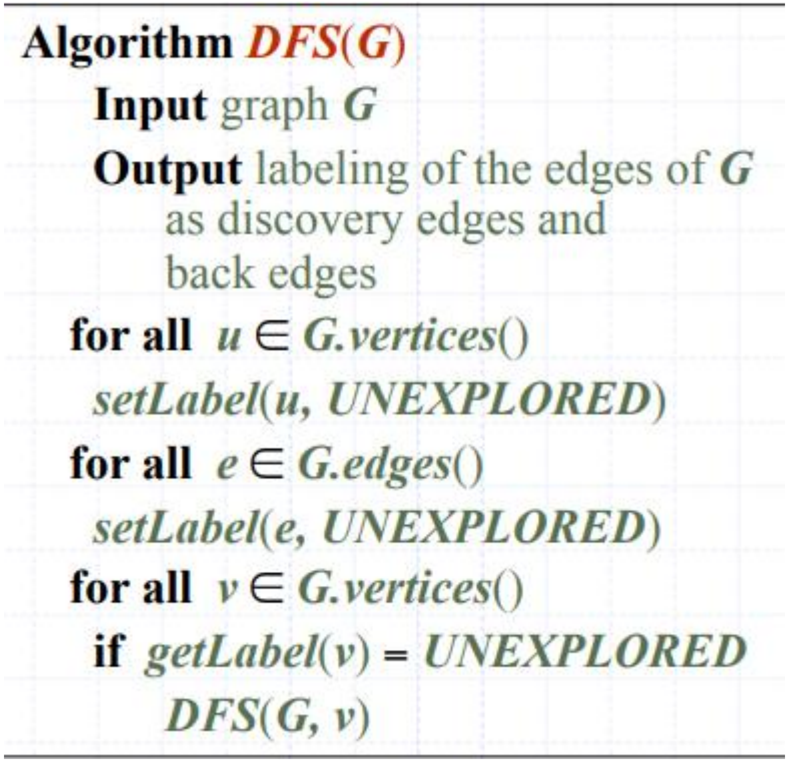
BITS Pilani
Hyderabad Campus

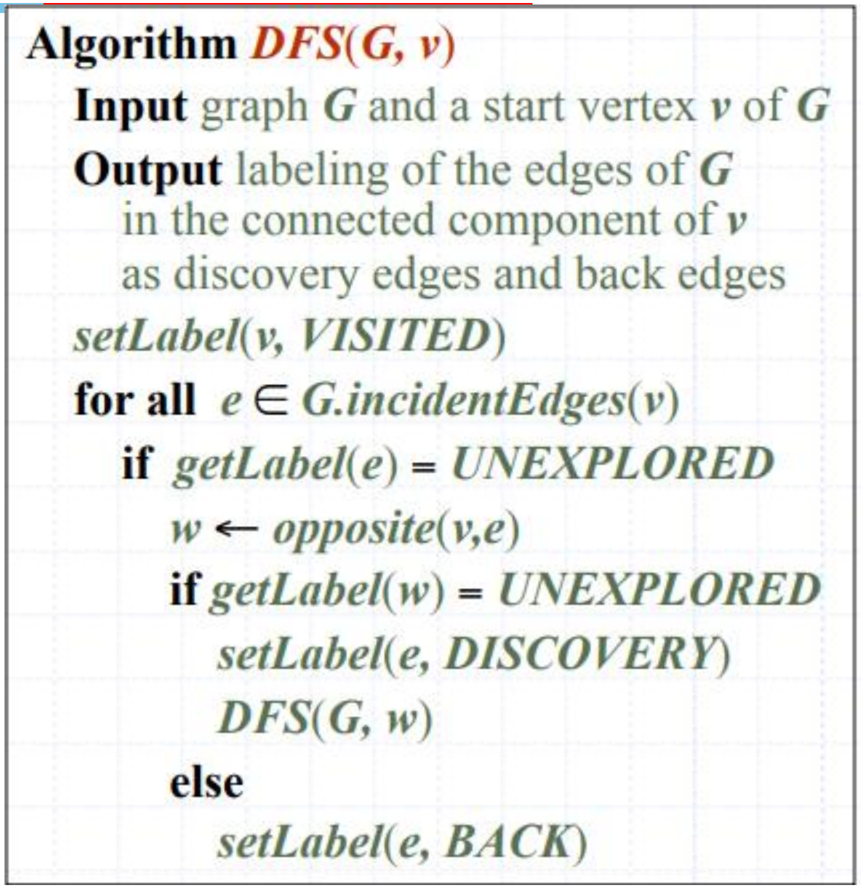
Data Structures and Algorithms Design

Febin.A.Vahab
2019-20

Depth-First Search

