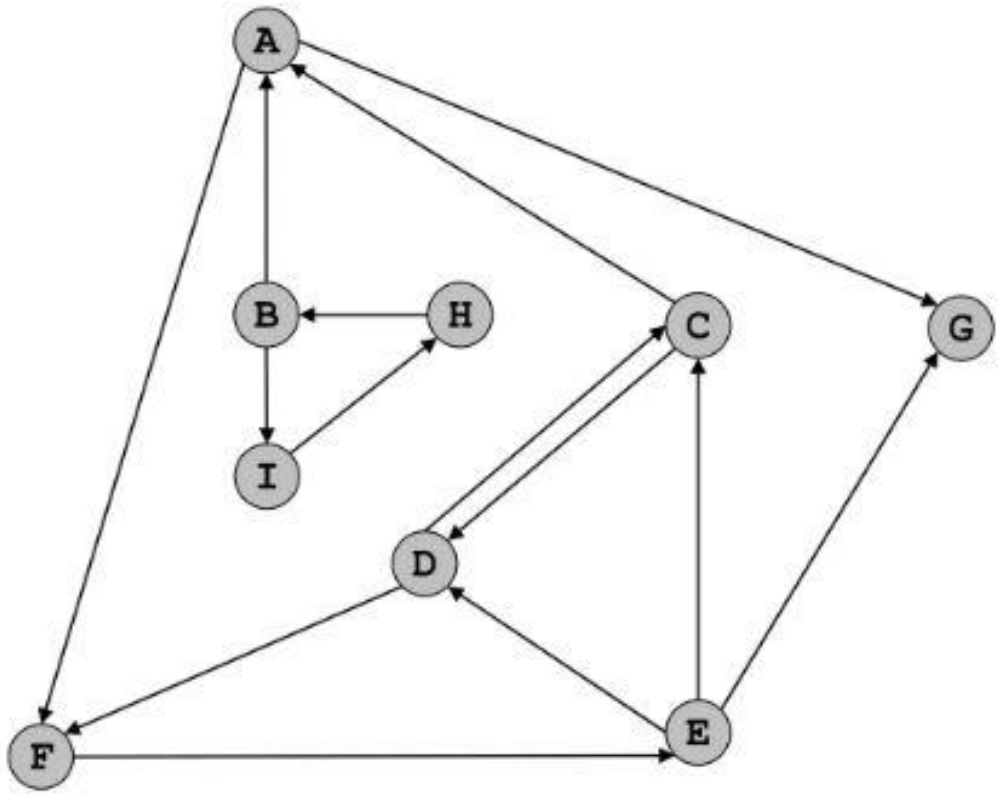
DFS

Depth First Search (DFS)

- The DFS algorithm is a recursive algorithm that uses the idea of backtracking.
- It involves exhaustive searches of all the nodes by going ahead, if possible, else by backtracking.
- This recursive nature of DFS can be implemented using stacks.

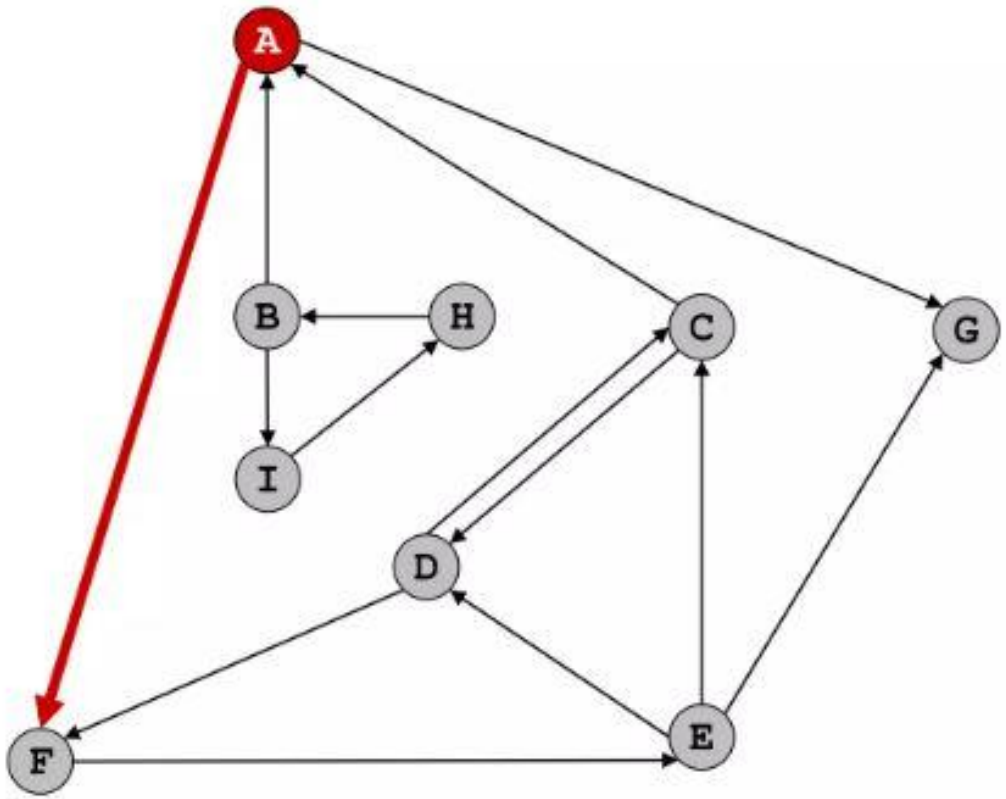
The basic idea is as follows:

- Pick a starting node and push all its adjacent nodes into a stack.
- Pop a node from stack to select the next node to visit and push all its adjacent nodes into a stack.
- Repeat this process until the stack is empty. However, ensure that the nodes that are visited are marked.
- This will prevent you from visiting the same node more than once.
- If you do not mark the nodes that are visited and you visit the same node more than once, you may end up in an infinite loop.

