CS & IT ENGINEERING

Algorithm

Graph Algorithms

Lecture No.- 02

By- Dr. Khaleel Khan sir

Recap of Previous Lecture

Topic DFS in Undirected Connected Graphs

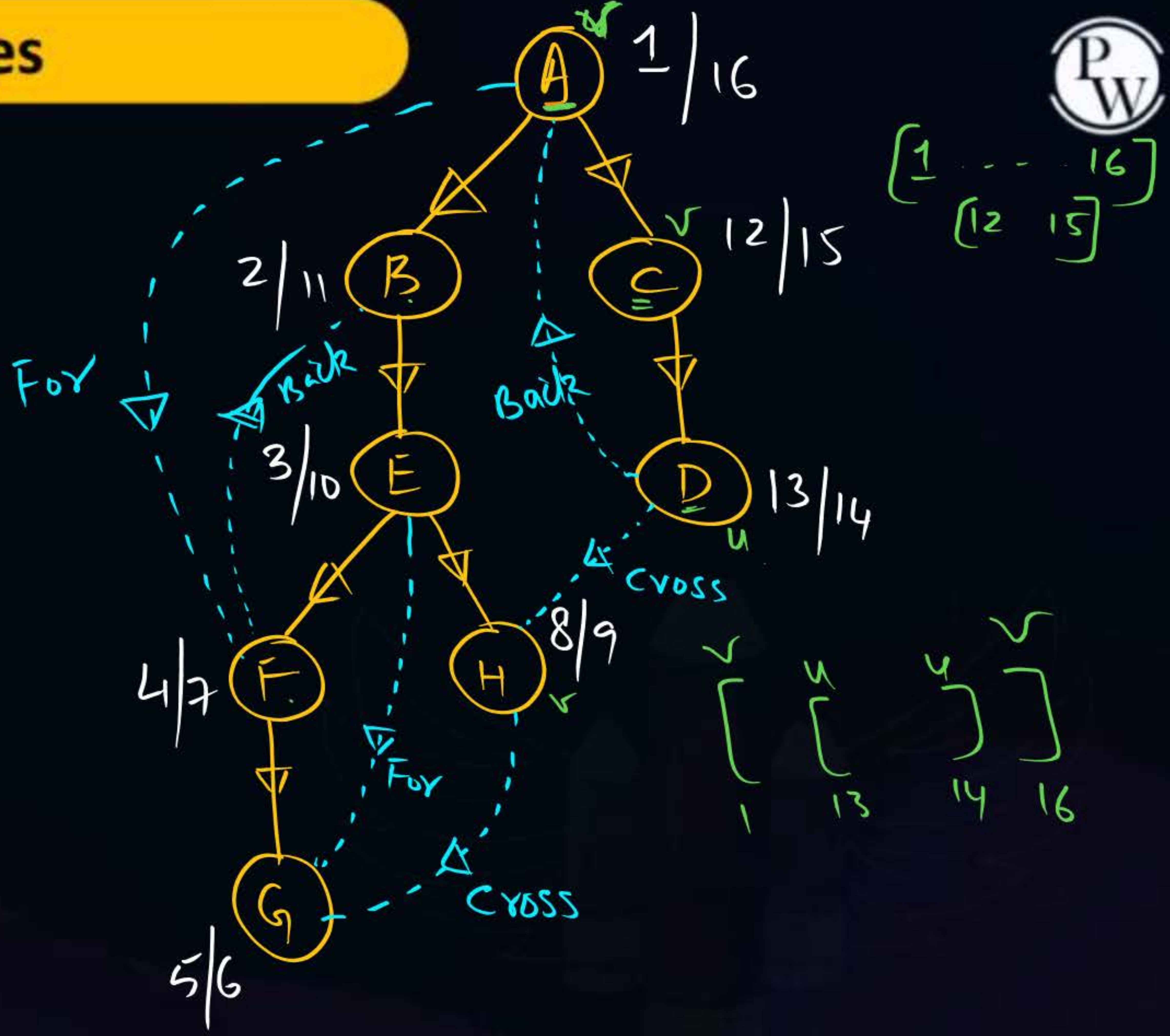
Topic

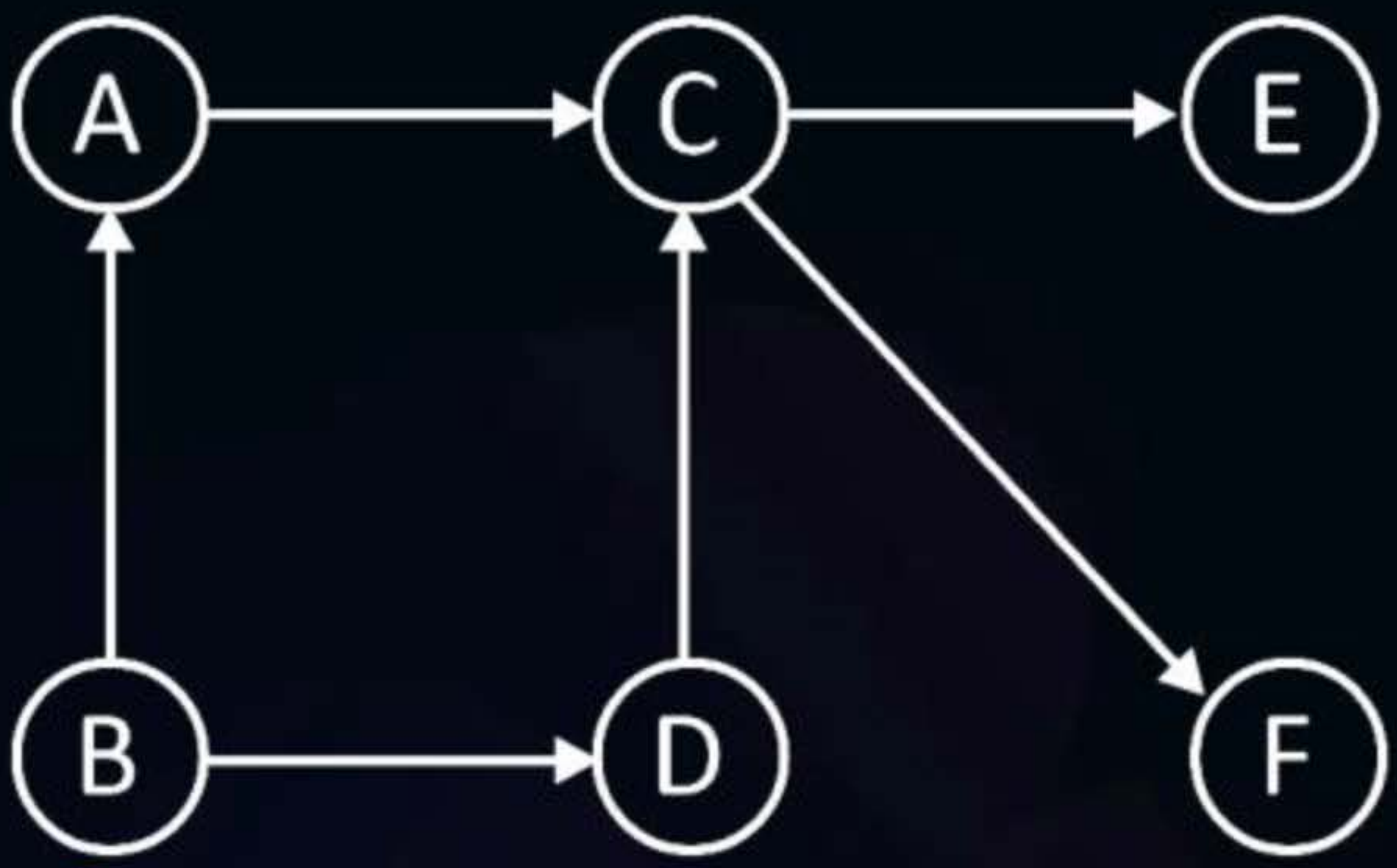
Topics to be Covered

Topic Depth First Search in Directed Graphs

Topic BFS

Topic : Graph Techniques





DFS in D.A.G
<Directed Acyclic Graph>

Topological Sort :

<Linear order of the vertices, repr. the activities maintaining precedences,

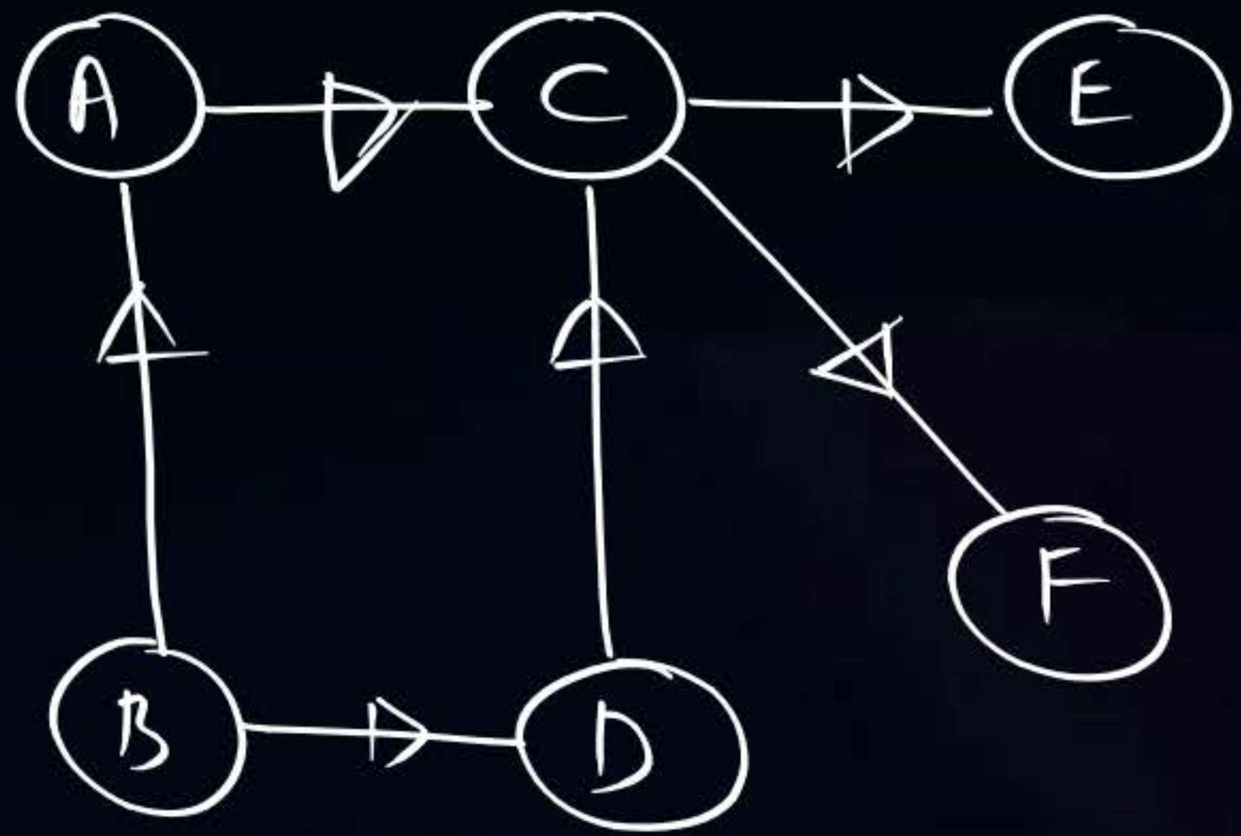
B — A — D — C — E — F — 1

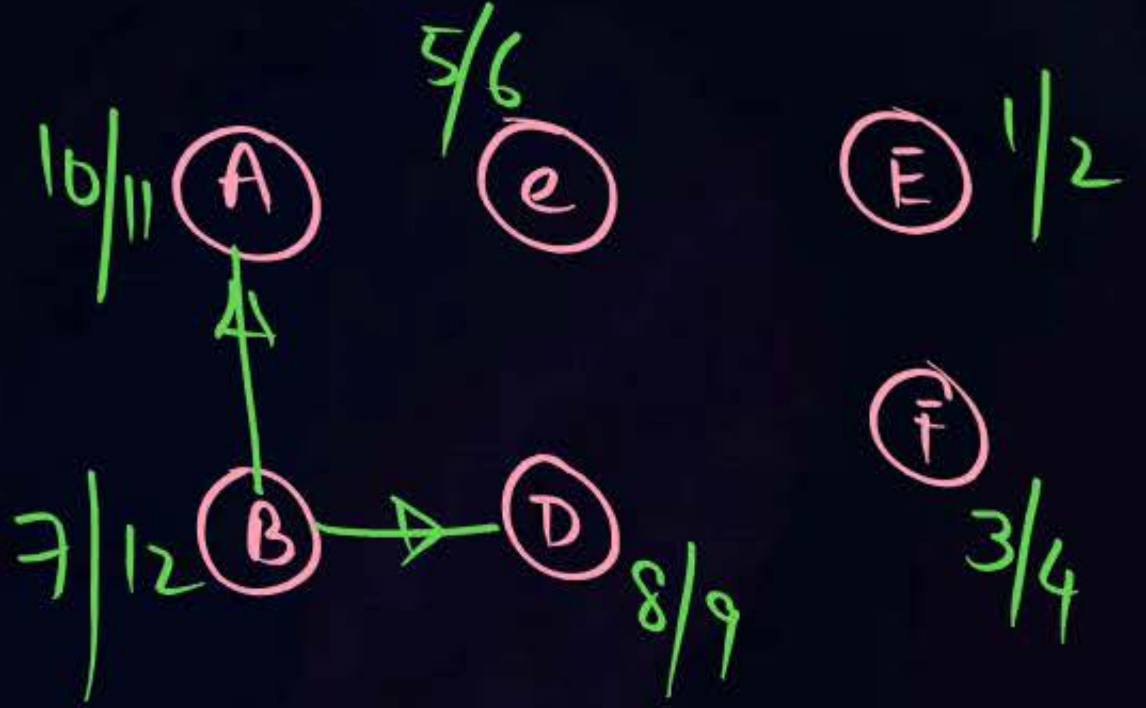
B — A — D — C — F — E — 2

B — D — A — C — E — F — 3

B — D — A — C — F — E — 4

Algorithm for Topological Sort using DFS:





Algo Topo(G)

{

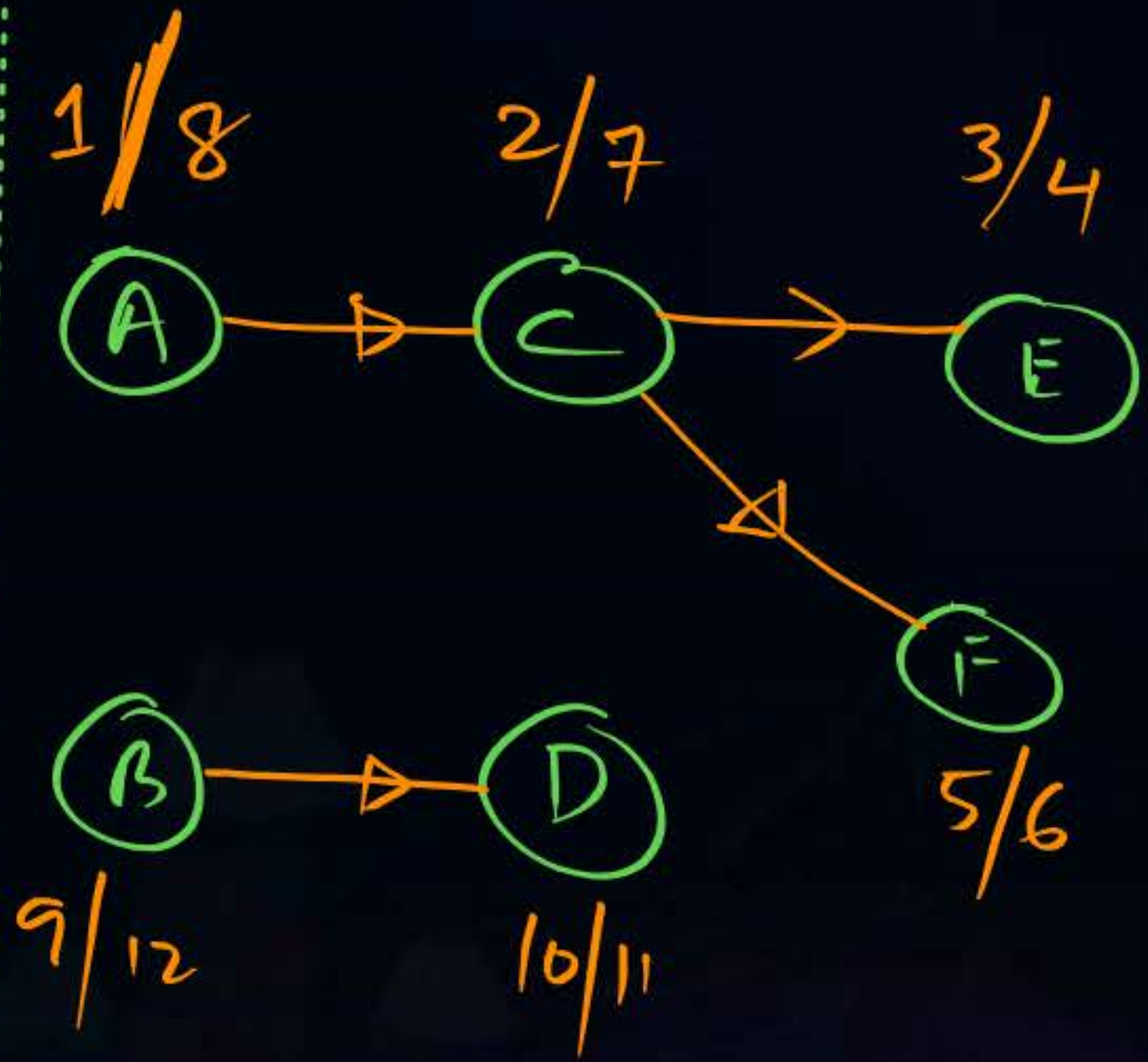
1. DFS(v);

2. Arrange all the Nodes of the Traversal in decreasing order of Finishing Times;

}

B - A - D - C - F - E

DFS Sp. Forest

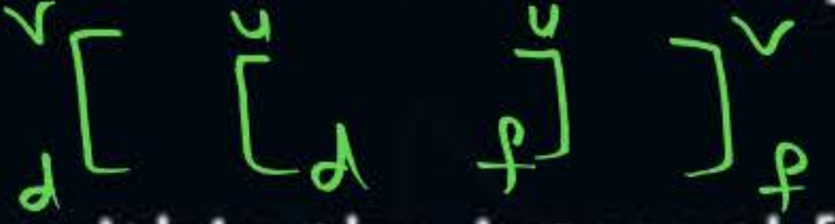


B - D - A - C - F - E

PARENTHESIZATION THEOREM

In any **Depth-First Search** of a (directed or undirected) graph G = (V, E), for any two vertices u and v, having an edge b/w them exactly one of the following three conditions holds:

- I the intervals [d[u], f [u]]and [d[v], f[v]] are entirely disjoint, and neither u nor v is a descendant of the other in the depth-first forest, : CROSS-EDGE
- II the interval [d[u], f[u]] is contained entirely within the interval [d[v], f[v]], and u is a descendant of v in a depth-first tree, or

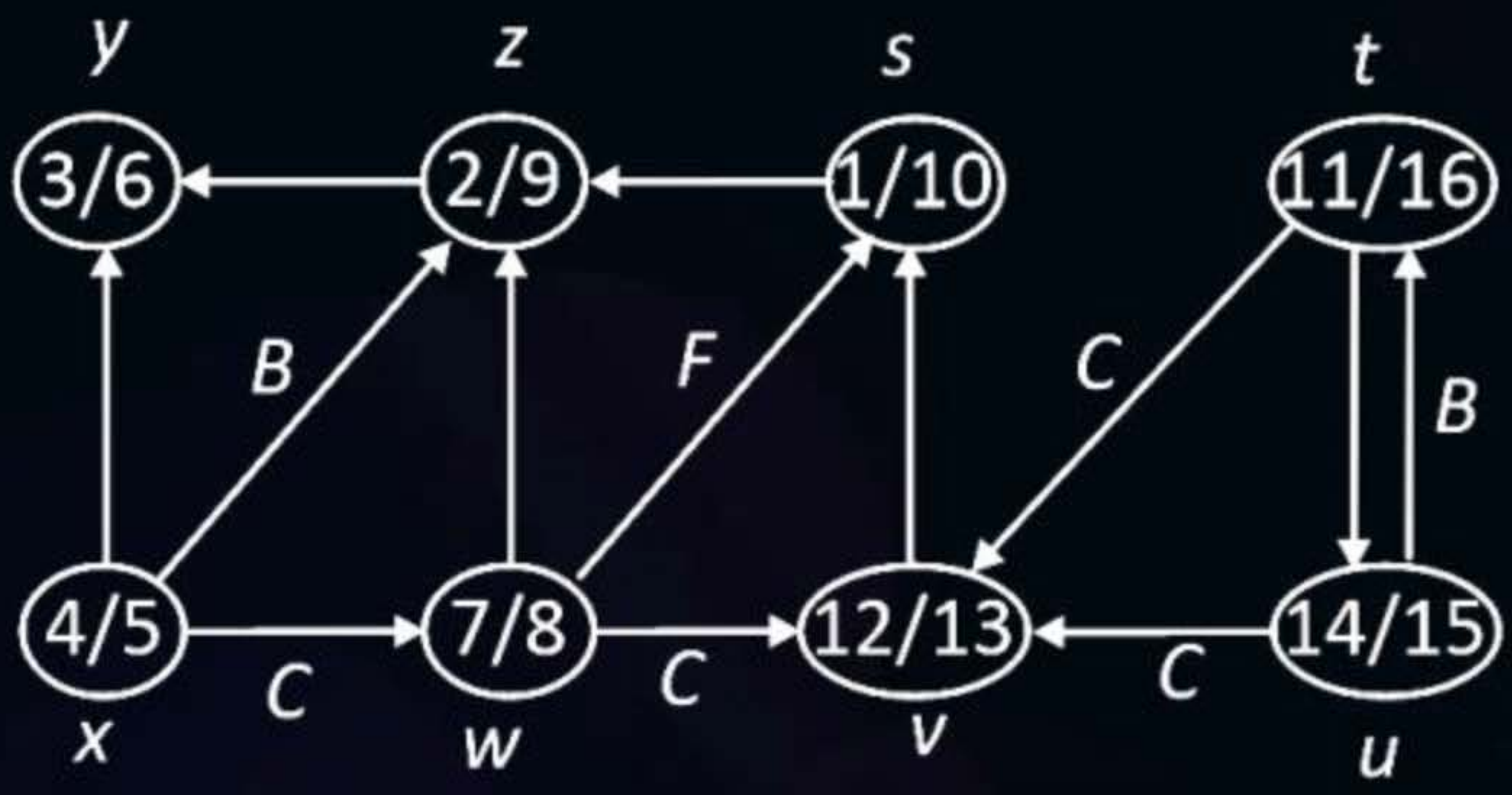


: BACK-EDGE

- III the interval [d[v], f[v]] is contained entirely within the interval [d[u], f[u]], and v is a descendant of u in a depth-first tree.