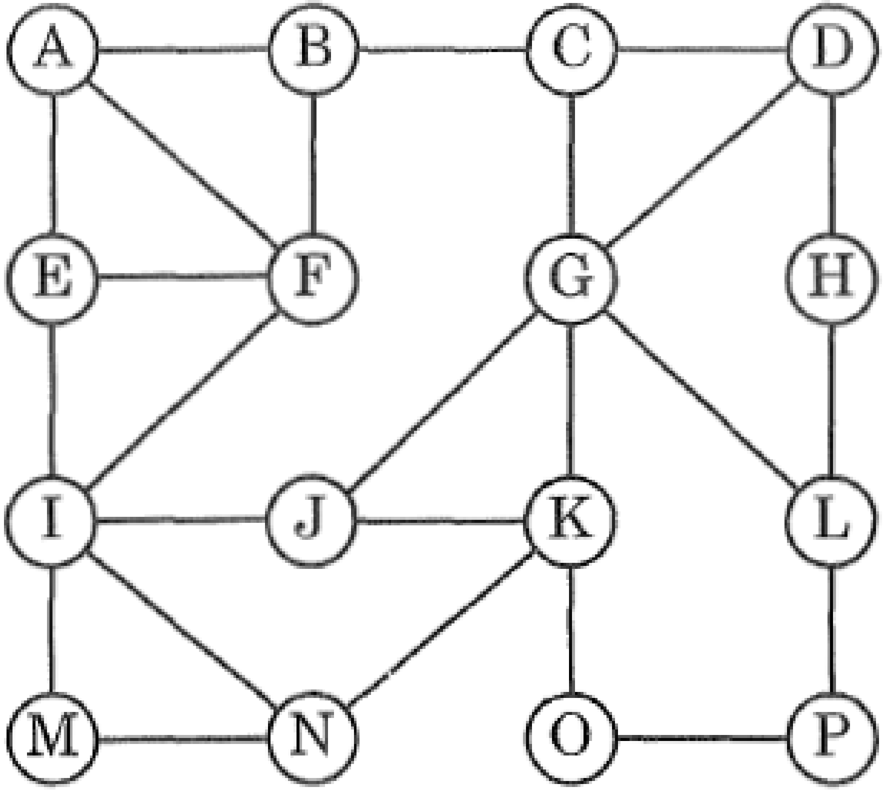


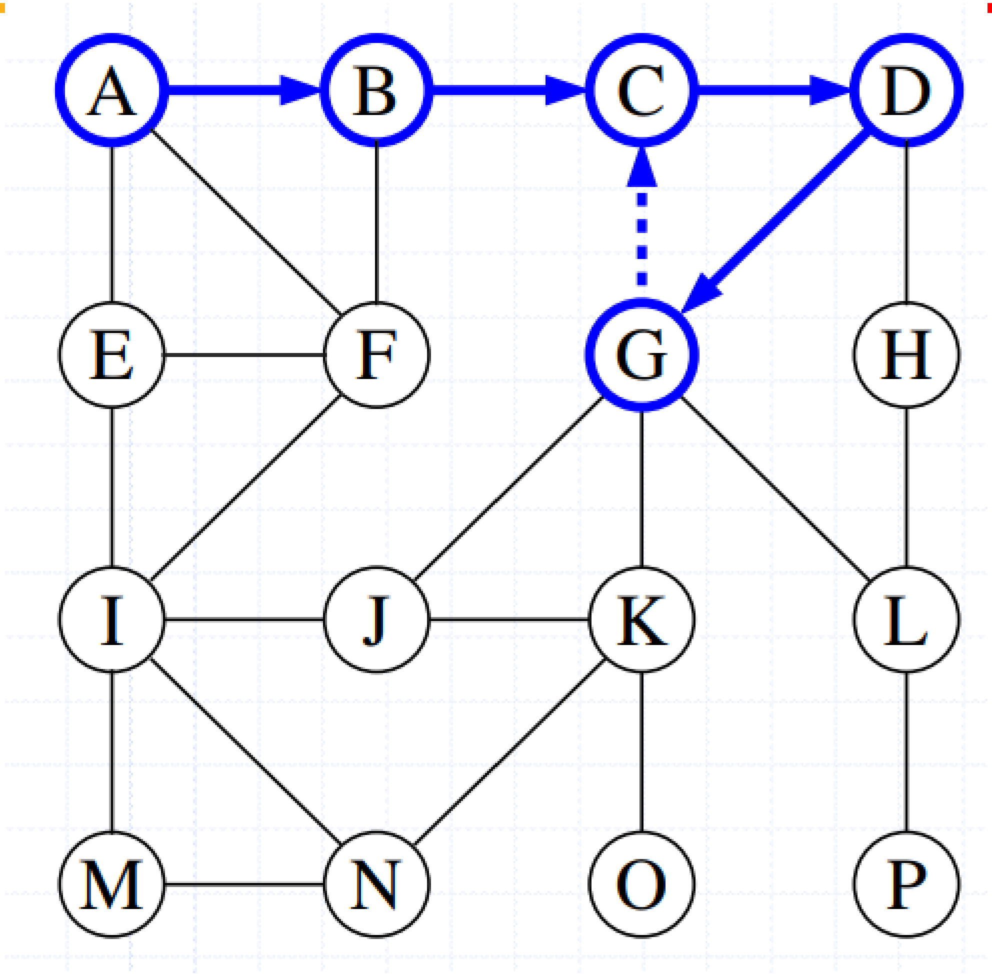
```
Algorithm DFS(G)
    Input graph G
    Output labeling of the edges of G
        as discovery edges and
        back edges
  for all $u \in G.vertices()$
    setLabel(u, UNEXPLORED)
  for all $e \in G.edges()$
    setLabel(e, UNEXPLORED)
  for all $v \in G.vertices()$
    if getLabel(v) = UNEXPLORED
        DFS(G, v)
```

