

①

Name	Manahil Khan
ID	14525
Program	BSCS
Semester	4 th
Subject	Design & Analysis of Algorithms
Instructor	Sir Muhammad Adil
Department	Computer Sci
Date	24/6/20

* Attempt all tasks

QNO#1

* fill in the blanks:-

- 1) A vertex is a junction where something take place in graph.
- 2) Nodes that share the same edge are called Multiple / parallel edges
- 3) Two edges that are incident on same node are called Adjacent edges
- 4) A path between two nodes covering minimum of nodes is called Simple path
- 5) A closed path with more than three edges is called cycle.
- 6) A node with zero ⁱⁿ⁻degree is called Source Node

(2)

7) A ~~node~~ node with zero out-degree is called Sink

8) Isolated/Null graph is a graph with no pair vertices having a common edge

9) Regular graph is a graph where each node is of the same degree.

10) Labeled graph is a graph where each edge is assigned a title.

QNO#2

① $D - Y * (F/G)$

* pre-fix

$D - Y * (F/G)$

$- D Y * (F/G)$

$- D * Y (F/G)$

$- D * Y (F/G)$

post fix

$D - Y * (F/G)$

$D Y * (F/G) -$

$D Y (F/G) * -$

$D Y (F/G) * -$

② $T/W^R + S * M - Y^K$

* pre-fix

$T/W^R + S * M - Y^K$

$+ T/W^R S * M - Y^K$

$+ / T W^R - S * M Y^K$

(3)

$$+ / T^w R - * S M^y K$$

* post-fix

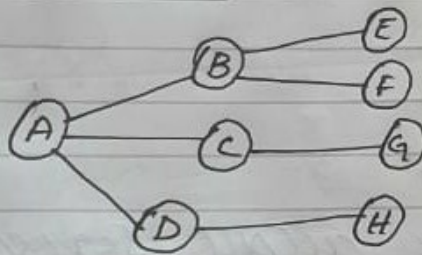
$$T / W^R + S * M - Y^K$$

$$T / W^R \quad S * M - Y^K +$$

$$T W^R / S * M Y^K - +$$

$$T W R^ / S M * Y K^ - +$$

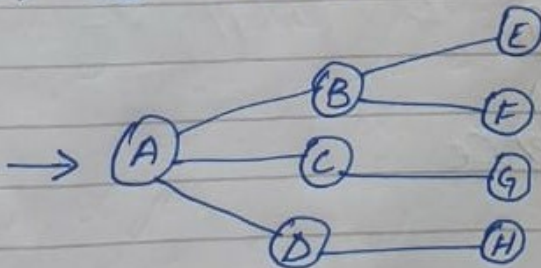
QNO#3



① Add Root A to the output Sequences.

* Mark A visited

* A IS CWN.



* output Sequence
A

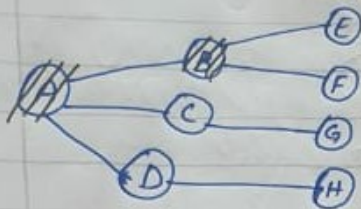
② * A is adjacent to B, C & D.

* Select B & push A into queue.



(4)

- * Add B to the output Sequence.
- * Mark B visited.



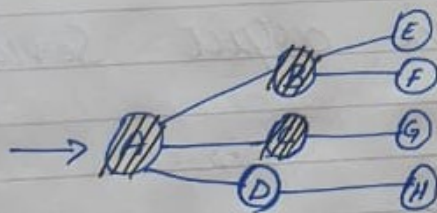
* Output Sequence
A, B

③ from CWN i.e 'A' the adjacent node is 'C'

- * 'C' is pushed into the Queue



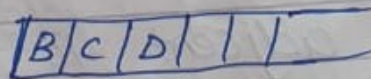
- * C is marked visited
- * C is added to the output Sequence



* Output Sequence
A, B, C

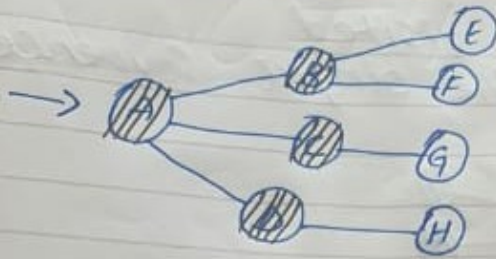
④ AS 'D' is also adjacent to 'A'.

