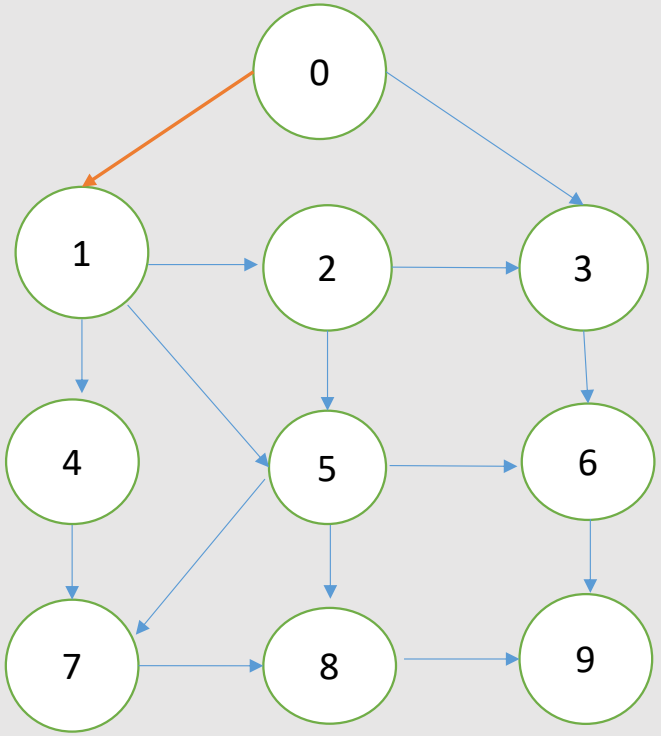
- Push all unvisited vertices adjacent to Poped vertex
- Unvisited vertices Adjacent to 0 are: 3 and 1: push 3 and 1

Initial state

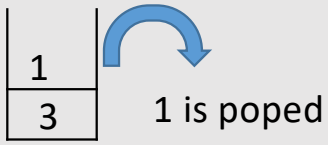
0	1	2	3	4	5	6	7	8	9

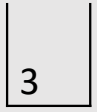
Visited state

0								



- Pop a vertex from the stack
- If Poped vertex is in initial state visit it and change the state from initial state to visited state.