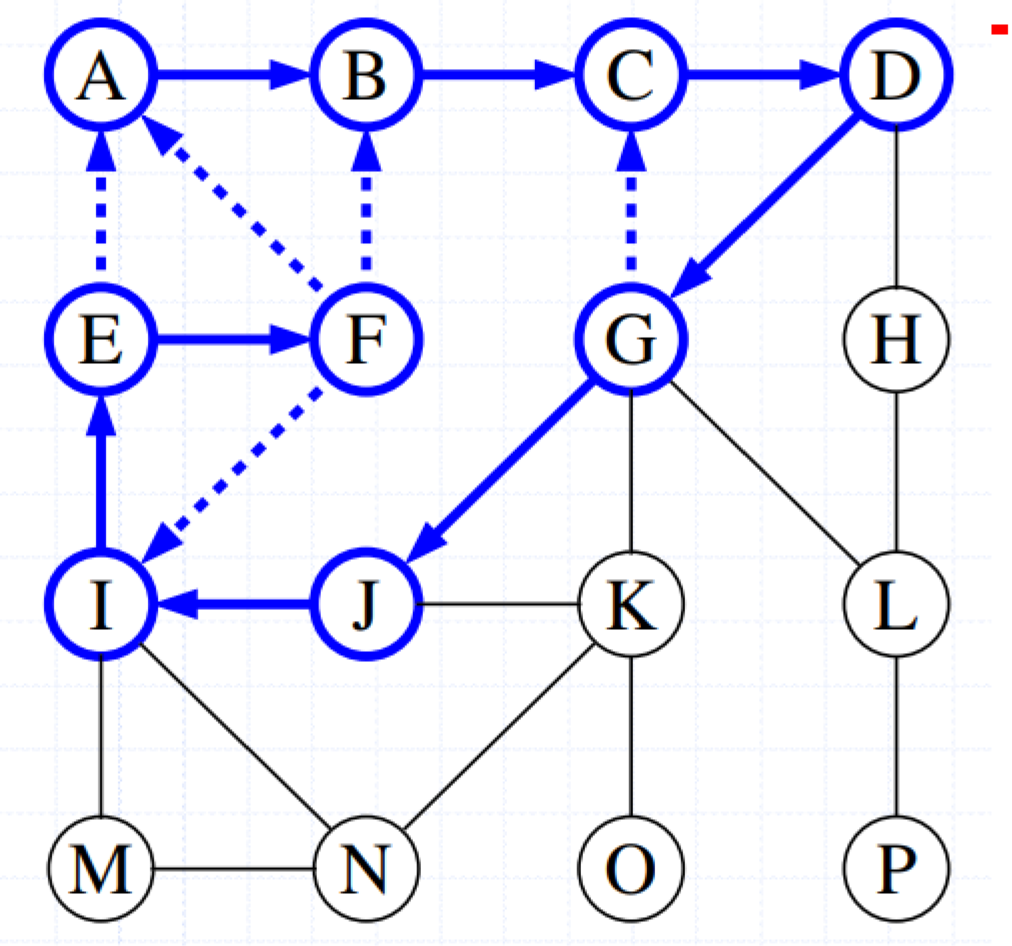


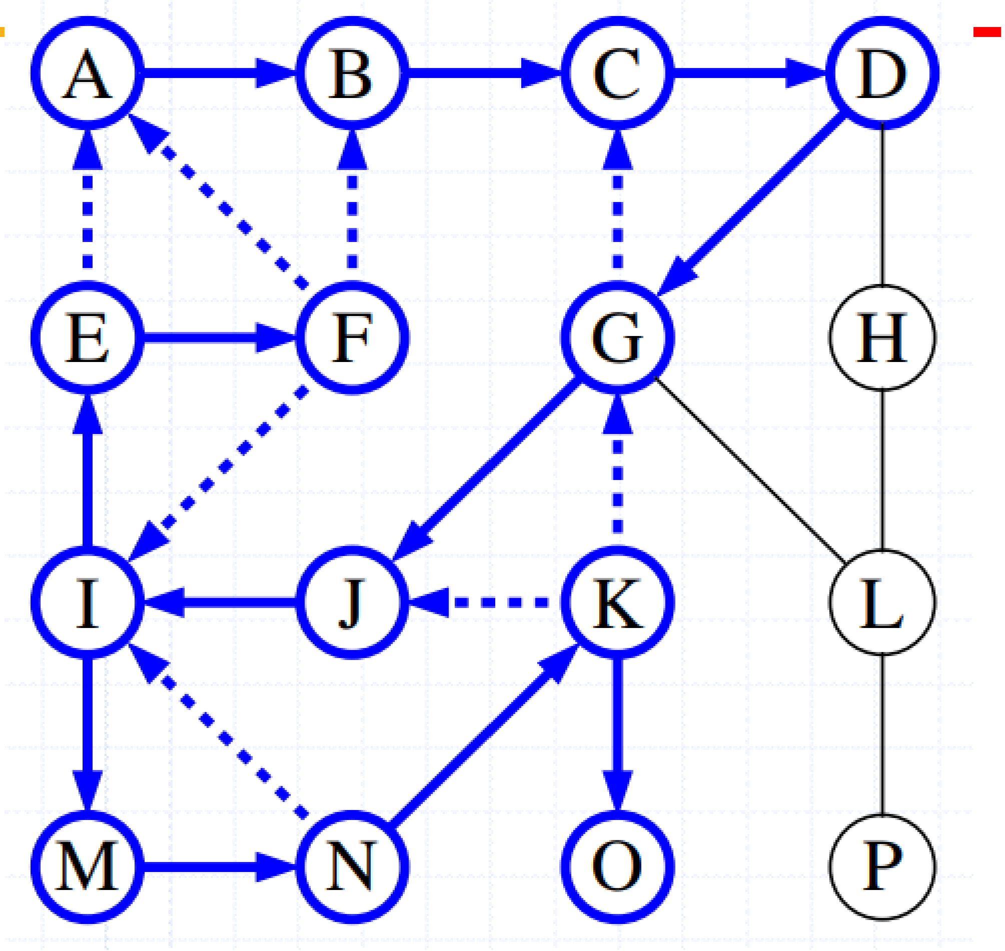
```
Algorithm DFS(G, v)
    Input graph G and a start vertex v of G
    Output labeling of the edges of G
        in the connected component of v
        as discovery edges and back edges
    setLabel(v, VISITED)
    for all $e \in G.incidentEdges(v)$
        if getLabel(e) = UNEXPLORED
            $w \leftarrow$ opposite(v,e)
            if getLabel(w) = UNEXPLORED
                setLabel(e, DISCOVERY)
                DFS(G, w)
            else
                setLabel(e, BACK)
```