- * D is pushed into the Queue



- * D is marked visited
- * D is added to the output Sequence.

(5)



* Output Sequence
A, B, C, D

* NOW CWN is updated.

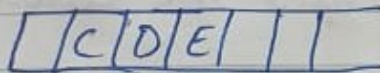
* 'B' is Selected as now CWN.

* 'B' is popped from Queue.



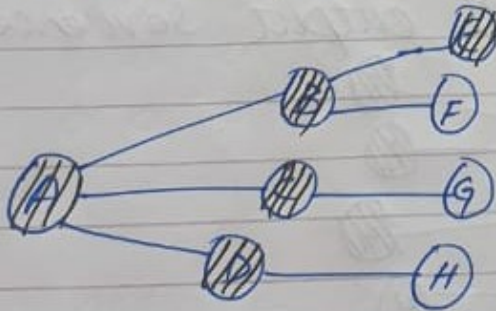
(5) A B is adjacent to E & F.

* 'E' is Selected & pushed into the Queue.



* 'E' is marked visited

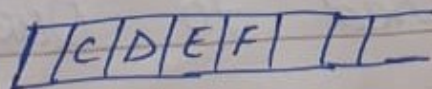
* 'E' is added to the output Sequence.



* Output Sequence
A, B, C, D, E

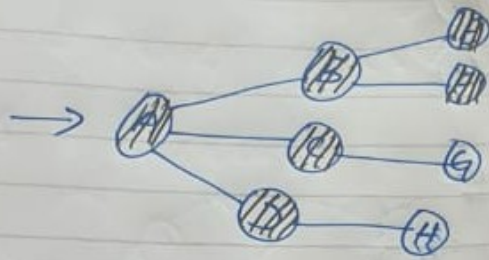
(6) * from CWN i.e 'B' the adjacent node 'F' is Selected.

* 'F' is pushed into the Queue



6

- * 'F' is marked visited
- * F is added to the output sequence

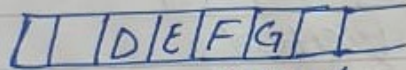


* output sequence
A, B, C, D, E, F

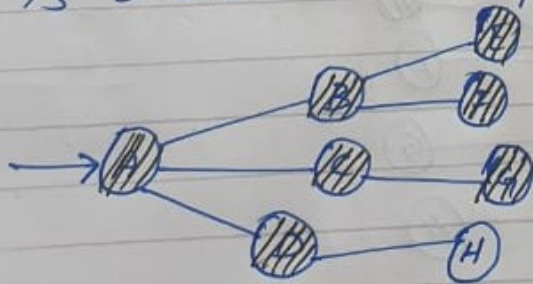
- * NOW CWN is updated to 'C'.
- * 'C' is popped from Queue.



- ⑦ *
- from cwn is 'c' the adjacent node is 'G'
 - * G is pushed into the Queue



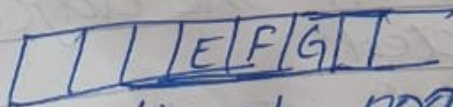
- * G is marked visited
- * 'G' is added to output sequence



* Output Sequence

A, B, C, D, E, F, G

- * NOW CWN is updated to 'D'
- * D is popped from Queue

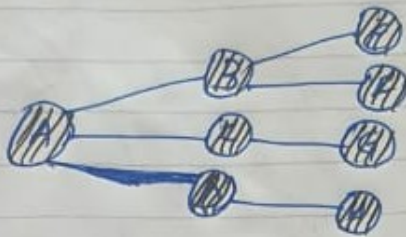


- ⑧ *
- 'H' is adjacent node to 'D'
 - * 'H' is pushed to Queue.