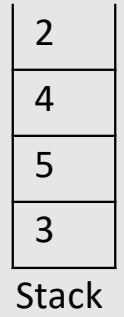


Here 1 is in initial state so visit 1 and change initial state to visit state

- Push all unvisited vertices adjacent to Poped vertex

Unvisited vertices Adjacent to 1 are: 2, 4 and 5

- Push 5, 4, 2 into the stack

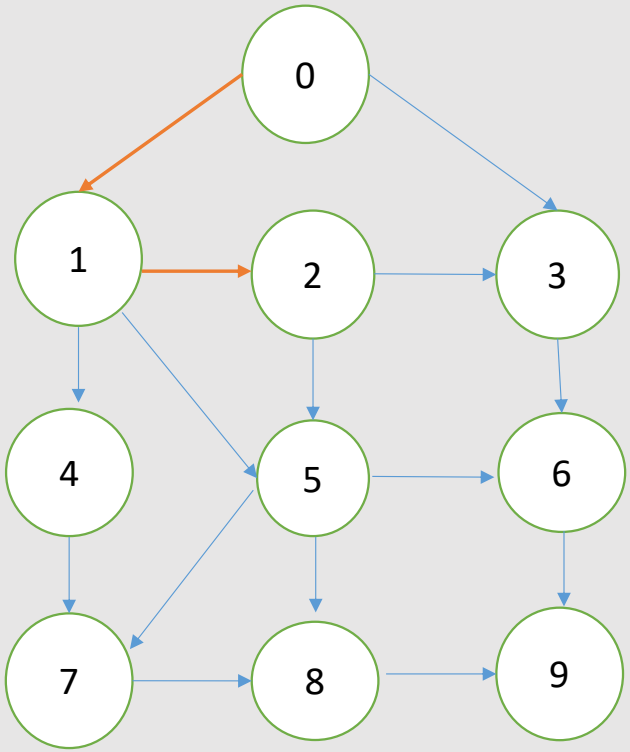


Initial state

1	2	3	4	5	6	7	8	9

Visited state

0	1							

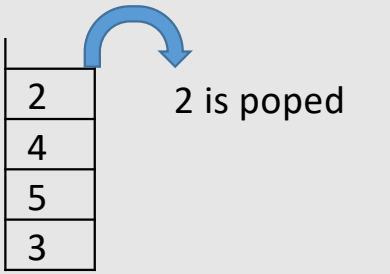


❖ Pop a vertex from the stack
2 is poped from the stack

❖ If Poped vertex is in initial state visit it and change the state from initial state to visited state.

Here 2 is in initial state so visit 2 and change initial state to visit state

❖ Push all un visited vertices adjacent to Poped vertex
Un visited vertices Adjacent to 2 are: 5,3
❖ Push 5,3 into the stack

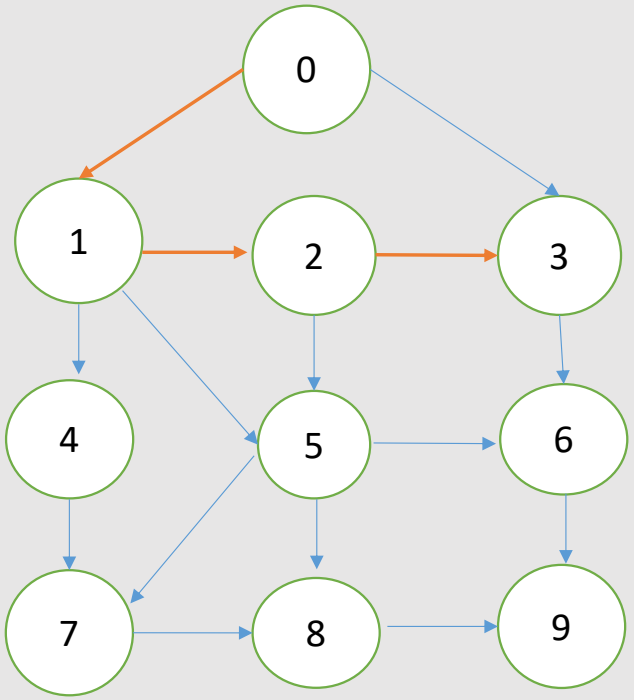


Initial state

2	3	4	5	6	7	8	9

Visited state

0	1	2						



- Pop a vertex from the stack
- If Poped vertex is in initial state visit it and change the state from initial state to visited state.