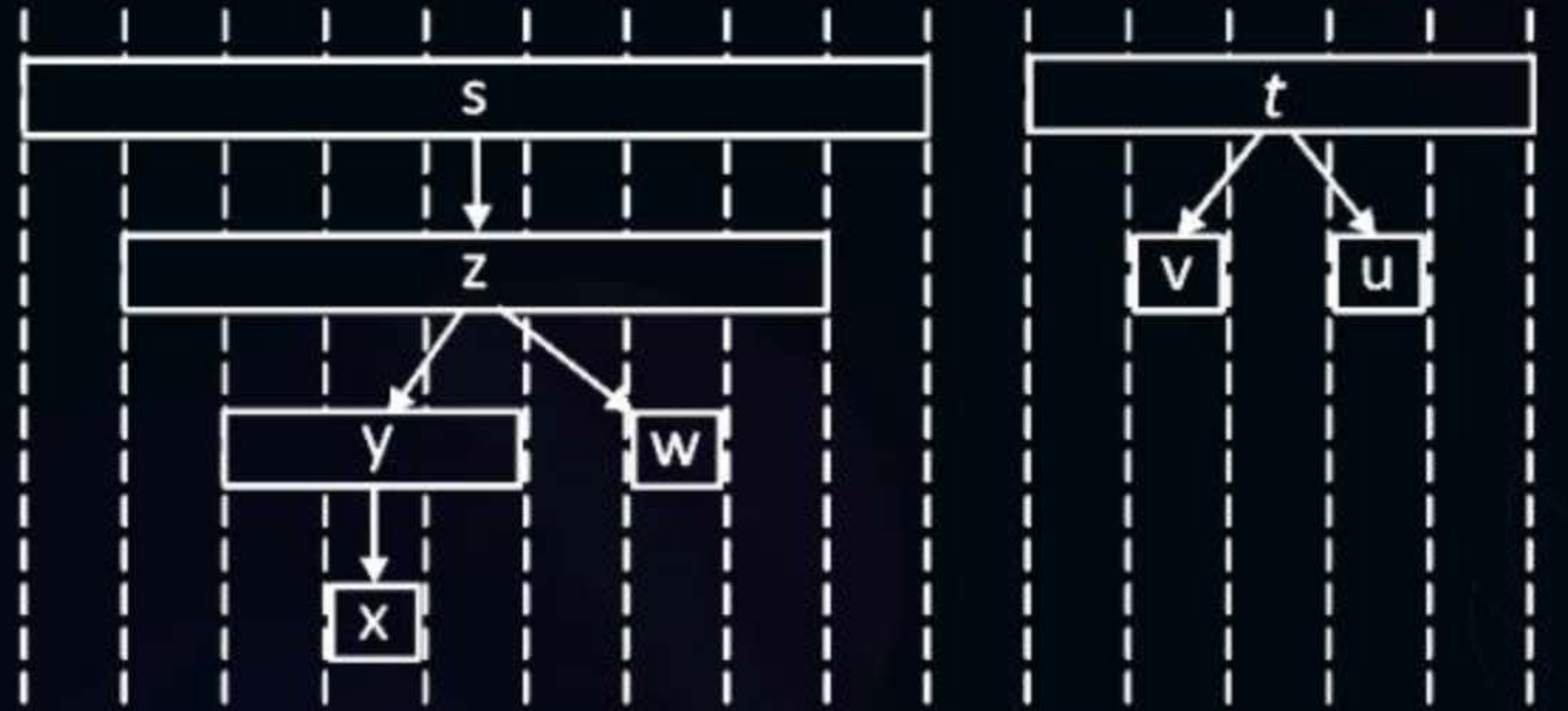
: FORWARD TREE

Topic : Algorithms



Topic : Algorithms

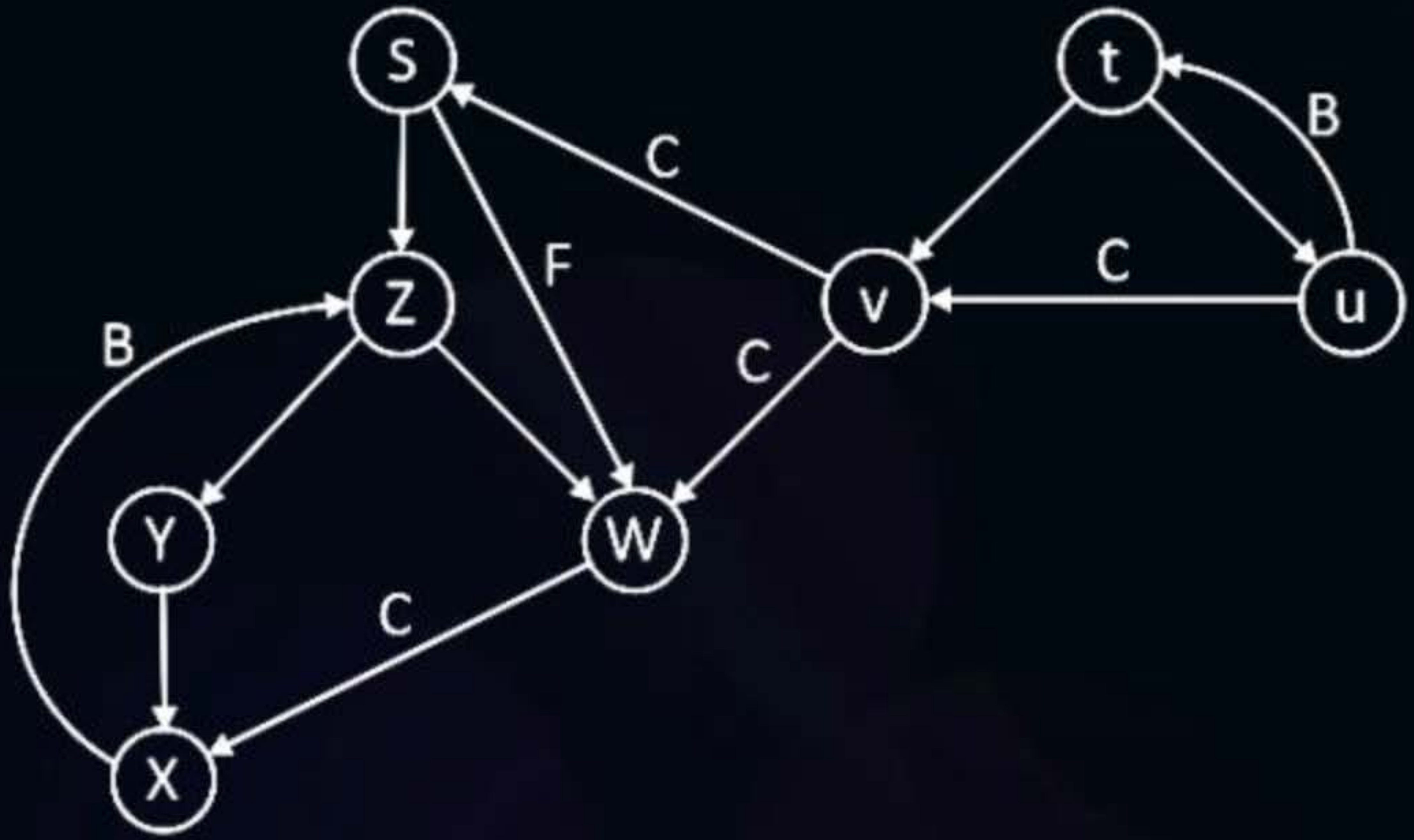
CORMEN



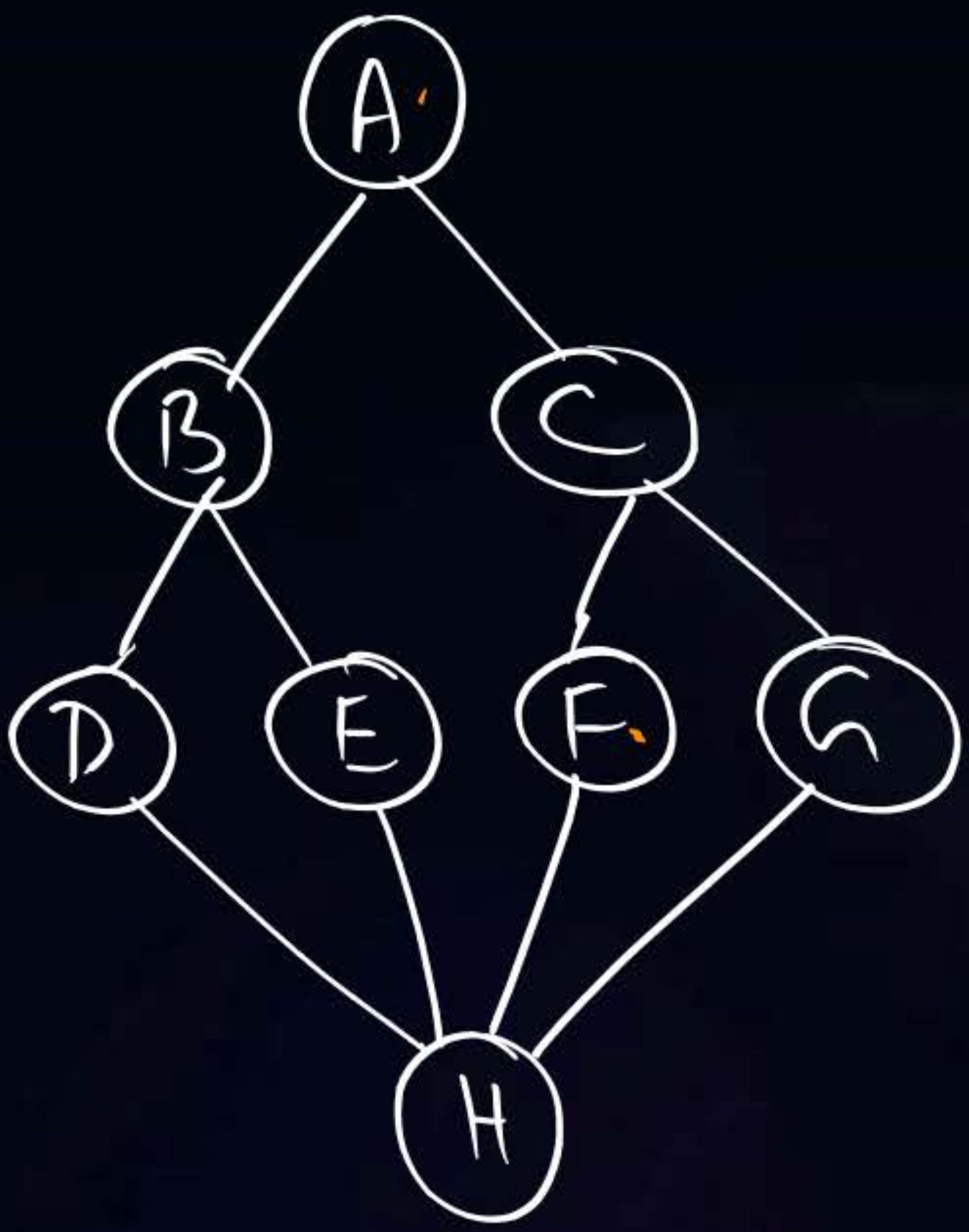
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

(s (z (y (x x) y) (w w) z) s) (t (v v) (u u) t)

Topic : Algorithms



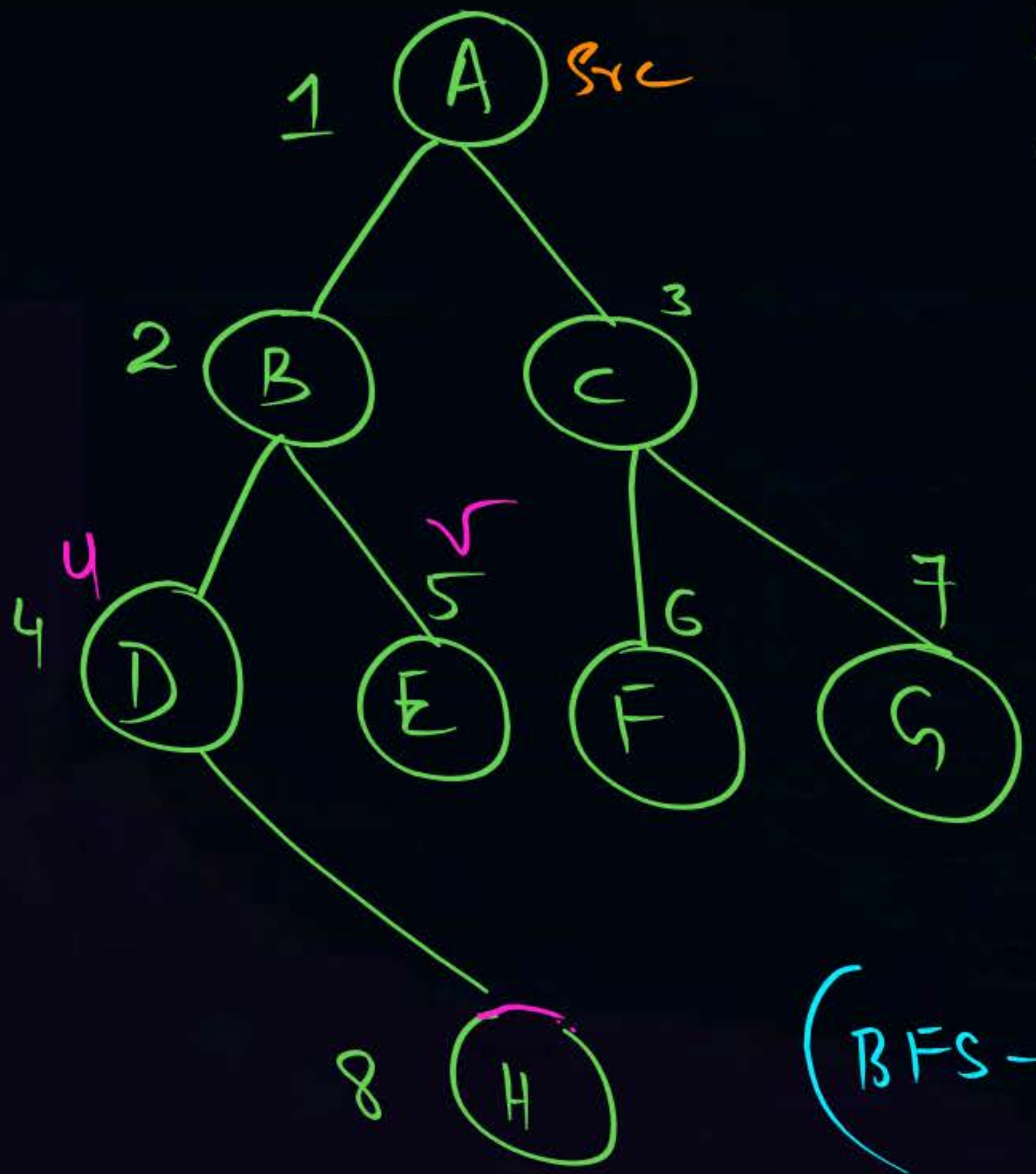
II. Breadth First Search (BFS)