(7)



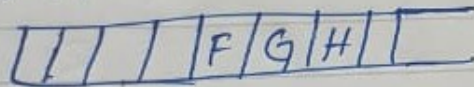
- * 'H' is marked visited
- * 'H' is added to the output Sequence



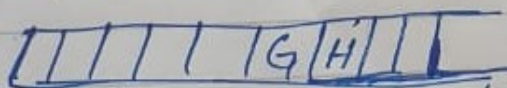
* Output Sequence

A, B, C, D, E, F, G, H

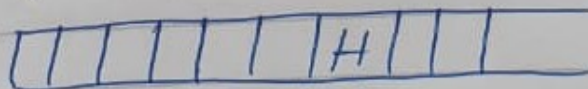
- * Now CWN is updated to 'E'
- * 'E' is popped from Queue



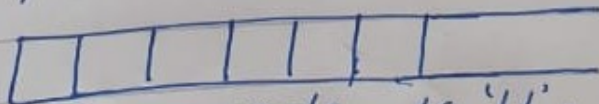
- * No adjacent node to 'F'.
- * Now again CWN is updated to F.
- * F is popped from Queue.



- * No adjacent node to 'F'
- * Now again CWN is update to 'G'.
- G is popped from Queue



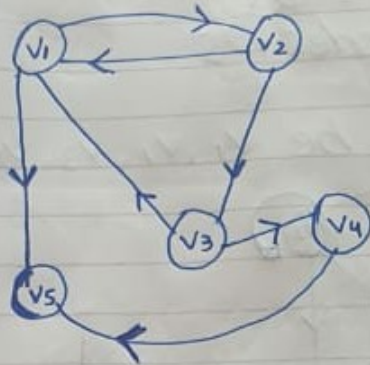
- * No adjacent node to 'G'
- * Now again CWN is updated to 'H'.
- * 'H' is popped from Queue



- * No adjacent node to 'H'.
- * Queue is empty, so BFS Steps.

8

Q NO #4



No of nodes = $m = 5$

order of $A = m \times m$
 $= 5 \times 5$
 $= 25$

	v_1	v_2	v_3	v_4	v_5
v_1	0	1	0	0	1
v_2	1	0	1	0	0
v_3	1	0	0	1	0
v_4	0	0	0	0	1
v_5	0	0	0	0	0

out degree

2
2
2
1
0
<hr/>
7

Indegree: 2 1 1 1 2

Q NO #5

$A[] =$

0	1	0	1	1
1	1	1	0	0
0	0	1	1	0
1	1	0	1	0
0	0	0	0	1

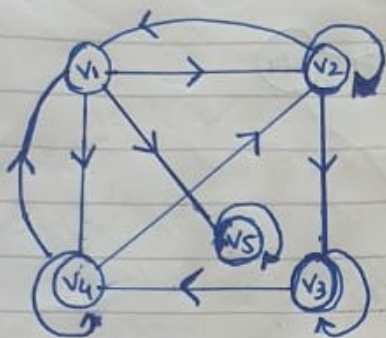
9

AS

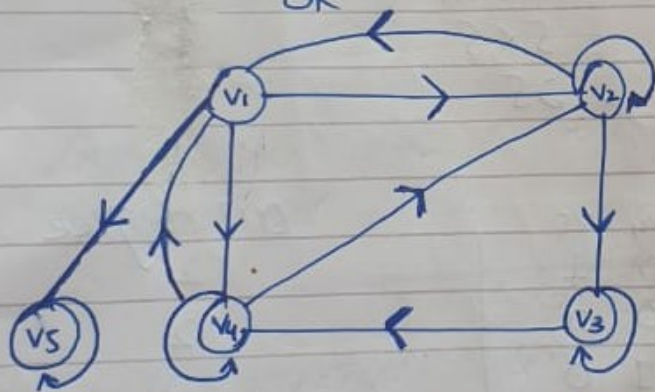
$$\begin{aligned} \text{Order of } A &= m \times m \\ &= 5 \times 5 \\ &= 25 \end{aligned}$$

So no of nodes = 5

lets the nodes be v_1, v_2, v_3, v_4, v_5



OR



The required graph