Here 3 is in initial state so visit 3 and change initial state to visit state

- Push all unvisited vertices adjacent to Poped vertex

Unvisited vertex Adjacent to 3 is: 6

- Push 6 into the stack

3 is Poped

5
4
5
3

6
5
4
5
3

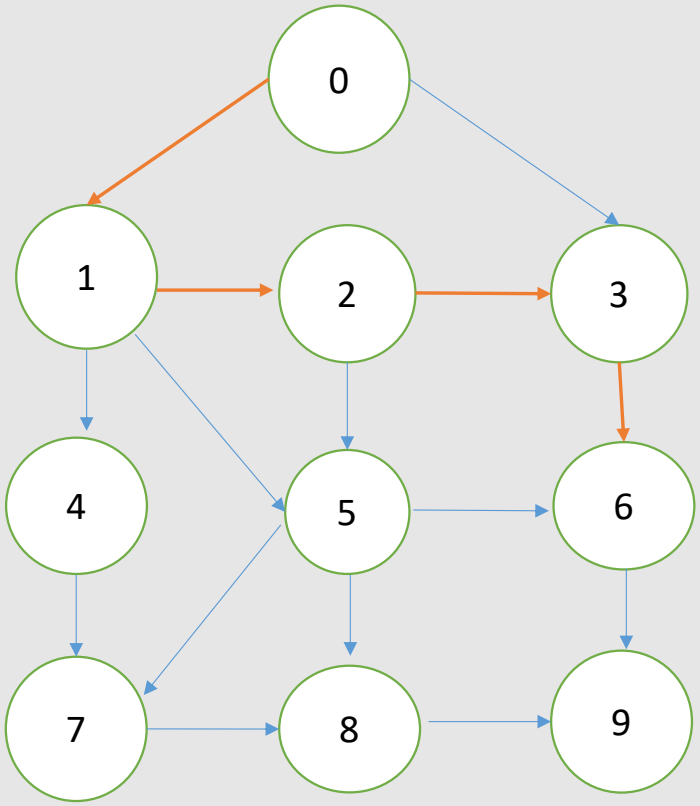
Stack

Initial state

3	4	5	6	7	8	9

Visited state

0	1	2	3					



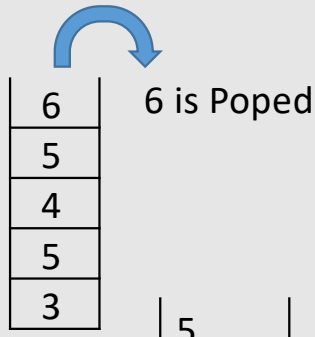
- Pop a vertex from the stack
- If Poped vertex is in initial state visit it and change the state from initial state to visited state.

Here 6 is in initial state so visit 6 and change initial state to visit state

- Push all unvisited vertices adjacent to Poped vertex

Un visited vertex Adjacent to 6 is: 9

- Push 9 into the stack

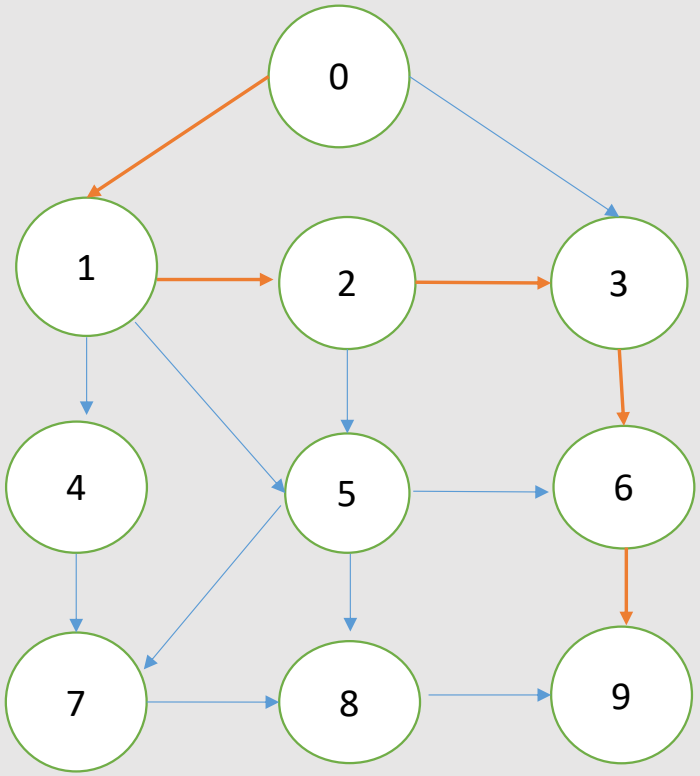


Initial state

4	5	6	7	8	9

Visited state

0	1	2	3	6				



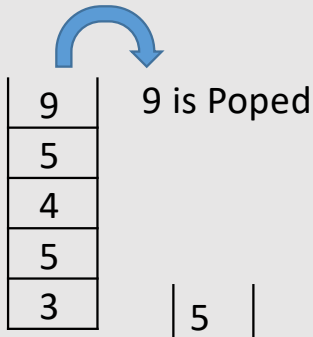
- Pop a vertex from the stack
- If Poped vertex is in initial state visit it and change the state from initial state to visited state.