$G = (V, E)$

FIFO-BFS : A; B; C; D; E; F; G; H

$d(r,u) \le d(r,v)$



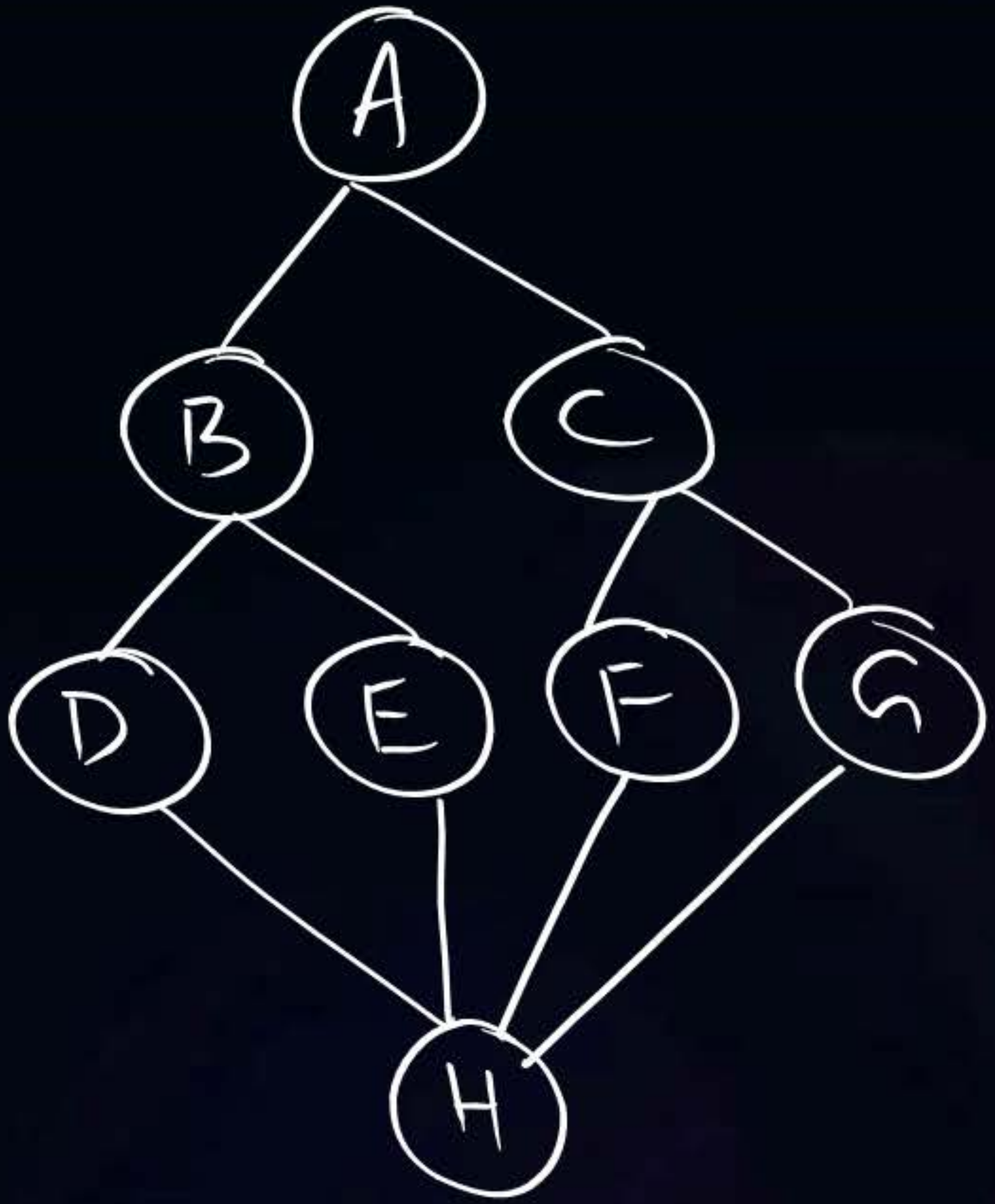
live	~~B~~	~~C~~	~~D~~	~~E~~	~~F~~	~~G~~	~~H~~
Parent	A	A	B	B	C	C	D

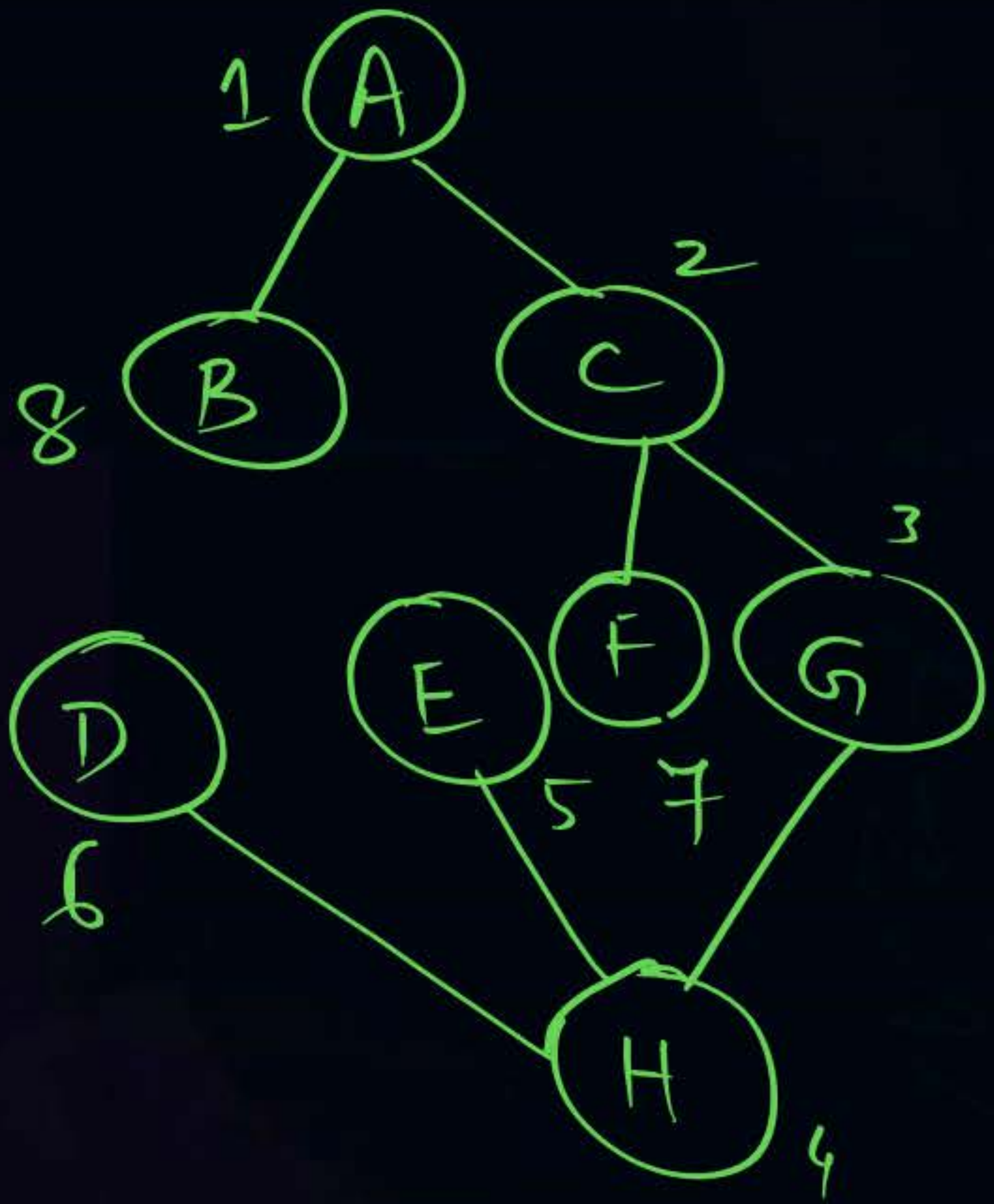
(Single Src. Shortest paths)

(BFS-Spanning Tree)

LIFO - BFS :

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





liv	~~B~~	C	~~F~~	~~G~~	~~H~~	~~D~~	~~E~~	~~F~~
Par	A	A	C	C	G	H	H	~~H~~

Applications of DFS & BFS

1) Time Complexity of DFS & BFS depends on repr. of Graph;
   (i) Adj. Matrix : $O(n^2)$
   (ii) Adj. List : $O(n+e)$

2) Both D.F.S & BFS Can be used to detect the presence of a cycle in the Graph;

3) Both DFS & BFS Can be used to know whether the given graph is Connected or NOT;

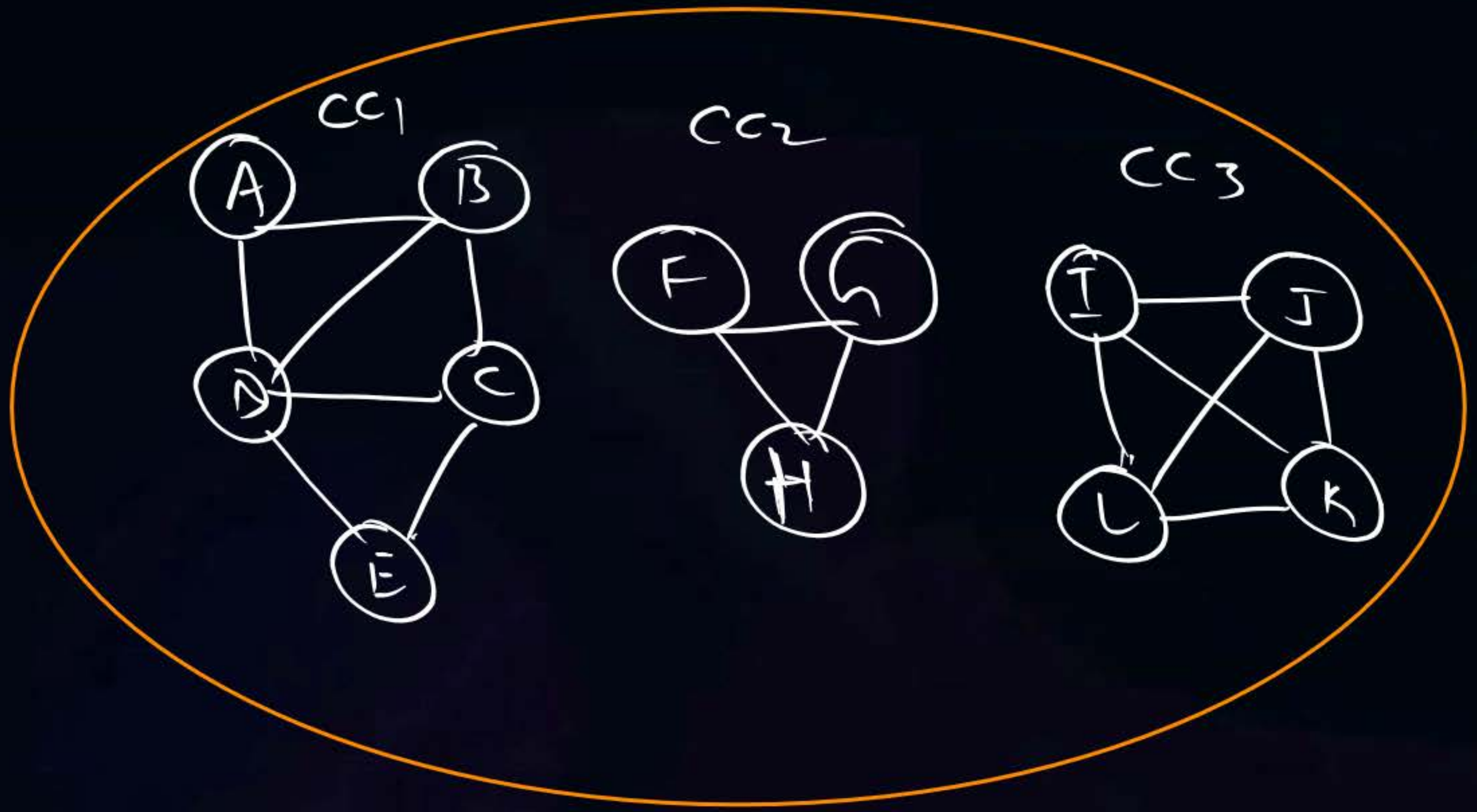
4) Both DFS & BFS Can be used to know whether the two vertices u & v are Connected or Not;

5) DFS is used to determine: Connected Components, Strongly connected ", Biconnected " & Articulation pts;