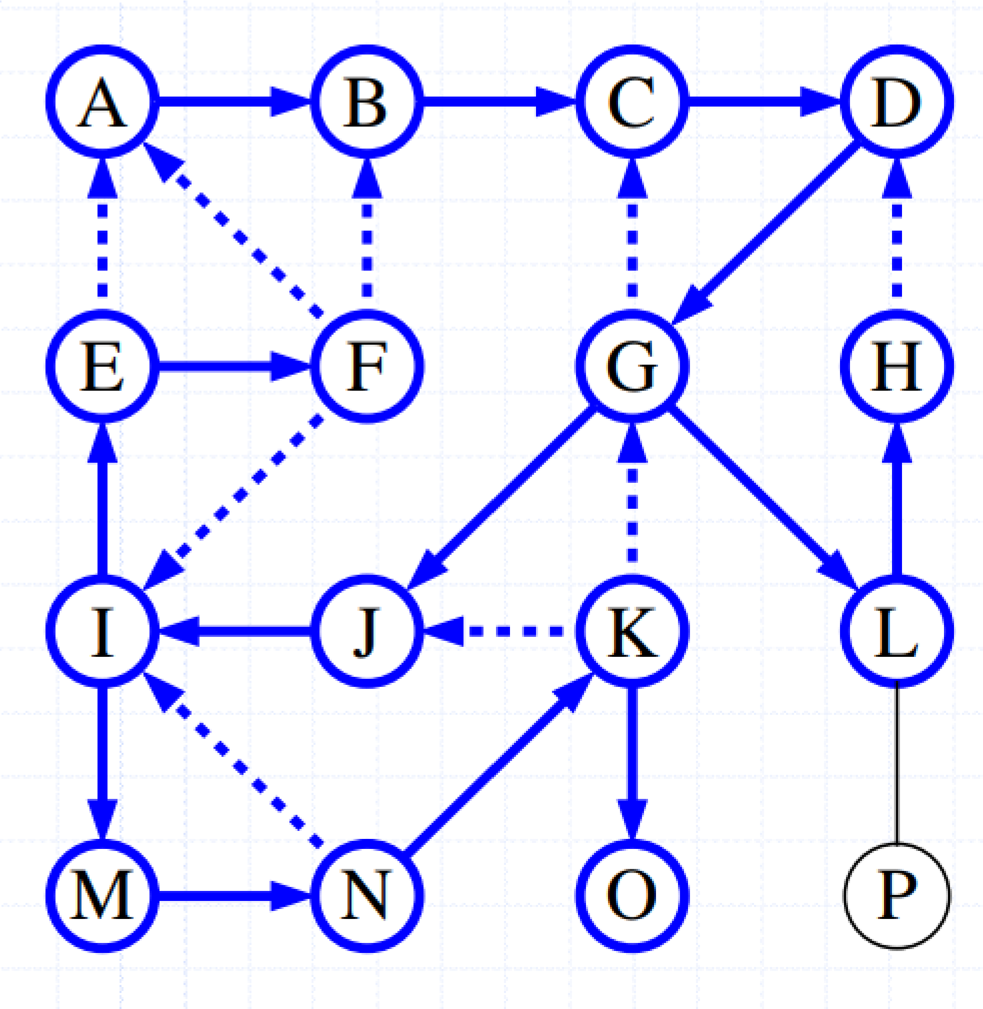
Example 1-Solved

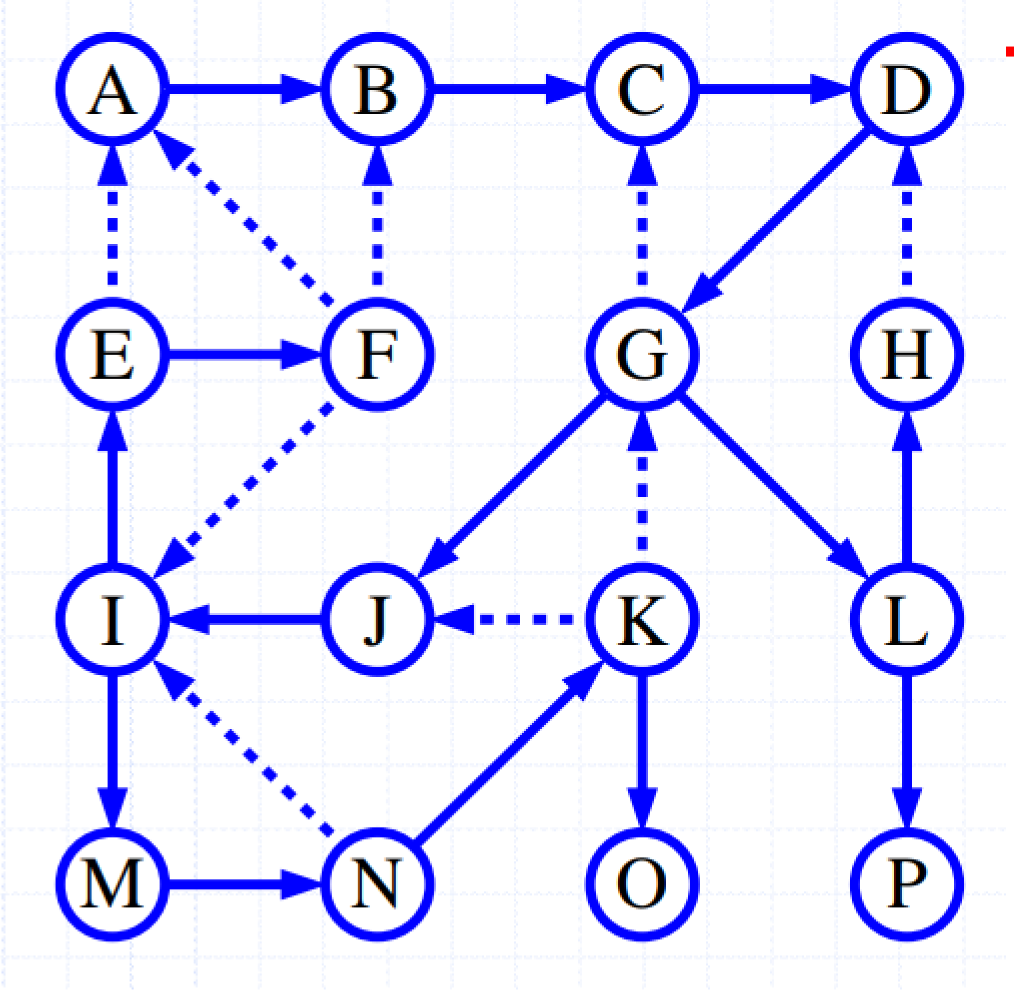



A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P

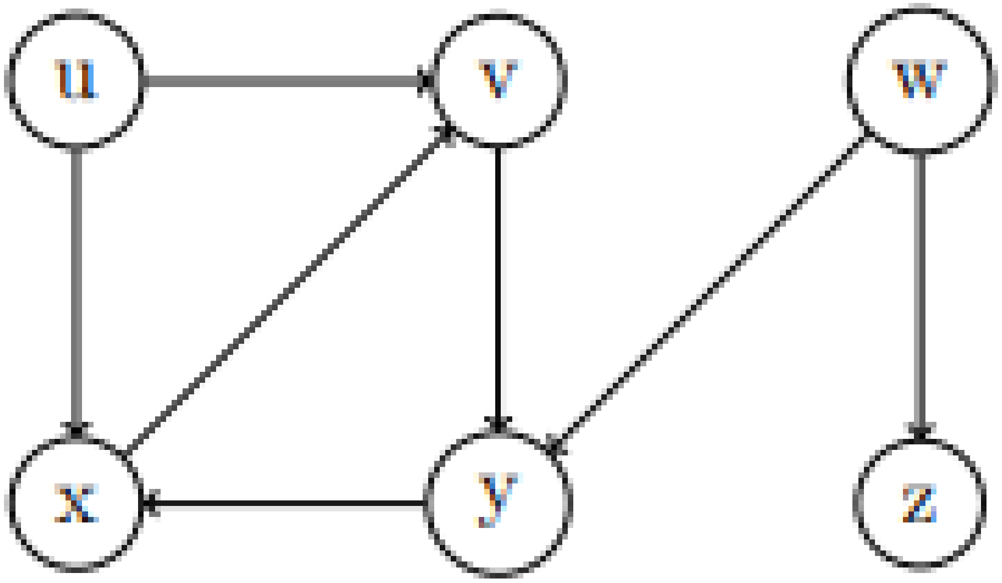

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P


A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P