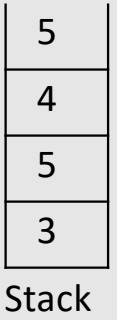
Here 9 is in initial state so visit 9 and change initial state to visit state

- Push all unvisited vertices adjacent to Poped vertex

No Un visited vertices Adjacent to 9 !

- No push



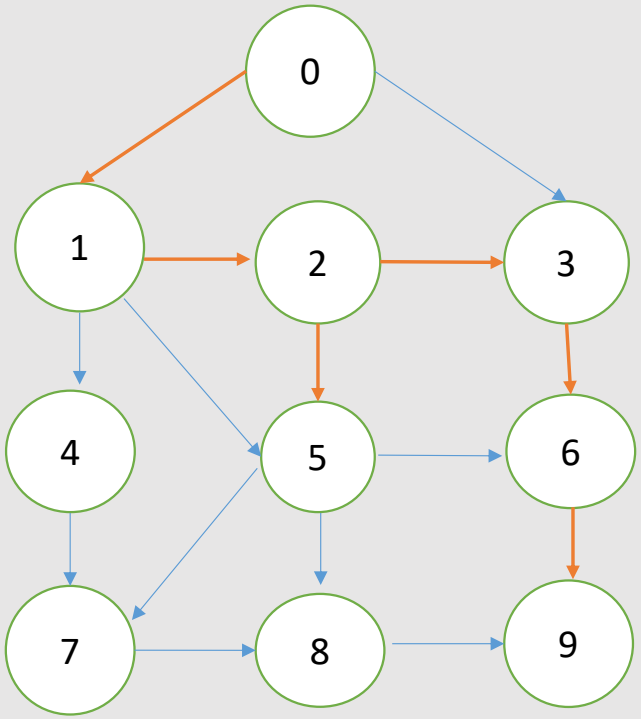


Initial state

4	5	7	8	9

Visited state

0	1	2	3	6	9			



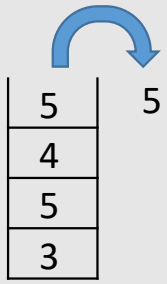
- Pop a vertex from the stack
- If Poped vertex is in initial state visit it and change the state from initial state to visited state.

Here 5 is in initial state so visit 5 and change initial state to visit state

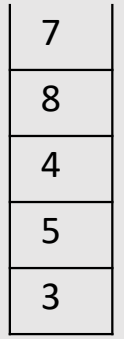
- Push all un visited vertices adjacent to Poped vertex

Unvisited vertices Adjacent to 5 are: 8 and 7 (6 already visited)

- Push 8, 7 into the stack



5 is Poped



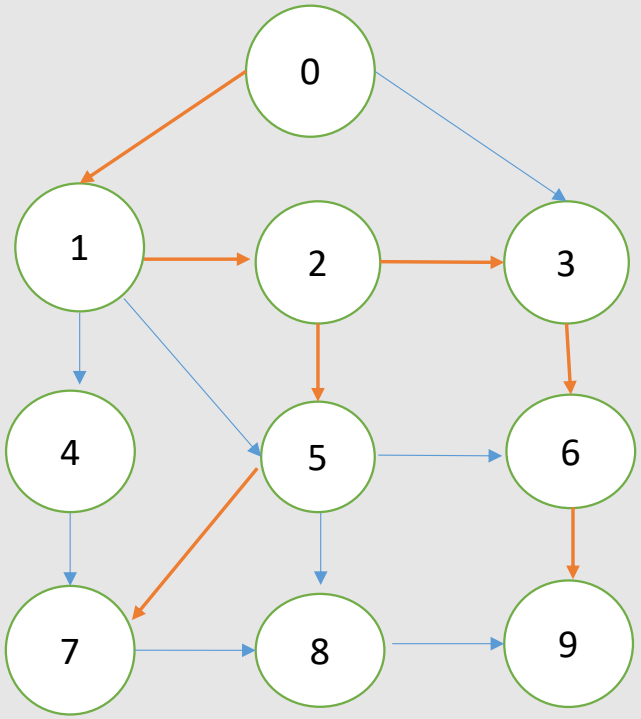
Stack

Initial state

4	5	7	8

Visited state

0	1	2	3	6	9	5		



- Pop a vertex from the stack
- If Poped vertex is in initial state visit it and change the state from initial state to visited state.