COMPONENTS

1) **Connected Components :** UNDIRECTED Graphs

↳ Maximal Subgraph that is connected

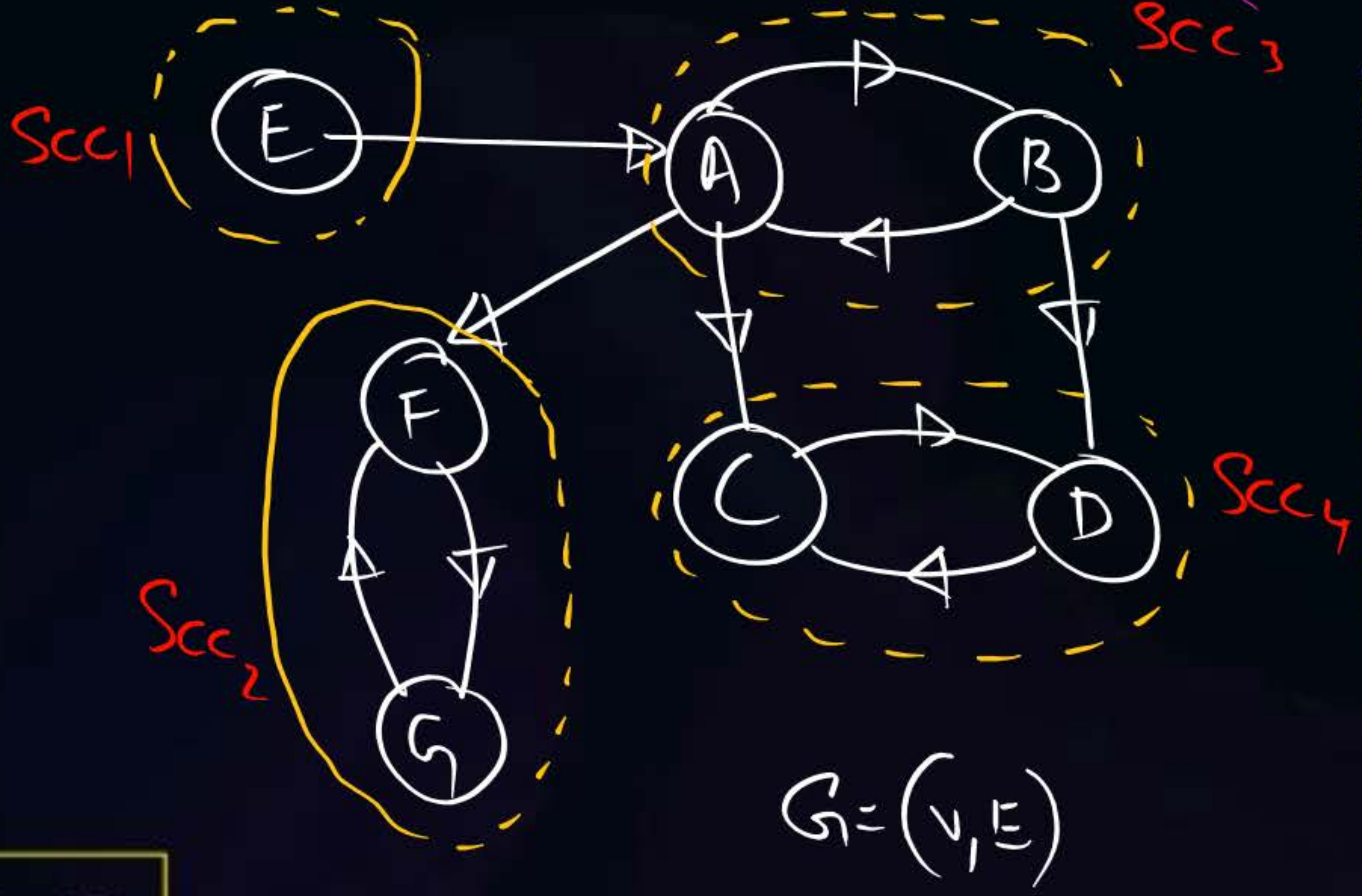


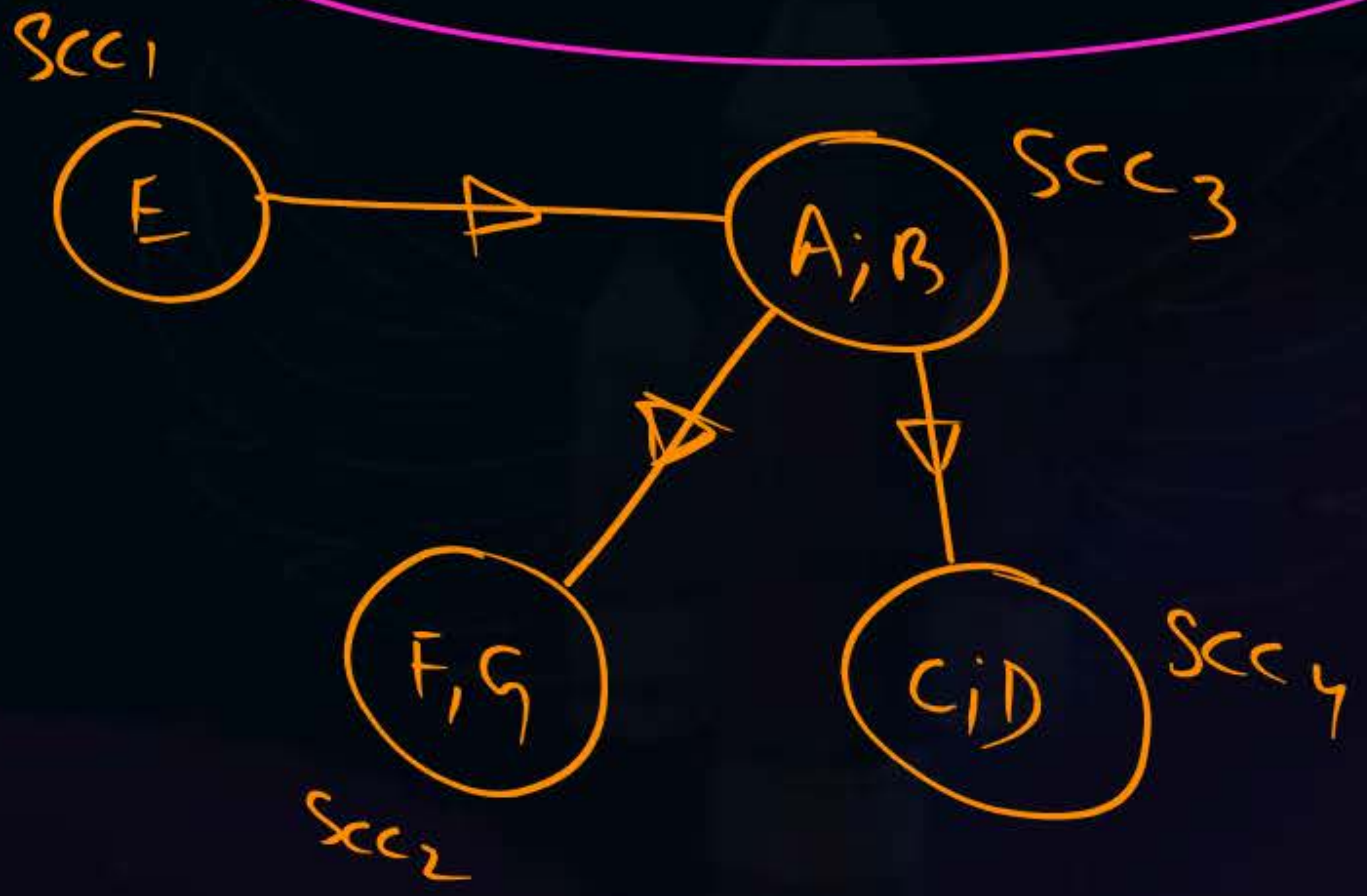
Topic : Strongly Connected Components:

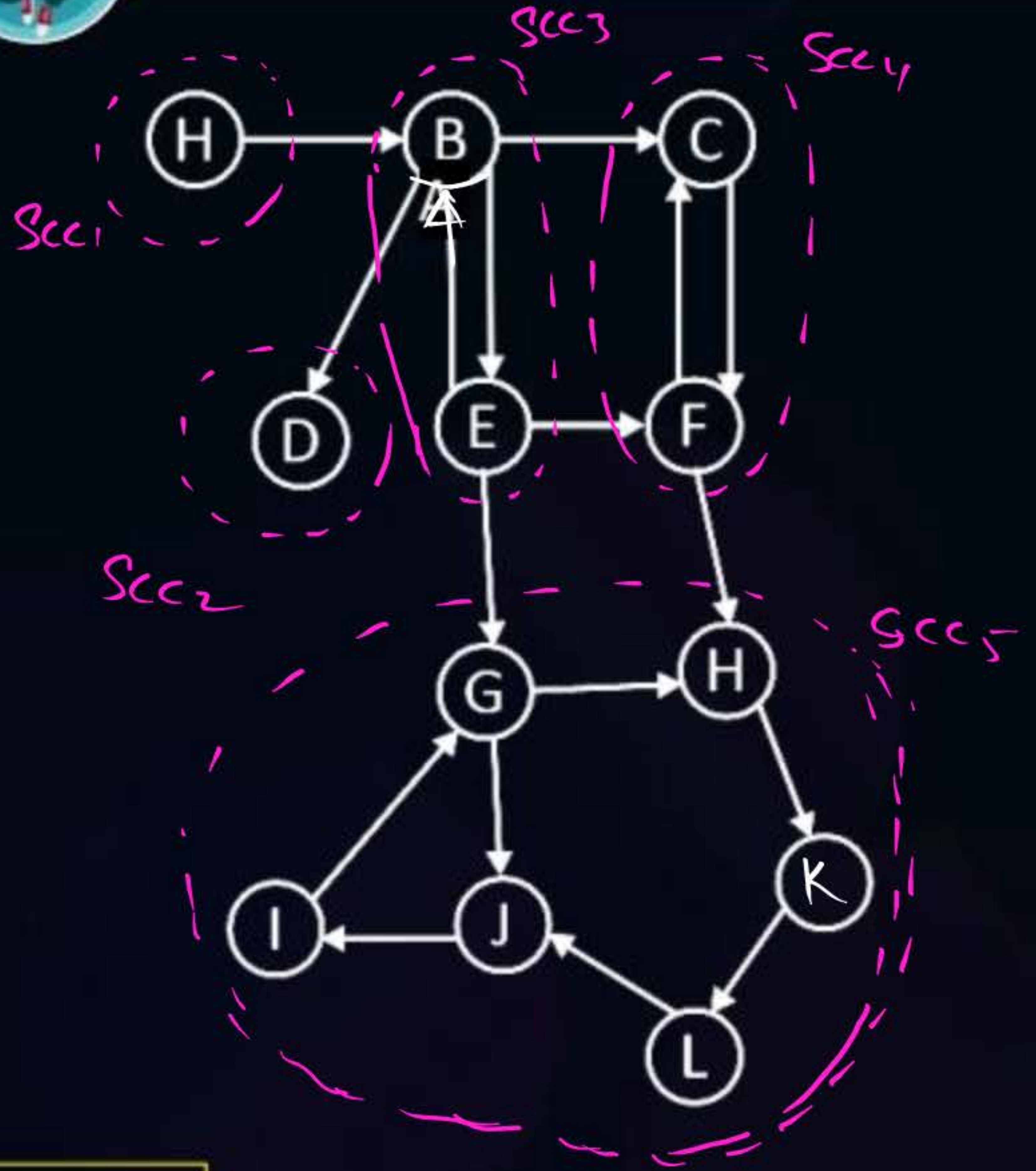
(Directed Graphs)

" Two nodes 'u' and 'v' of a directed graph are connected, if there is a path form 'u' to 'v' and a path from 'v' to 'u; " This relation partitions the vertex of 'V' into disjoint sets known as strongly connected components. (Maximal)