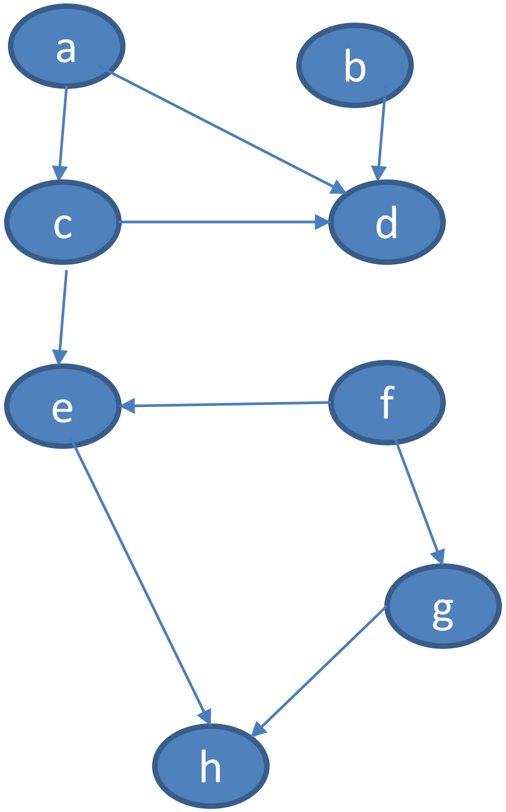

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P


A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P

Example 2-Directed graph-Find the DFS tree-Discuss in Canvas



Example 3-Directed graph-Find the connected Components-Discuss in Canvas



BITS Pilani
Hyderabad Campus

THANK YOU!