Here 7 is in initial state so visit 7 and change initial state to visit state

- Push all un visited vertices adjacent to Poped vertex

Un visited vertex adjacent to 7 is 8

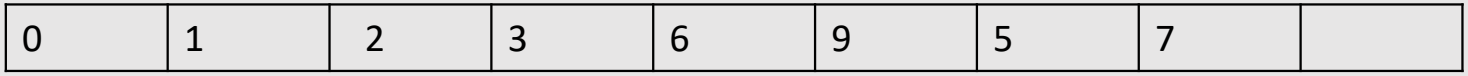
- Push 8 into the stack

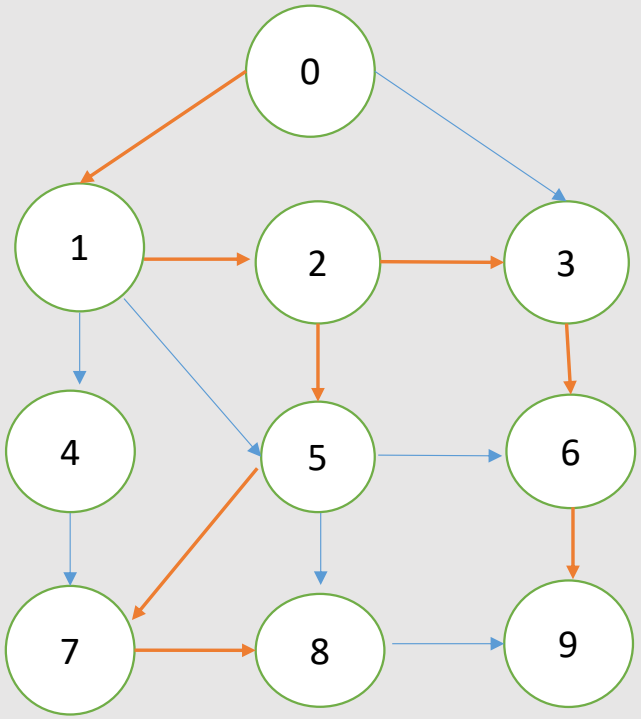
7 is Poped

Stack

Initial state

Visited state





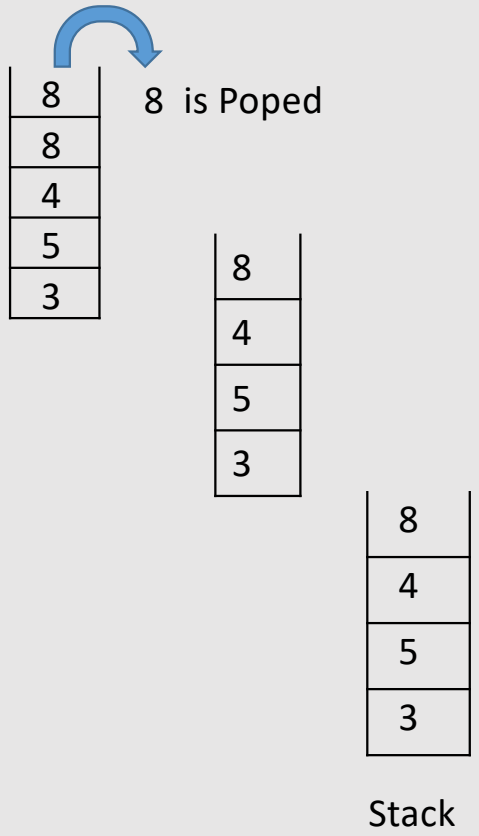
- Pop a vertex from the stack
- If Poped vertex is in initial state visit it and change the state from initial state to visited state.