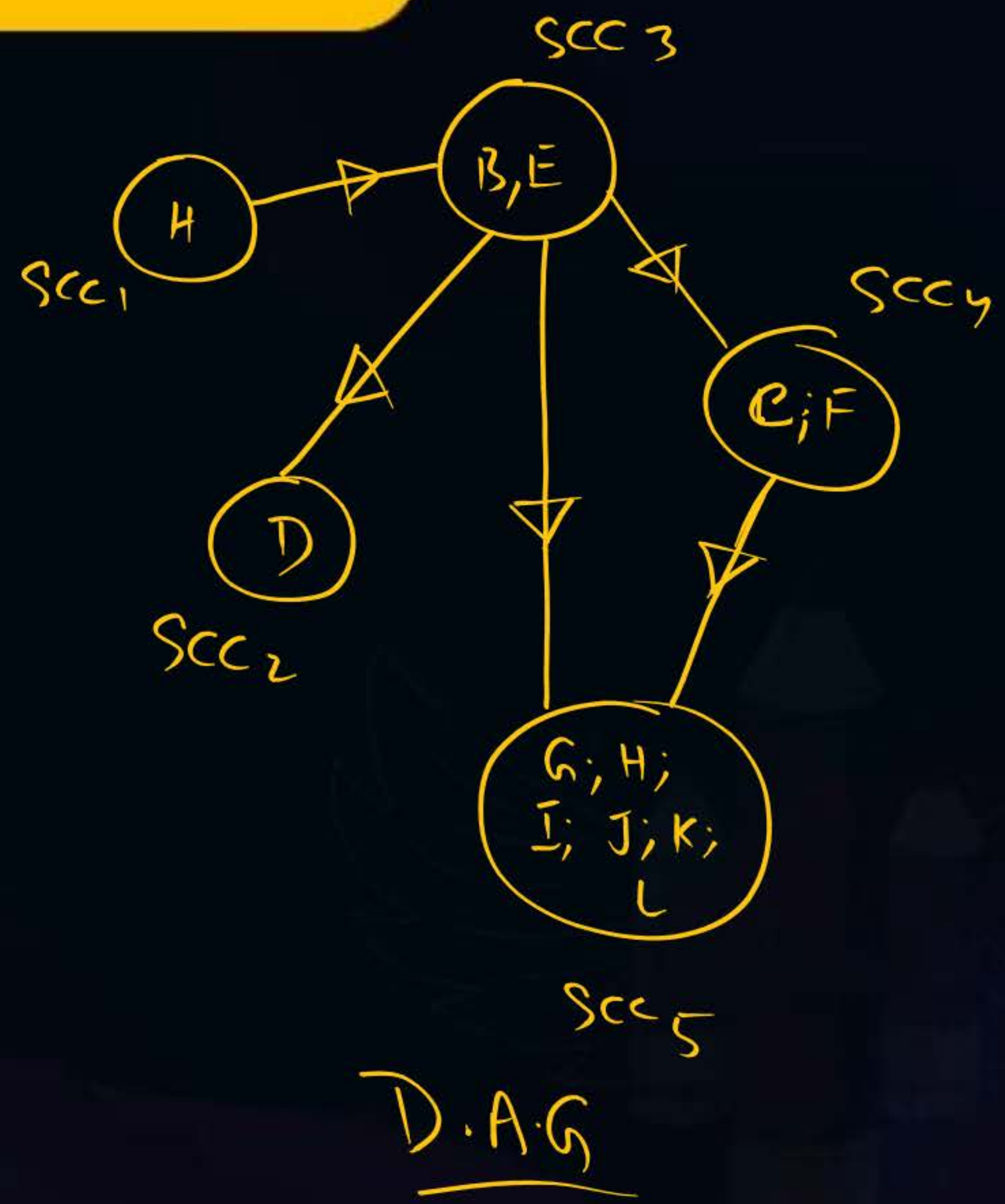
{A, B} {C, D} {F, G} {E}





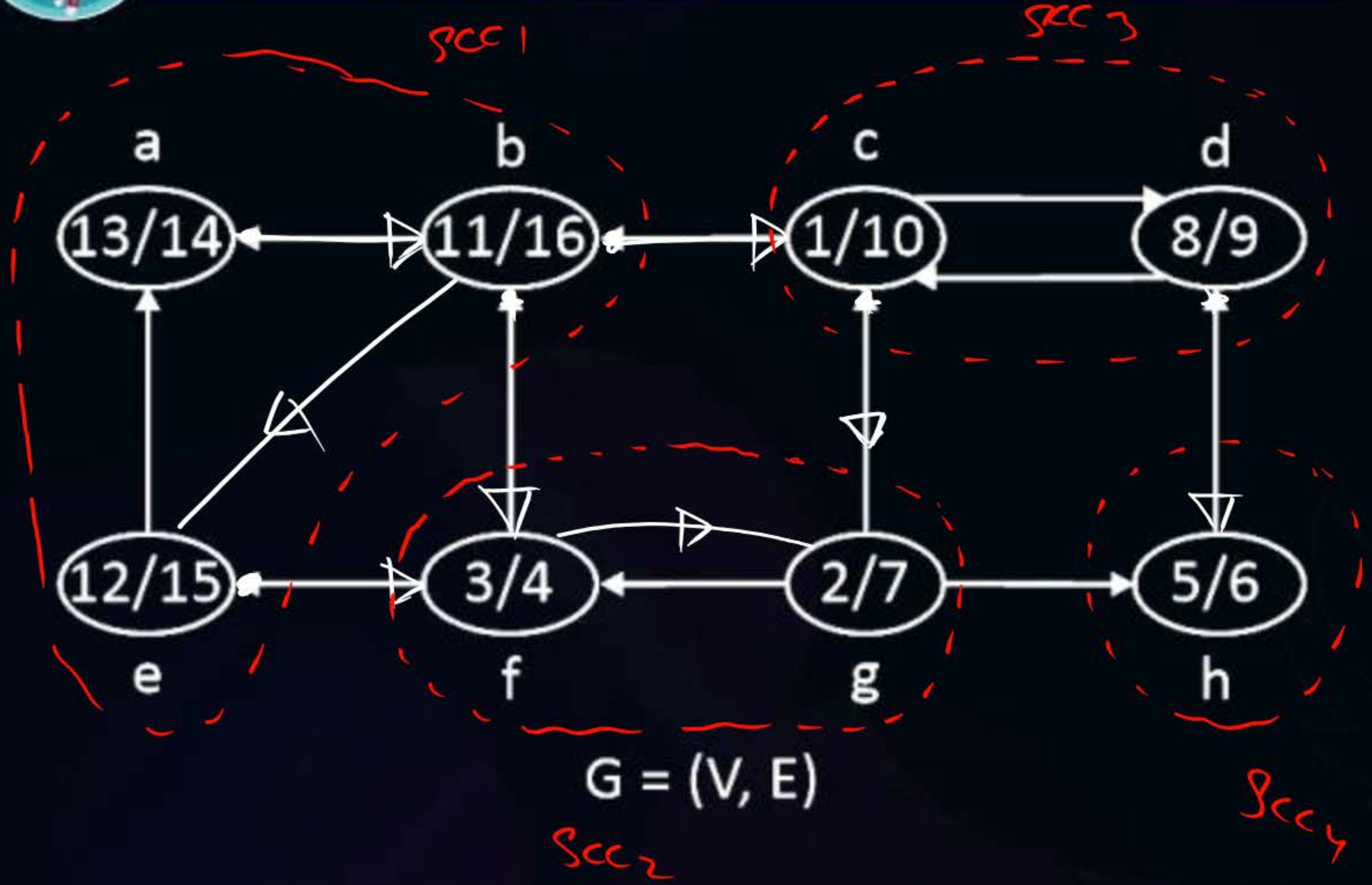


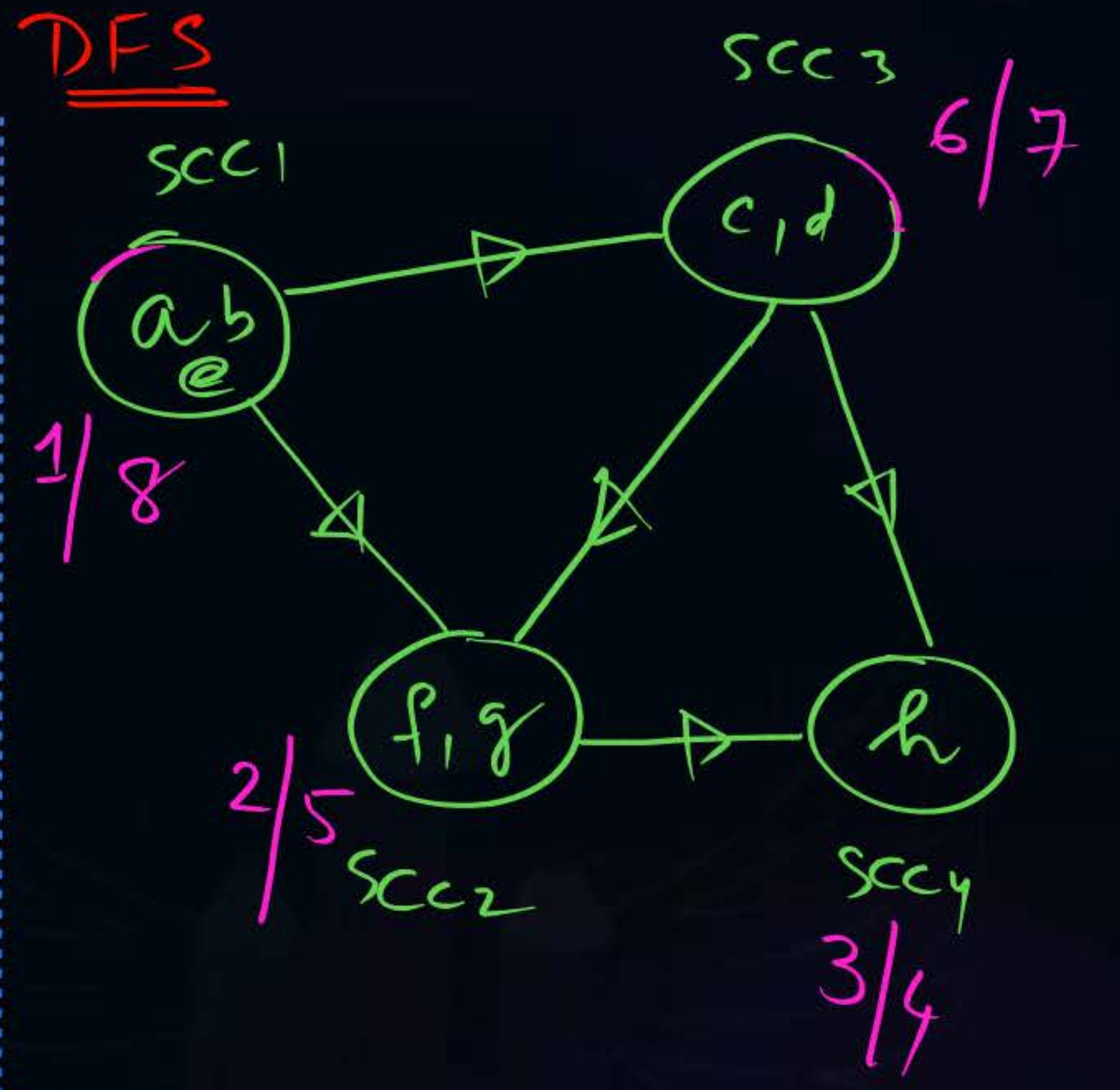
scc3
scc4
scc1
H
B
C
D
E
F
scc2
scc5
G
H
I
J
K
L



SCC 3
H
B,E
Scc1
Sccy
C;F
D
SCC2
G; H;
I; J; K;
L
scc5
D.A.G

Topic : Algorithms





Slide 12

Topic : Strongly Connected Components:

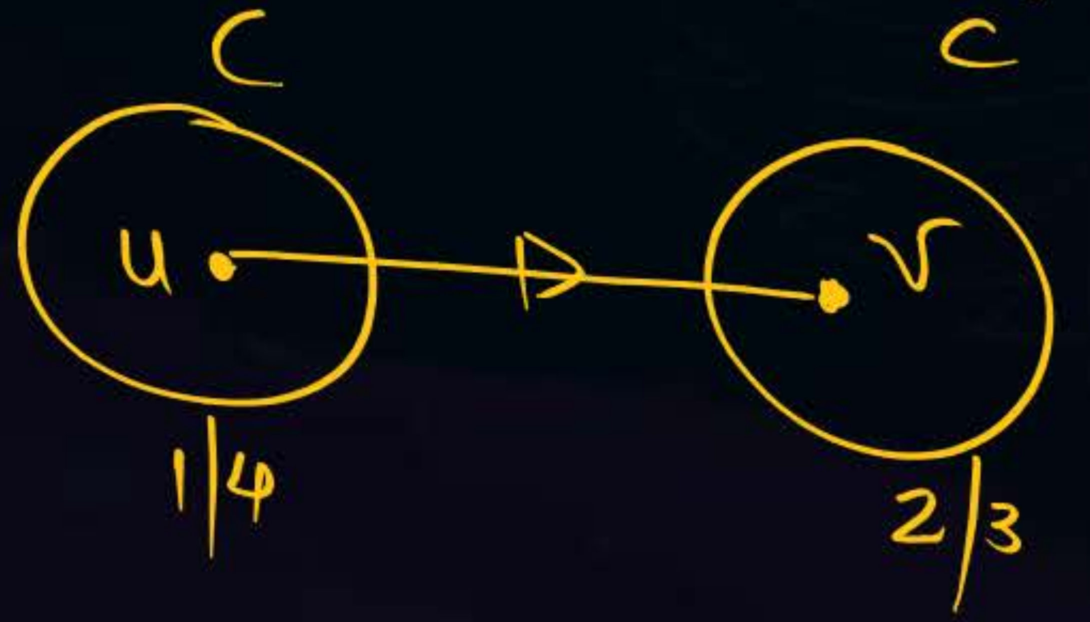
Properties of Strongly C.C

Property 1: Every directed graph is a D.A.G of its strongly connected components.

Property 2: Let C & C^1 be distinct strongly connected components in directed graph $G = (V,E)$, let $u,v \in C$ and $(u^1, v^1) \in C^1$, suppose that there is a path $u \sim u^1$ in G, then there cannot also be a path $v^1 \sim v$ in G.

Property 3 : If 'C' and 'C^1' are strongly connected components of , and there is an edge from a node in C to a node in C^1, then the highest post number in C is bigger than the highest post number in C^1.