Here 8 is in initial state so visit 8 and change initial state to visit state

- Push all un visited vertices adjacent to Poped vertex

No unvisited vertices Adjacent to 8

- No Push

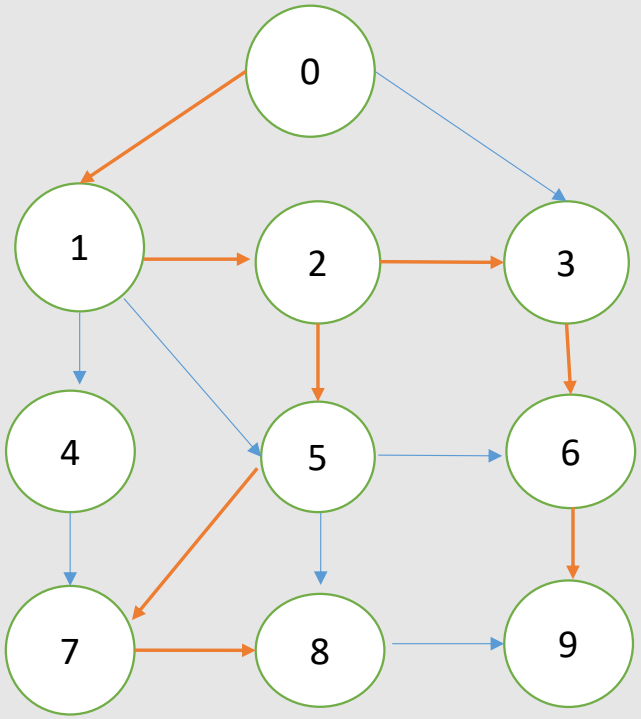


Initial state

4	8

Visited state

0	1	2	3	6	9	5	7	8



❖ Pop a vertex from the stack

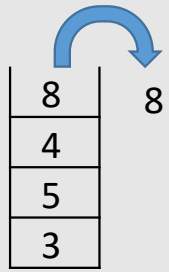
If Poped vertex is in initial state visit it and change the state from initial state to visited state.

Here 8 is in initial state so visit 8 and change initial state to visit state

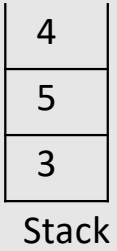
❖ Push all un visited vertices adjacent to Poped vertex

No unvisited vertices Adjacent to 8

❖ No Push



8 is Poped



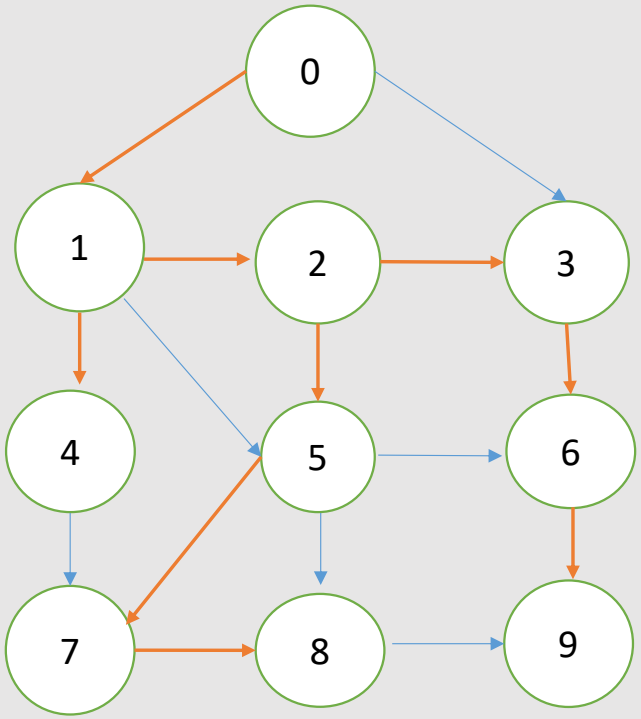
Stack

Initial state

4	8

Visited state

0	1	2	3	6	9	5	7	8



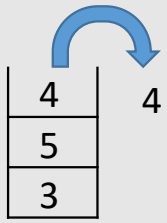
- Pop a vertex from the stack
- If Poped vertex is in initial state visit it and change the state from initial state to visited state.