→ finishing Time



Articulation Points (Cut vertex) & Biconnected Components

1) **Articulation Point:** is that vertex in the graph, the removal of which along with all its edges, partitions the graph into 2/more non-Empty Components;

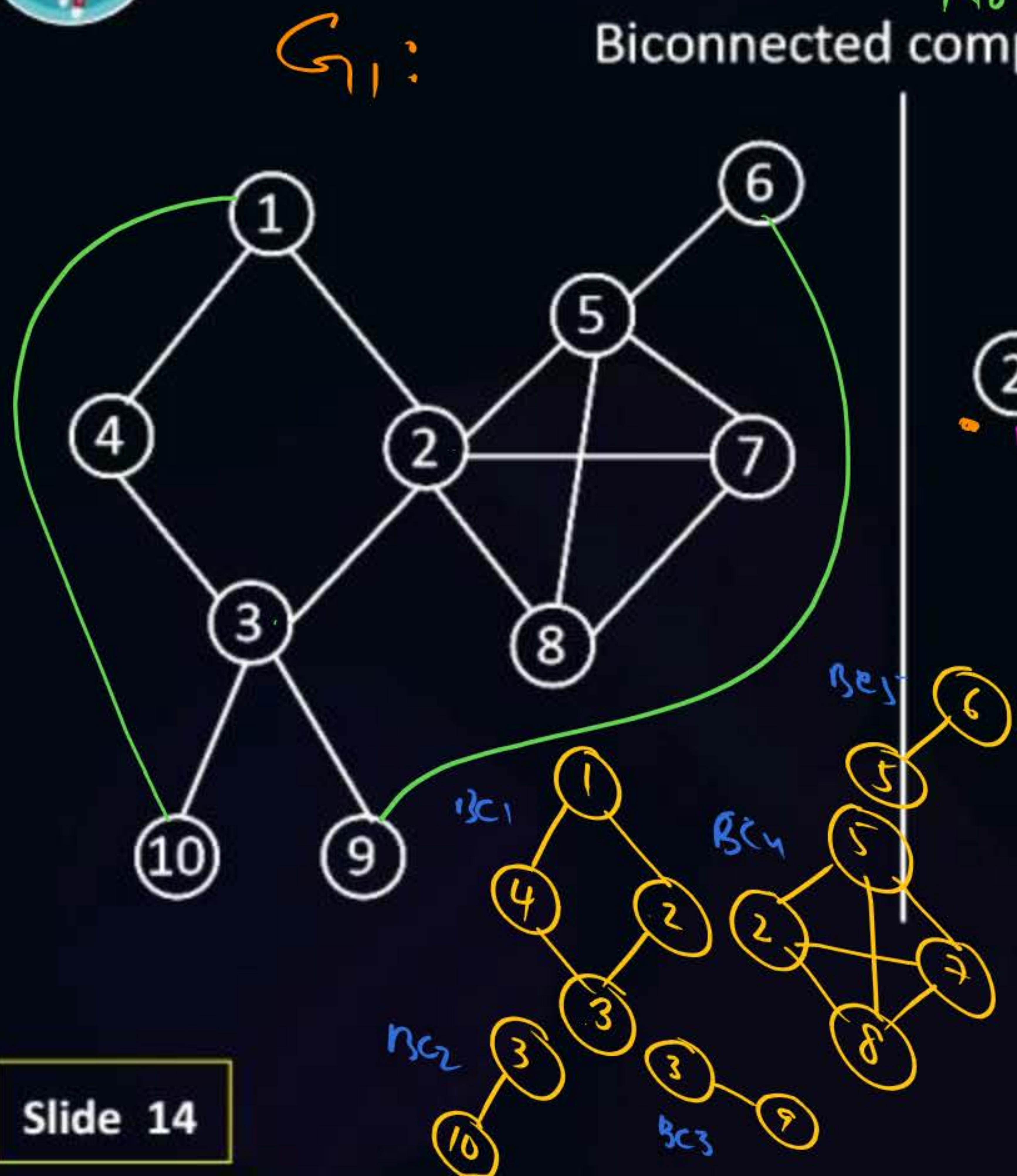
2) A graph is said to be Bi-Connected if it does not contain any Articulation point

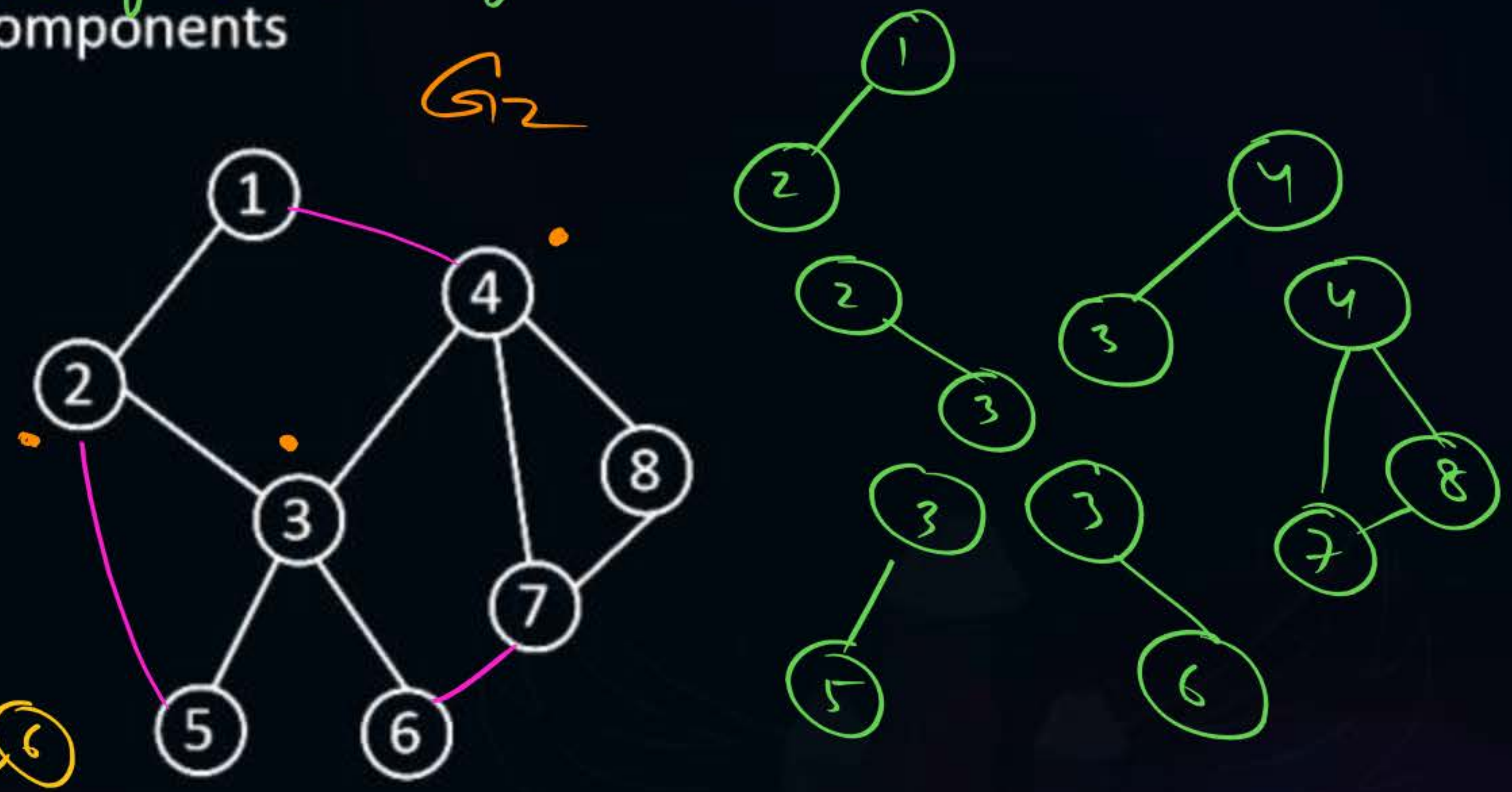
3) If the graph is not Bi-Connected then its maximal Subgraph which is BiConnected is Biconnected Component

Topic : Algorithms

No. of edges ≤ No. of A.P's

Biconnected components

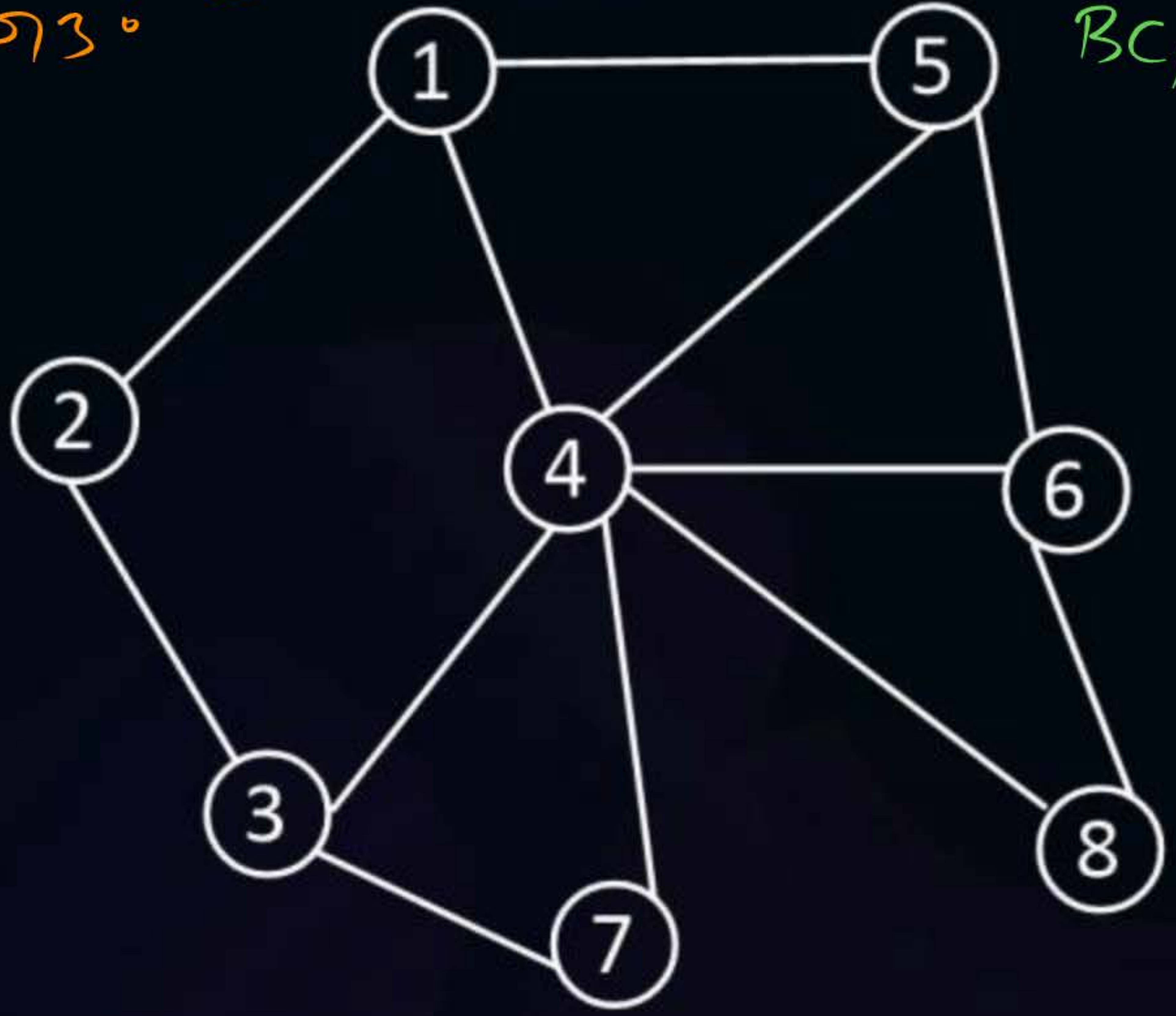




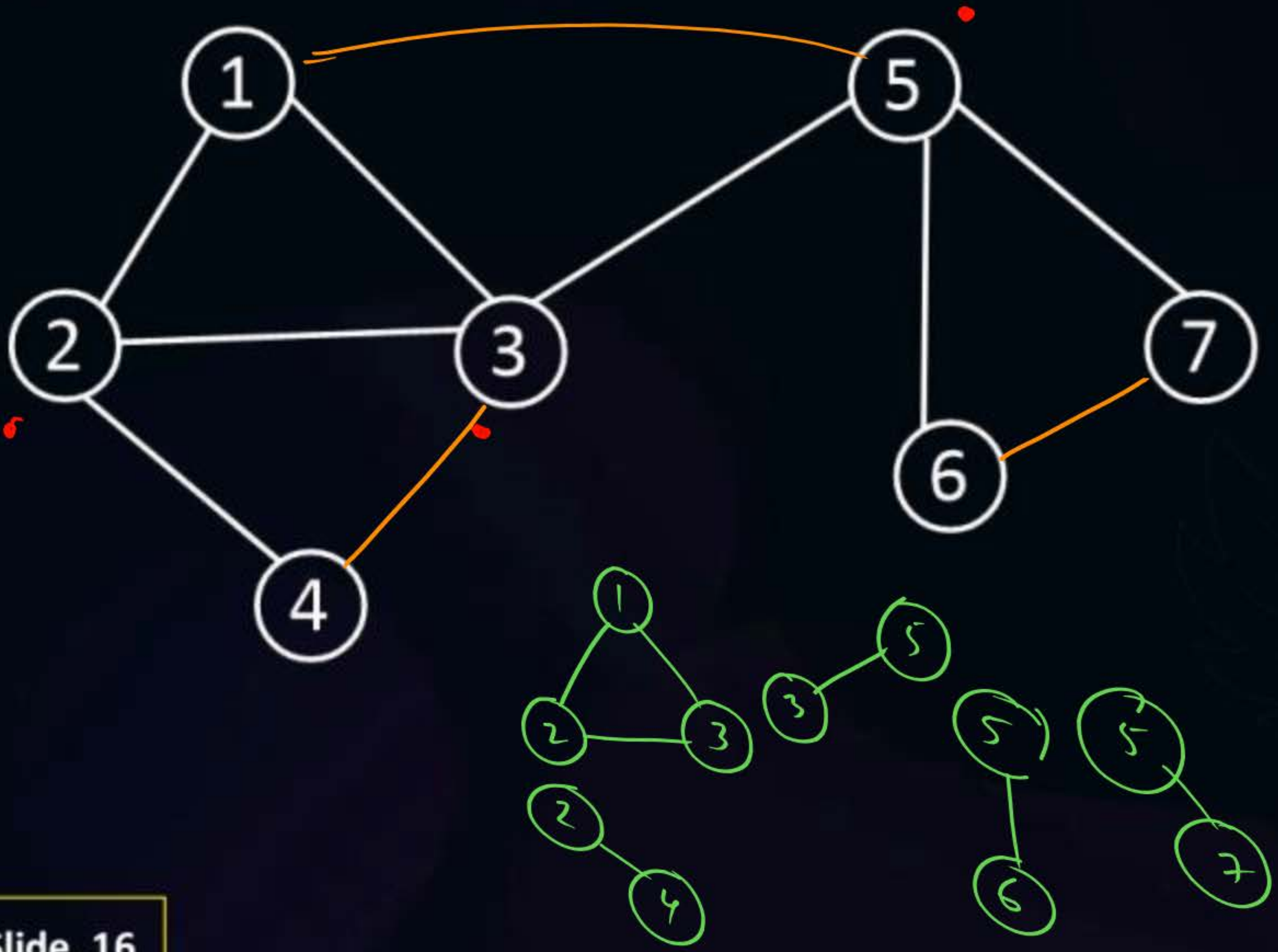
Slide 14

Topic : Algorithms

G_3 : '0'



BC,


1
2
3
4
5
6
7
1
2
3
3
5
2
4
5
6
5
7

Topic :

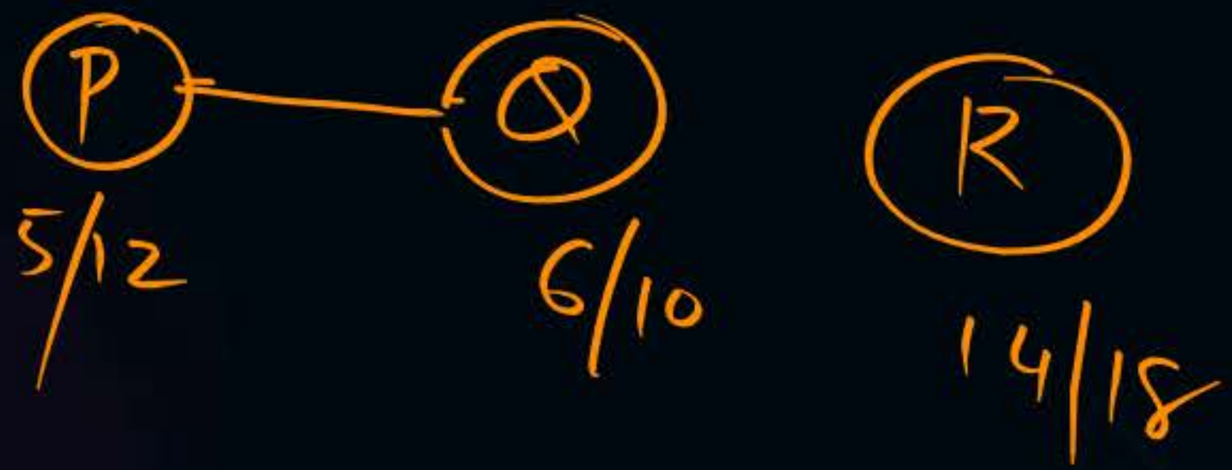
#Q. Consider the depth-first-search of an undirected graph with 3 vertices P,Q & R. let discovery time d(V) represent the time instant when the vertex 'V' is visited first, and finish time f(V) represent finishing time; given

d(p) =5 f(p) =12

d(Q)=6 f(Q) = 10

d(R) =14 f(R)= 18

Which is true?



A. There is only one connected component.

B. There are two connected components , P &R are connected

C. There are two connected components, Q &R are

D. There are two connected components , P & Q are connected

Topic :

#Q. Consider an undirected graph (unweighted). If BFS of G is done from a node 'r' let d (r,u) and d(r,v) be the lengths of the shortest paths from r to u & v. if 'u' is visited before 'v', during the traversal, then which is true?

A. $d(r,u) < d(r,v)$

B. $d(r,u) > d(r,v)$

C. $d(r,u) \leq d(r,v)$

D. none

Topic :

#Q. In a DF-traversal of a graph 'G' with n-vertices, 'k' edges are marked as tree edges, the number of connected components of 'G' is

A. K

B. K+1

C. (n-k-1)

D. n-k

$n \rightarrow K = (n-1)$: Connected (1-Component)

$n - (n-1) = 1$

Topic :

#Q. DFS is performed on a directed a cyclic graph. D(u) is discovery time and f(u) is finishing time. Which is true for all edges (u,v) in the graph?

$u \rightarrow v$

A. d[u] < d[v]

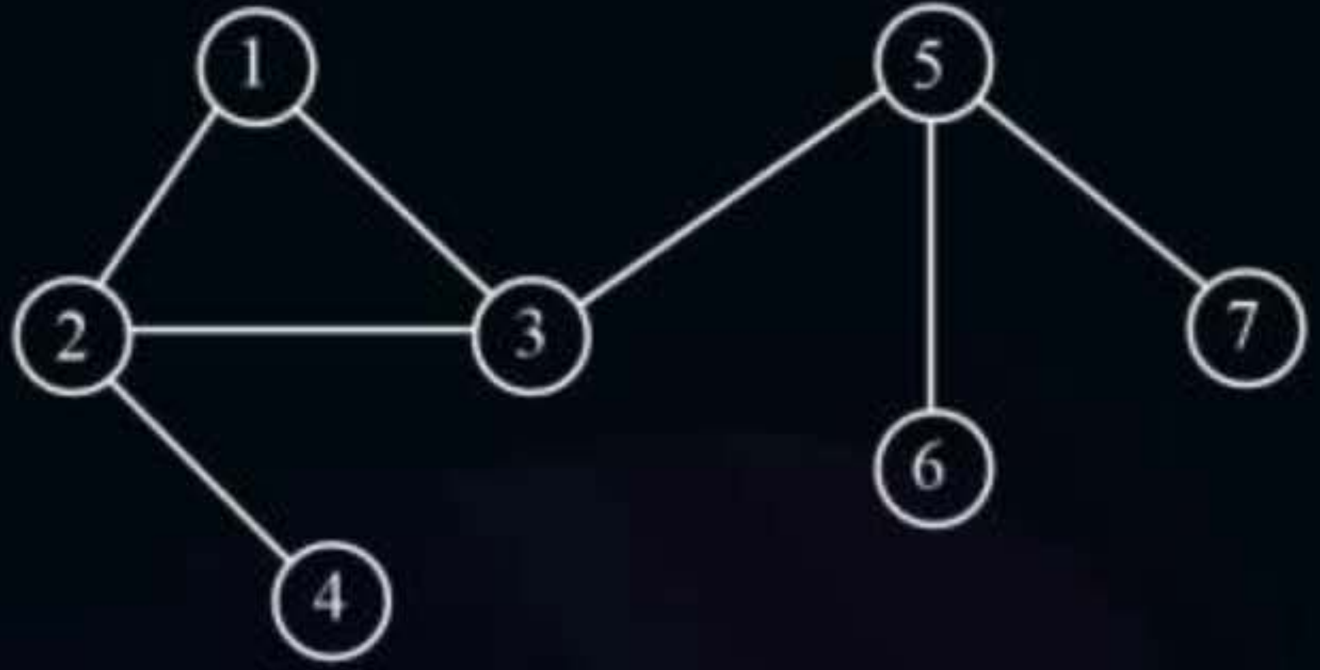
B. d[u] < f[v]

C. F[u] < f[v]

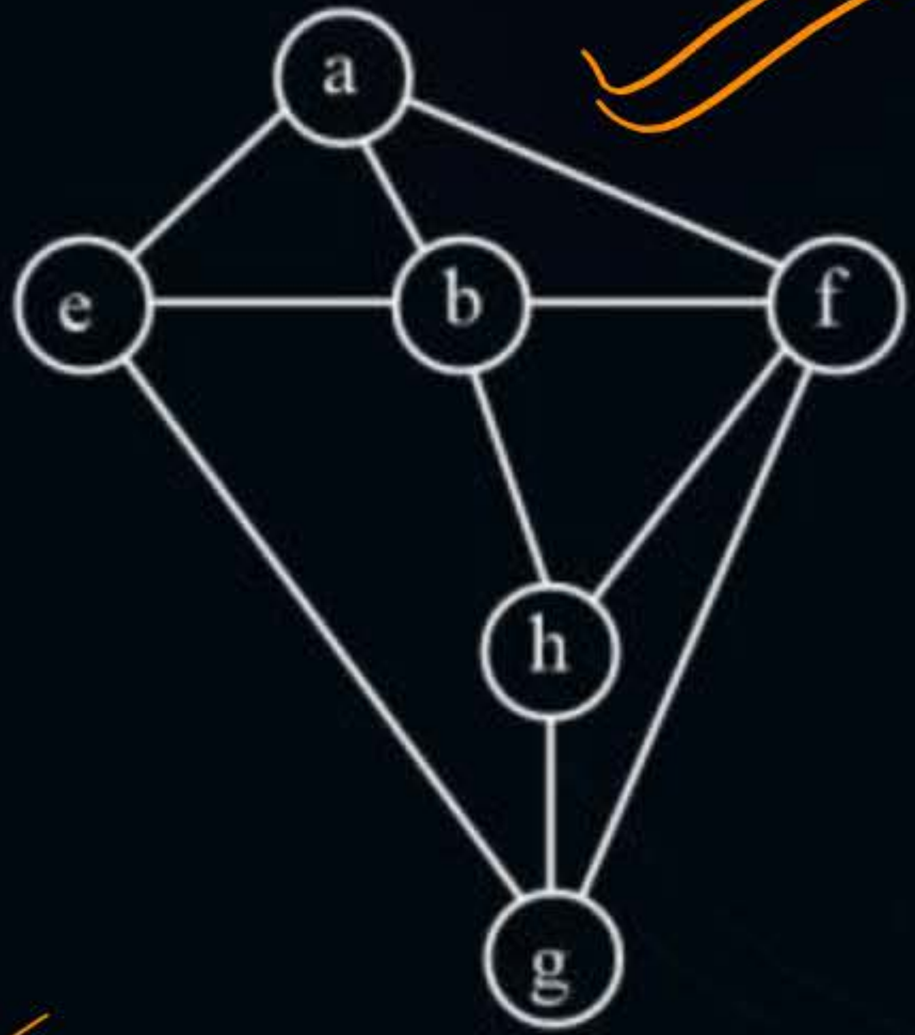
D. f[u] > f[v]

Topic :

1.



No. of Articulation Pts.



DFS : valid | Invalid

A. a b e g h f

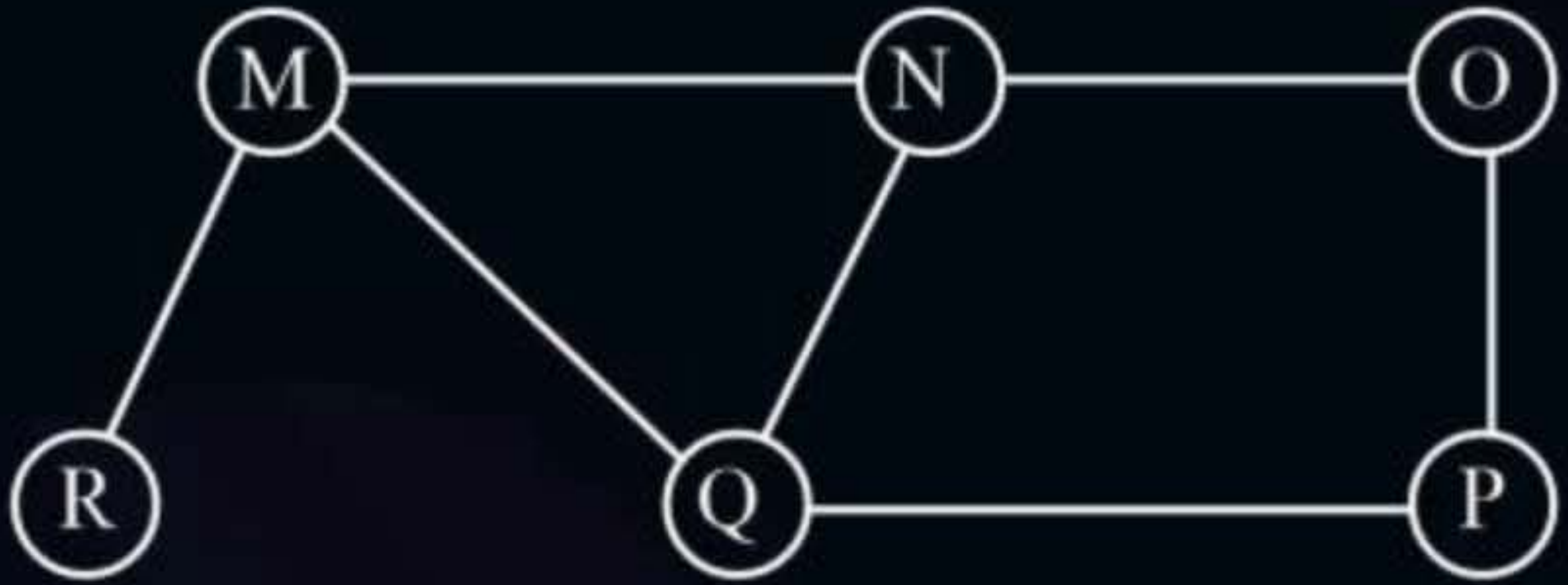
B. a b f e h g

C. a b f h g e

D. a f g h b e

Topic :

#Q.



BFS : Valid

×A. M N O P Q R

×B. N Q M P O R

✓C. Q M N P R O

D. Q M N P O R

THANK - YOU