Here 4 is in initial state so visit 4 and change initial state to visit state

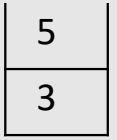
- Push all un visited vertices adjacent to Poped vertex

No unvisited vertices Adjacent to 4

- No push



4 is Poped



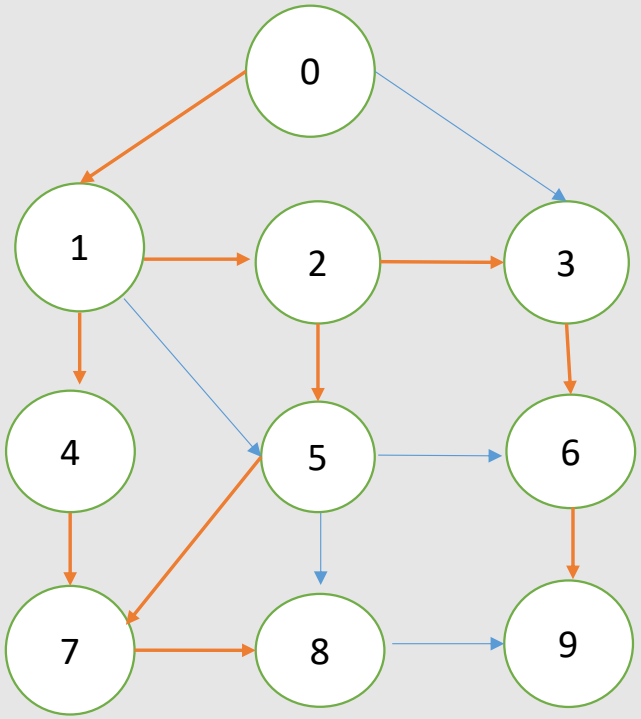
Stack

Initial state

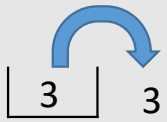
4

Visited state

0	1	2	3	6	9	5	7	8	4



❖ Pop a vertex from the stack



3 is Poped

3 is already in visited state.

❖ Push all un visited vertices adjacent to Poped vertex

No unvisited vertices Adjacent to 3

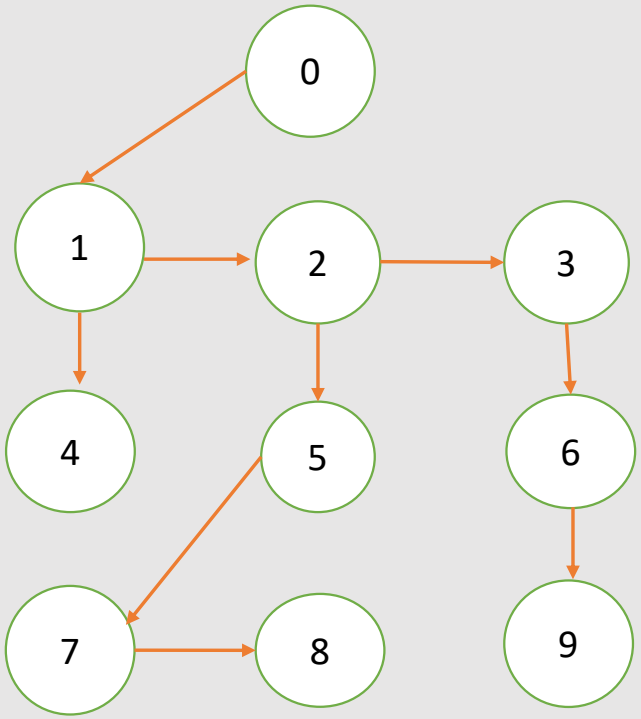
❖ No push

Empty

Stack

Visited state

0	1	2	3	6	9	5	7	8	4



== Depth First Search Tree

Stack is empty

Traversing is complete All nodes are visited.

Number of edges = n-1 = 10-1 = 9

0	1	2	3	6	9	5	7	8	4

Traversing nodes