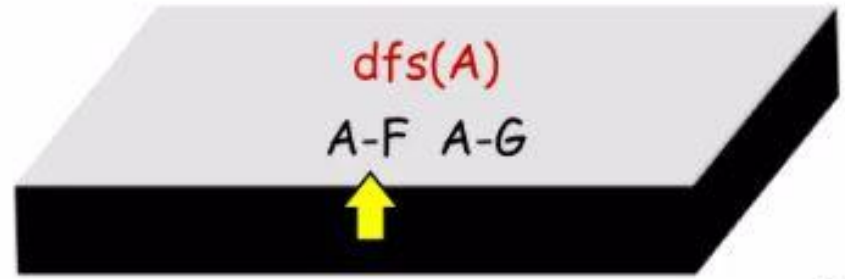


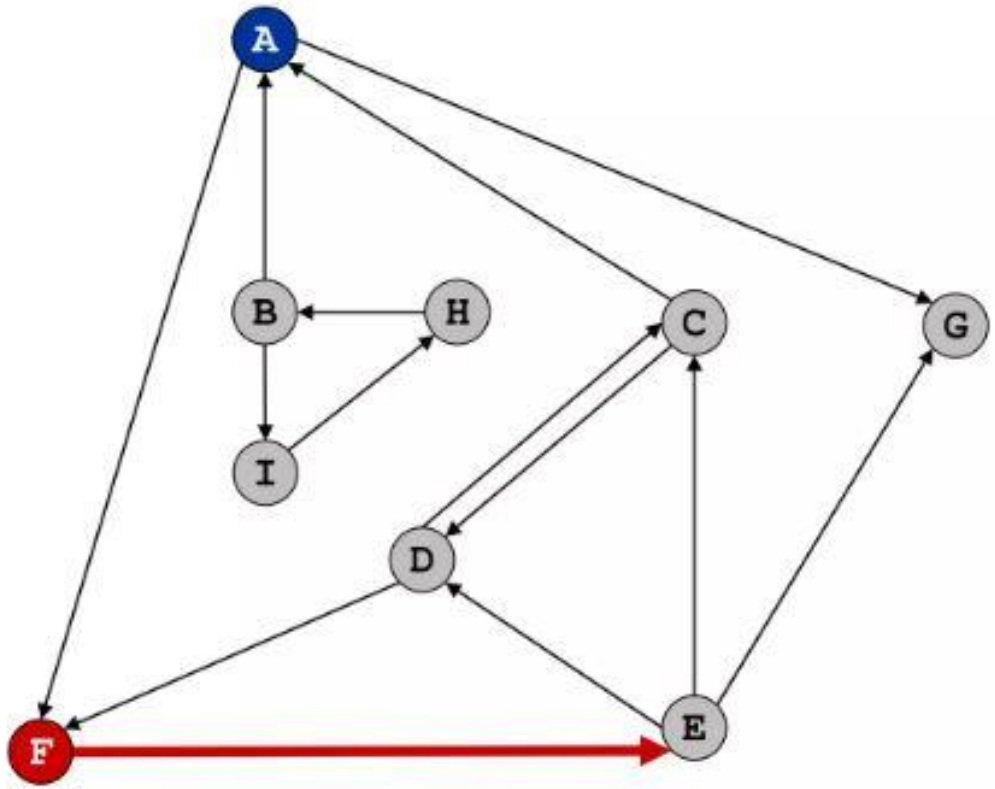
Adjacency Lists

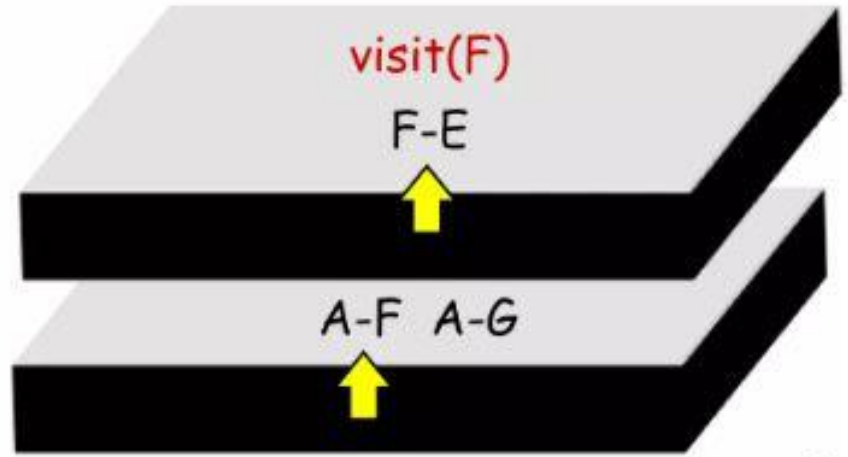
```
A:   F G
B:   A H
C:   A D
D:   C F
E:   C D G
F:   E
G:
H:   B
I:   H
```



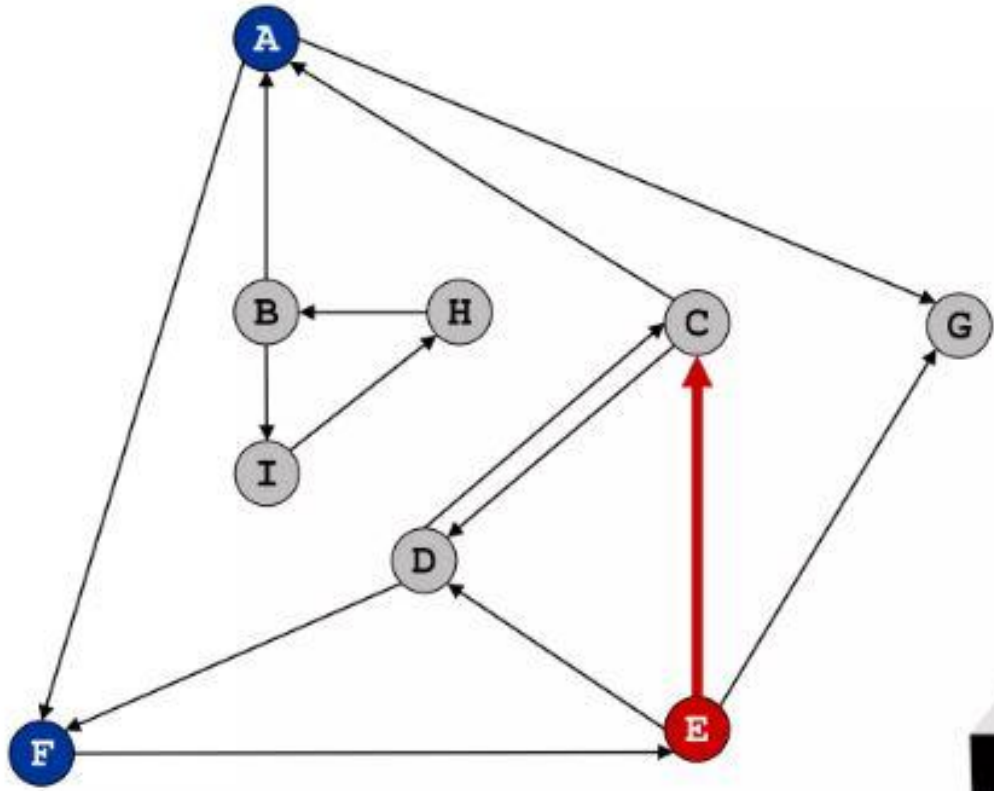
Function call stack:

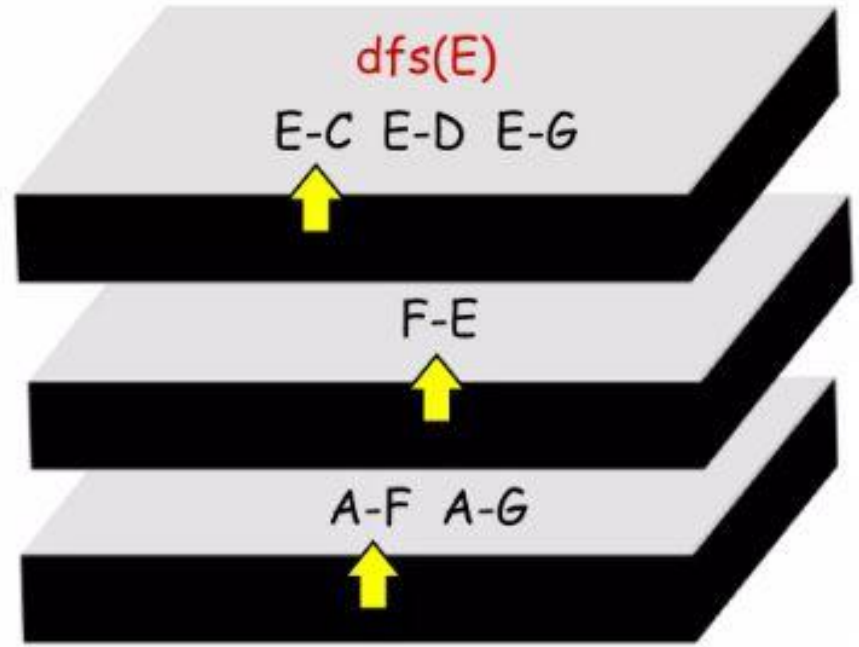




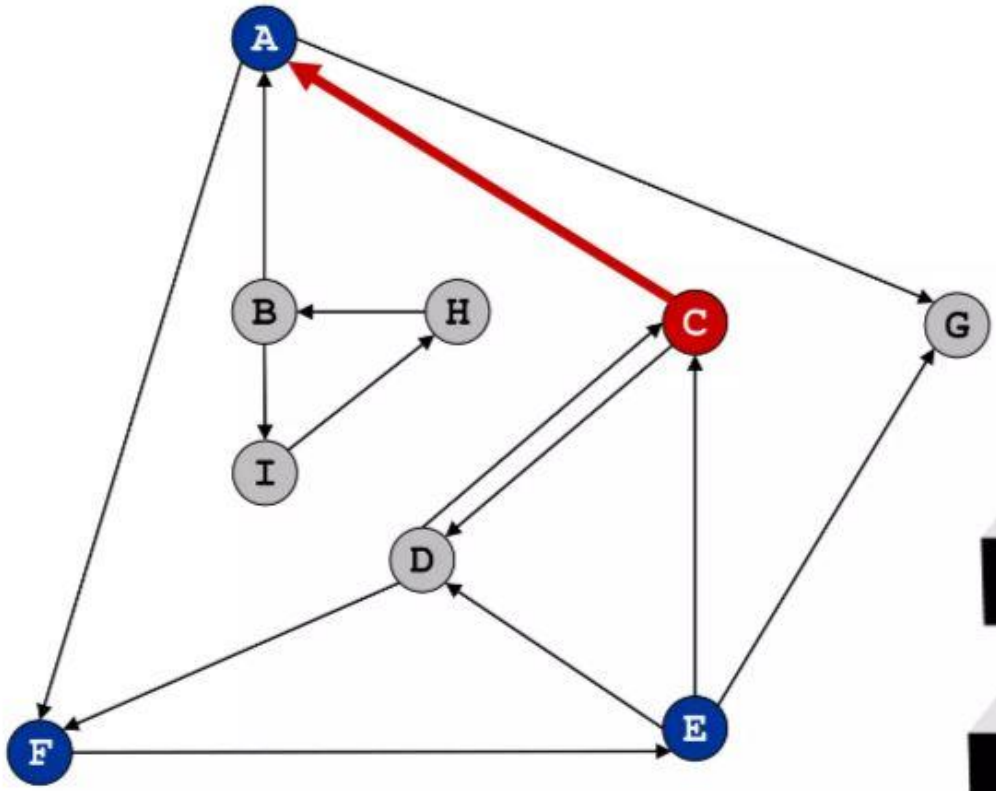


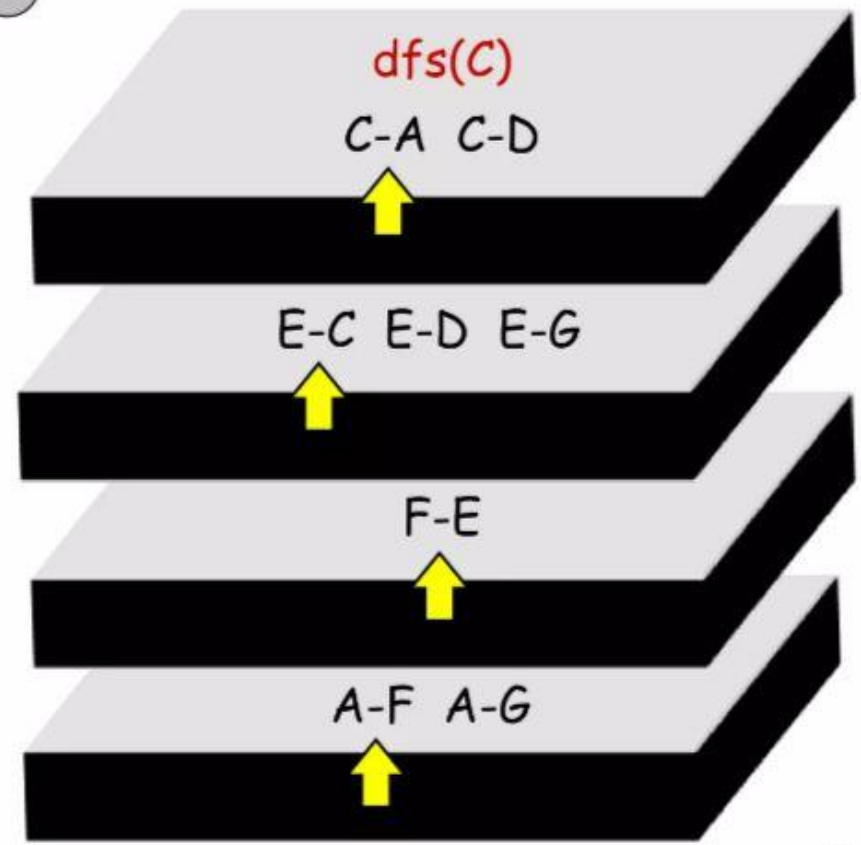
Function call stack:



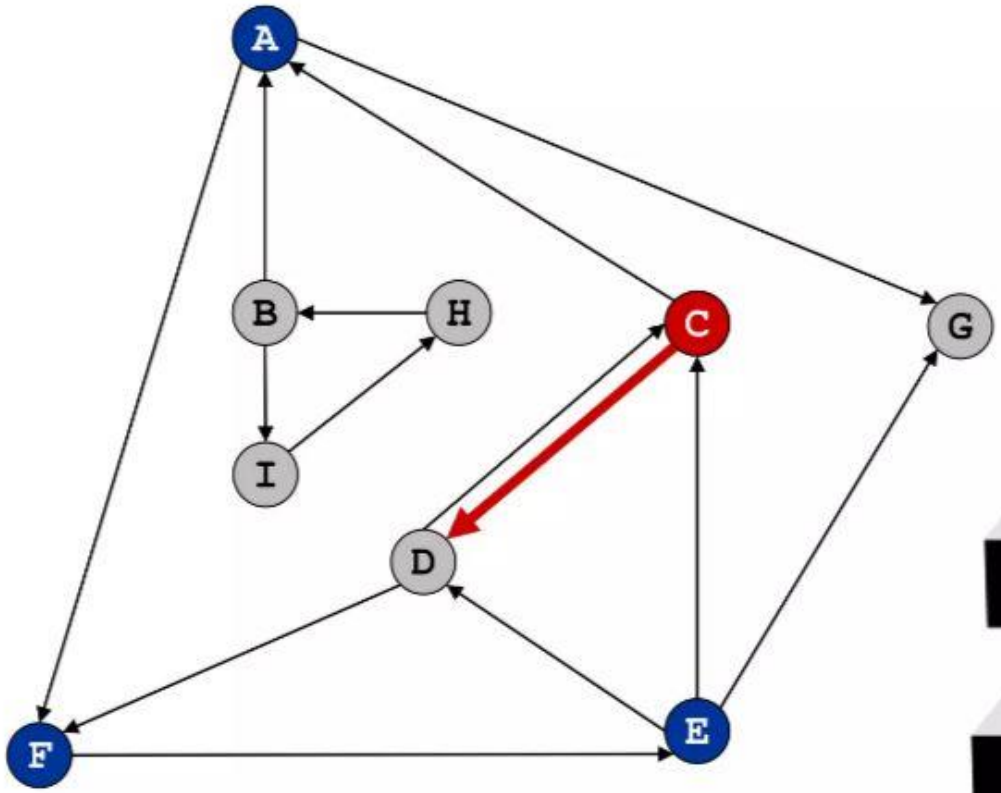


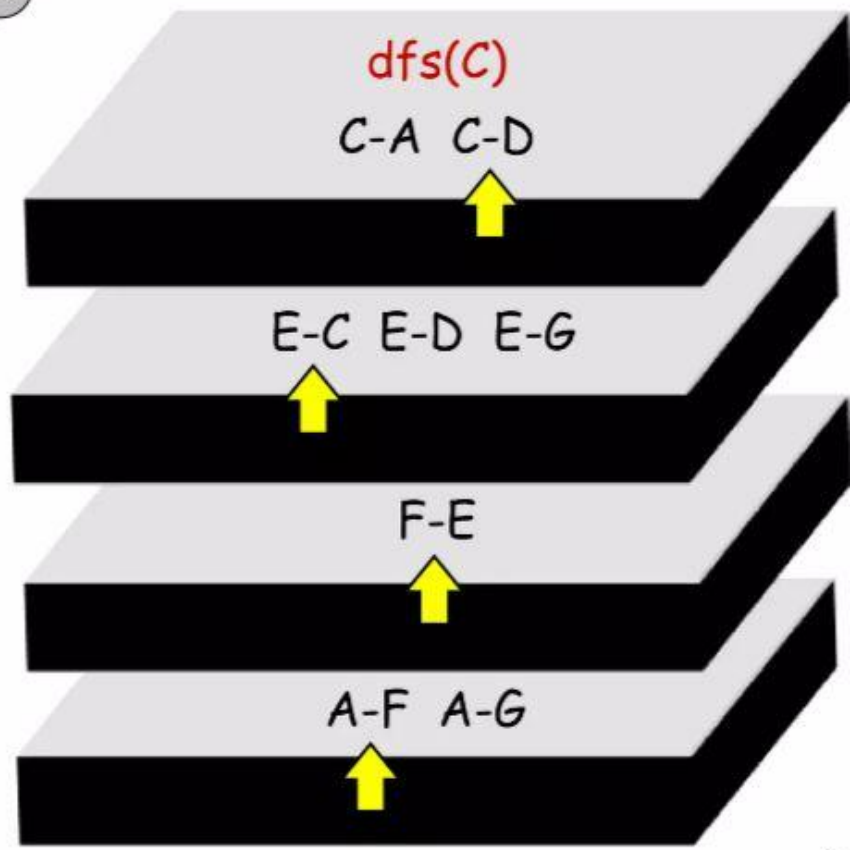
Function call stack:



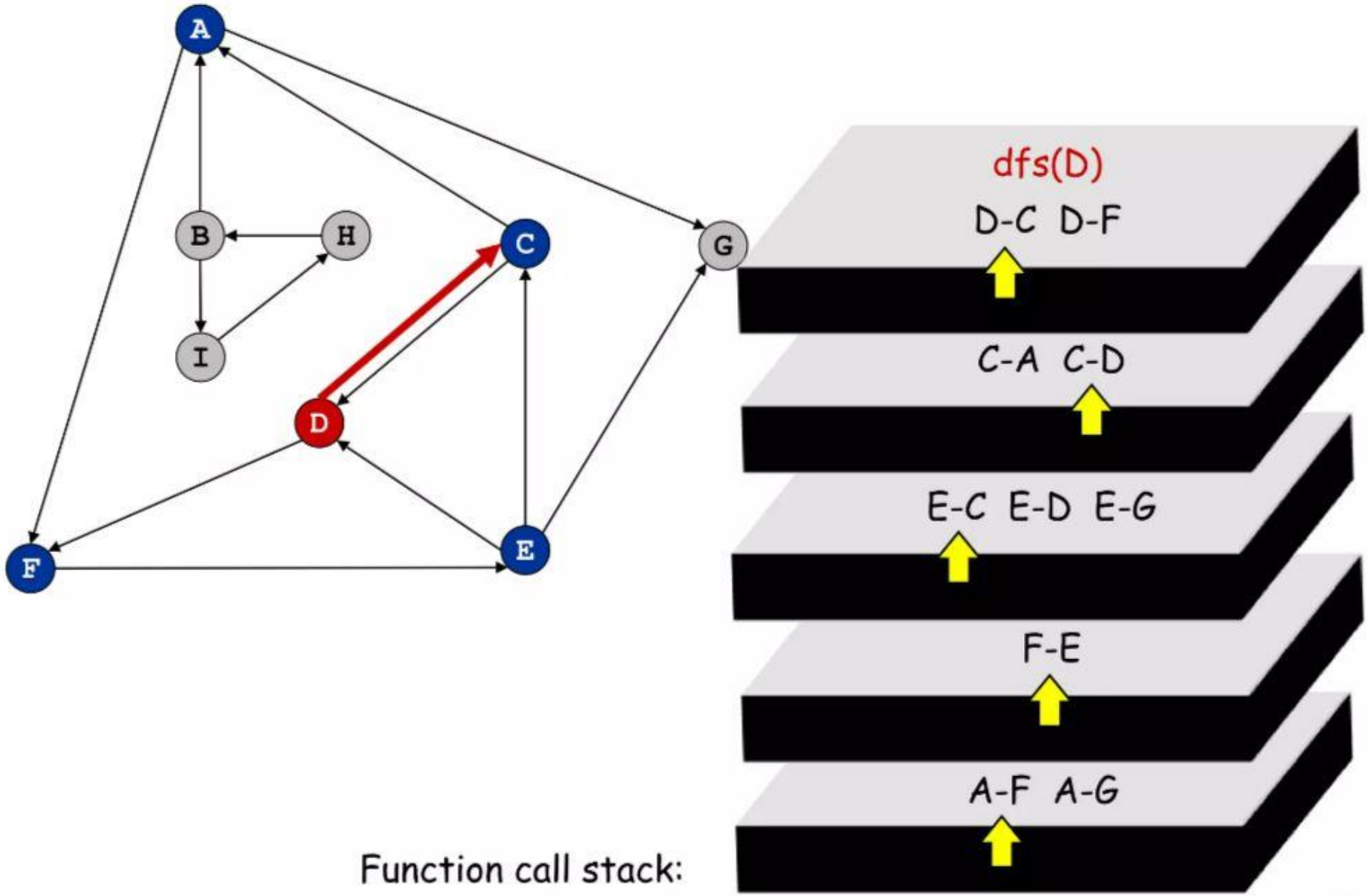


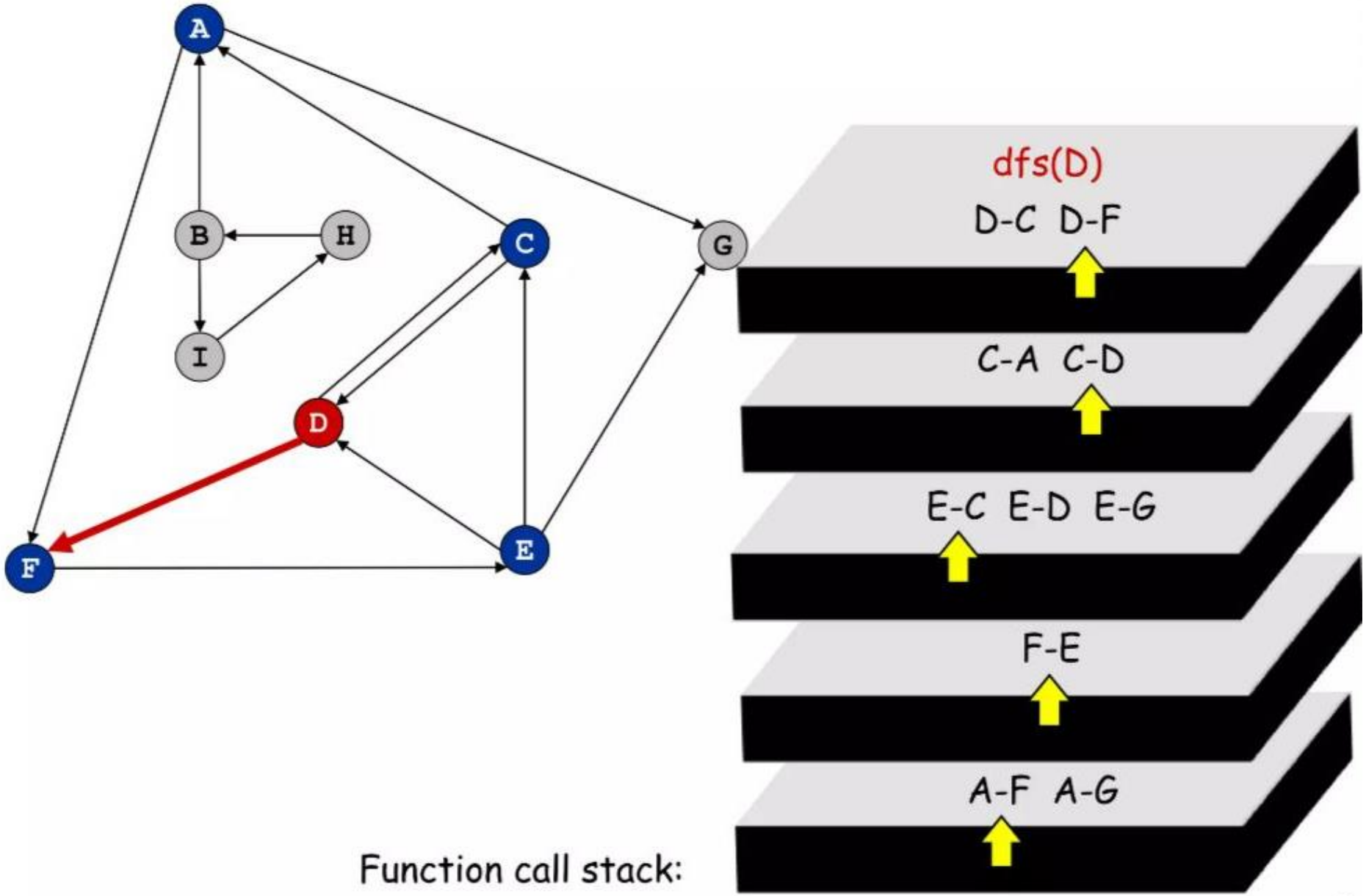
Function call stack:

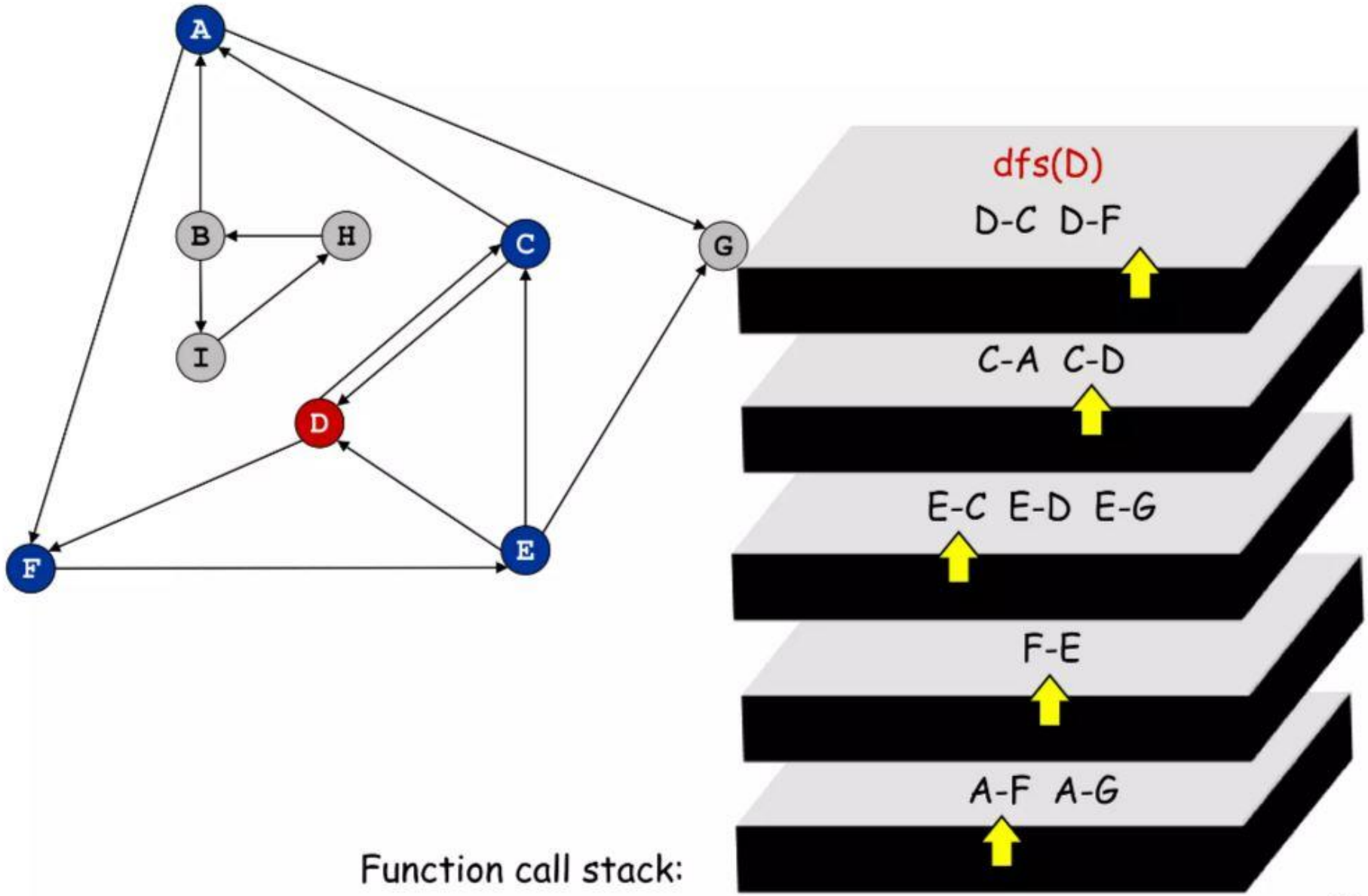


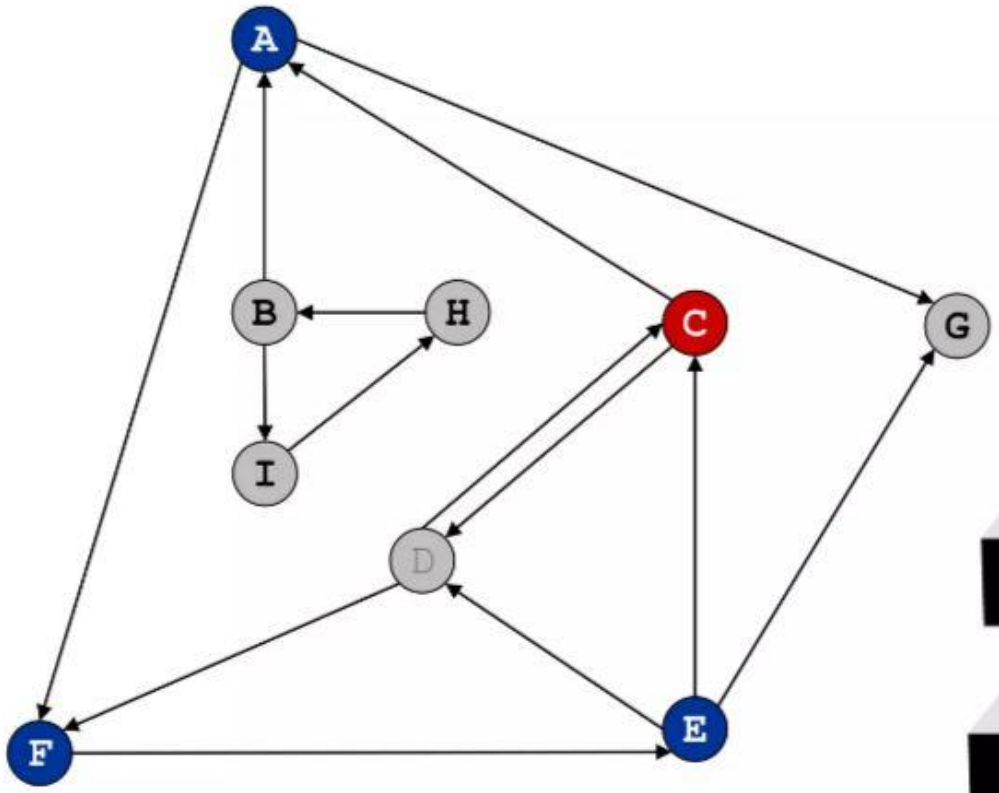


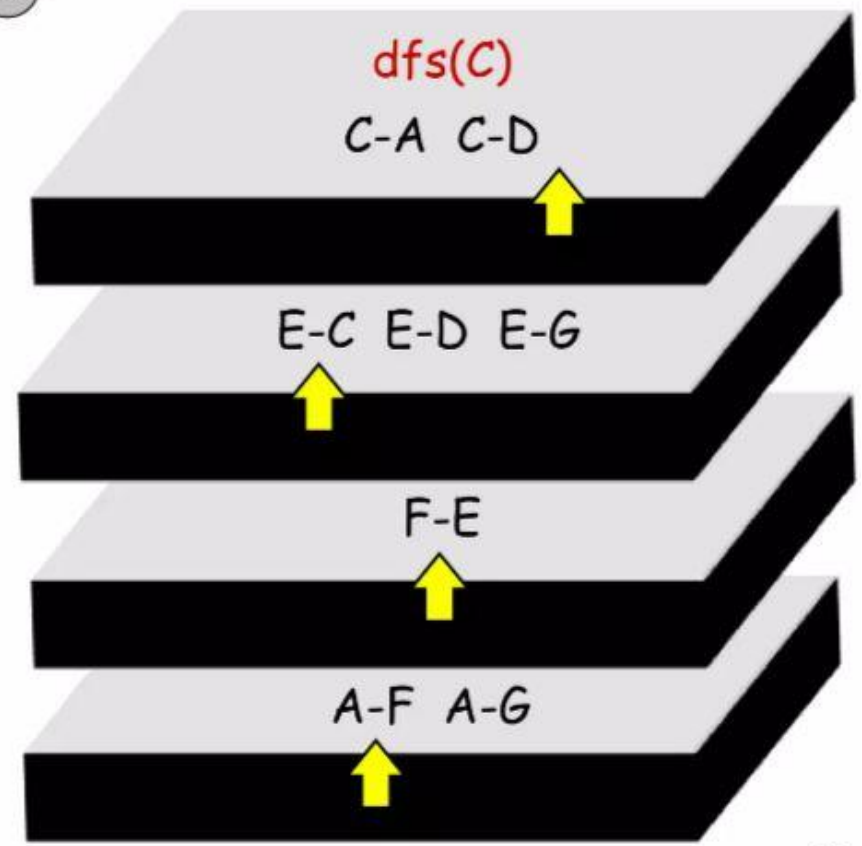
Function call stack:


A
B
H
C
G
I
D
F
E
dfs(D)
D-C D-F
C-A C-D
E-C E-D E-G
F-E
A-F A-G
Function call stack:

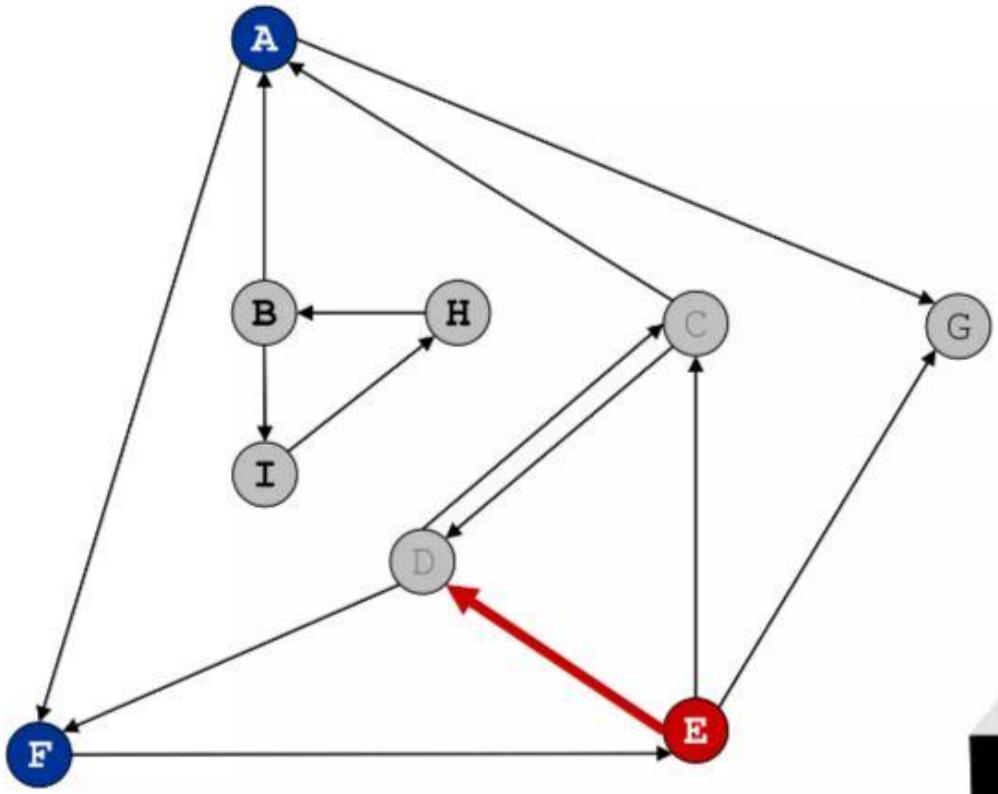

A
B
H
C
G
I
D
F
E
dfs(D)
D-C D-F
C-A C-D
E-C E-D E-G
F-E
A-F A-G
Function call stack:

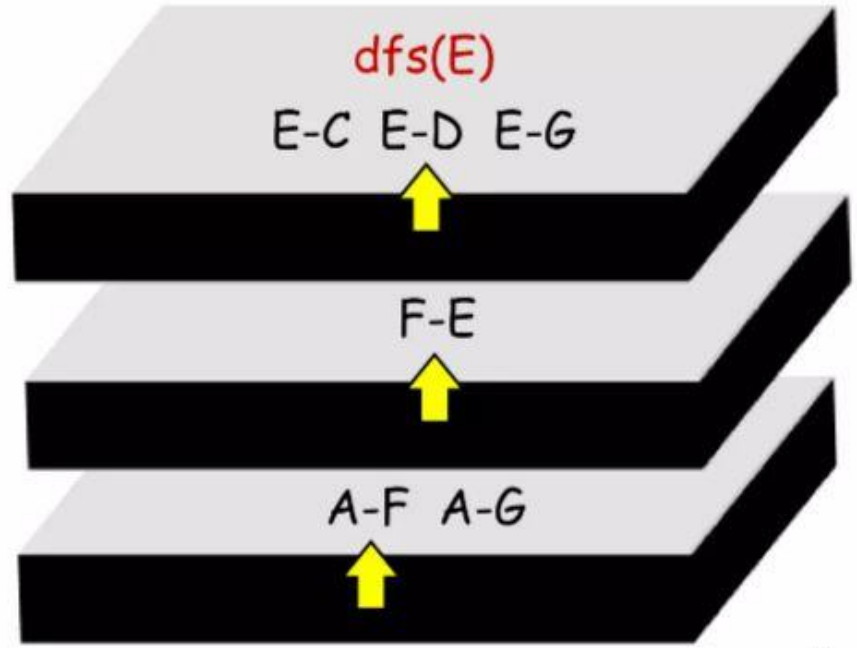

A
B
H
C
G
I
D
F
E
dfs(D)
D-C D-F
C-A C-D
E-C E-D E-G
F-E
A-F A-G
Function call stack:



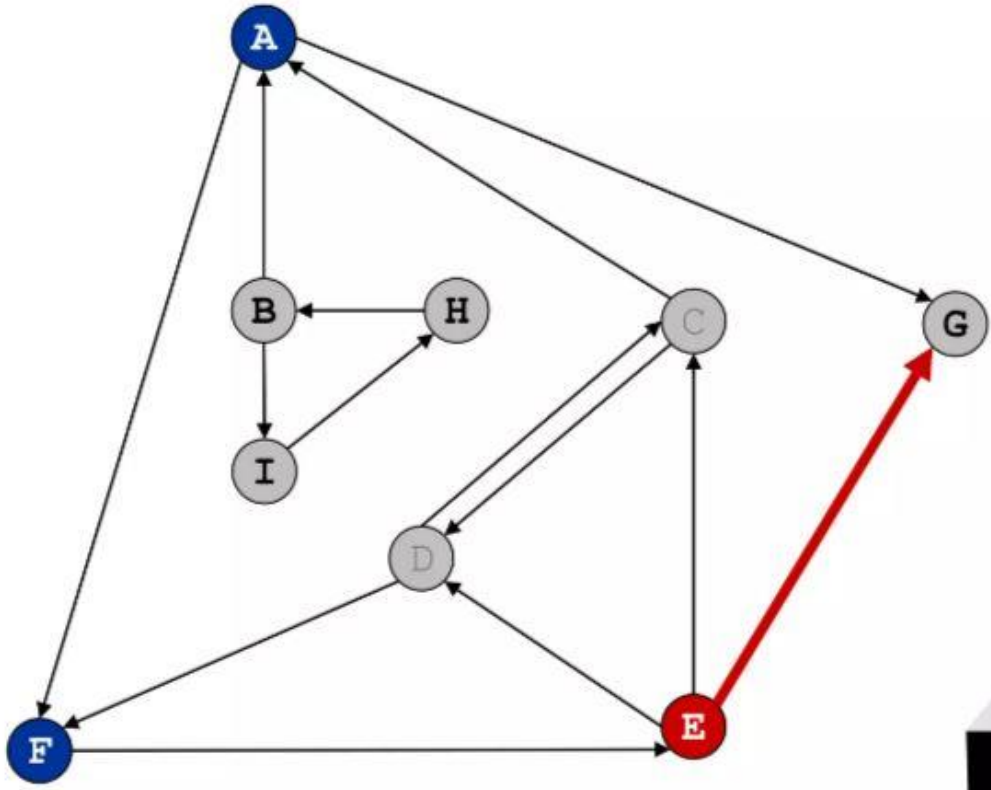


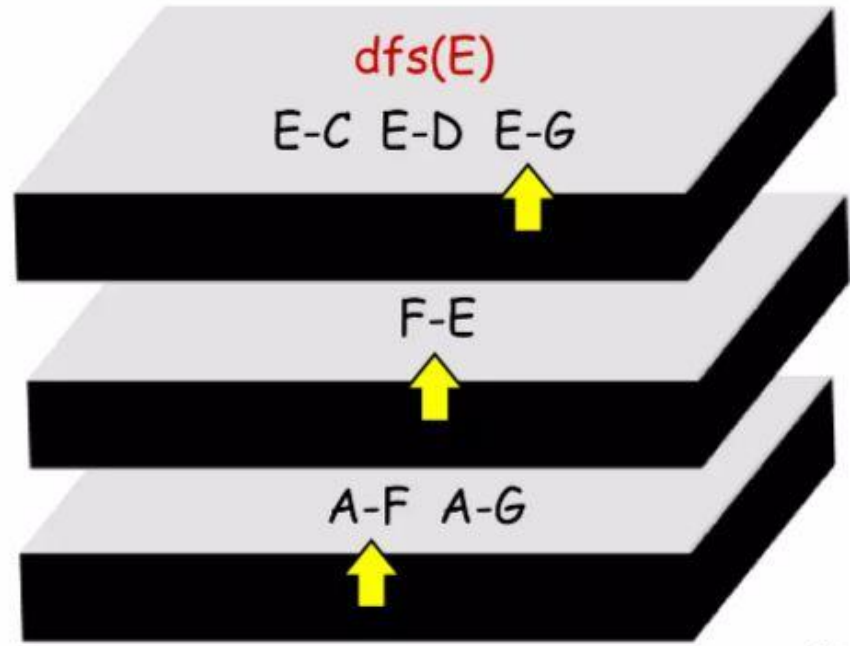
Function call stack:



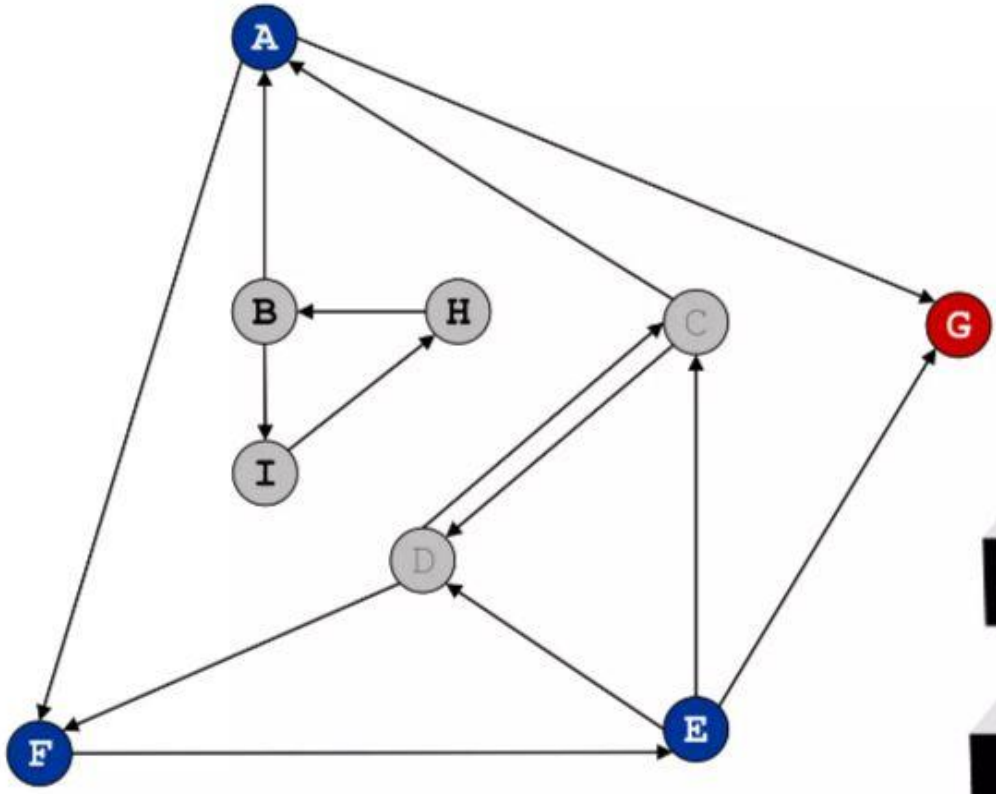


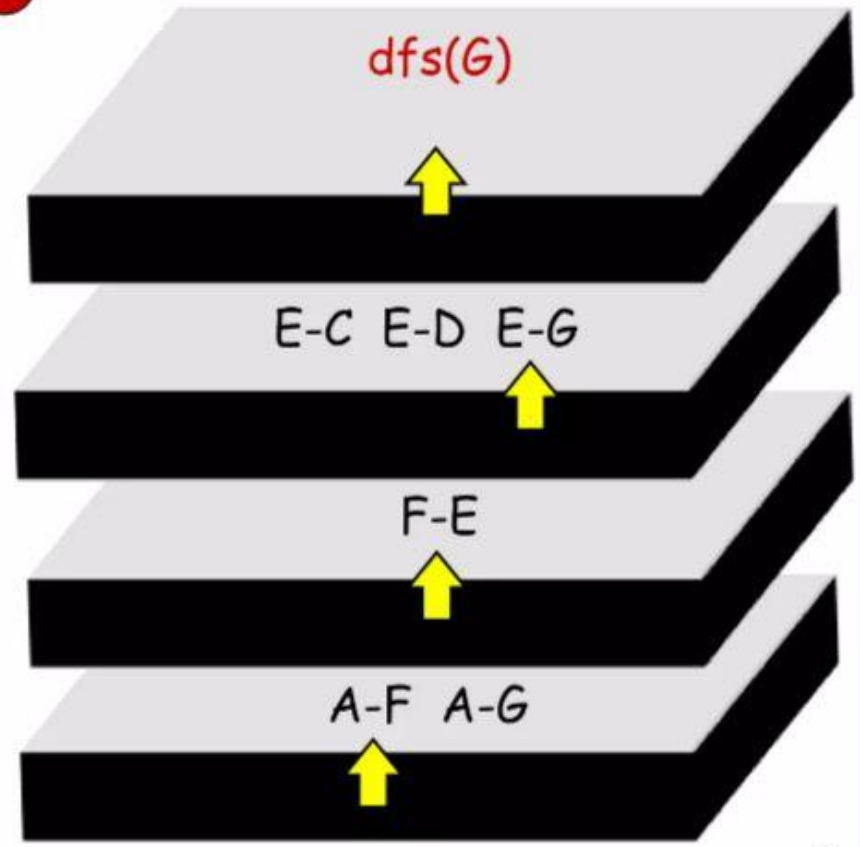
Function call stack:



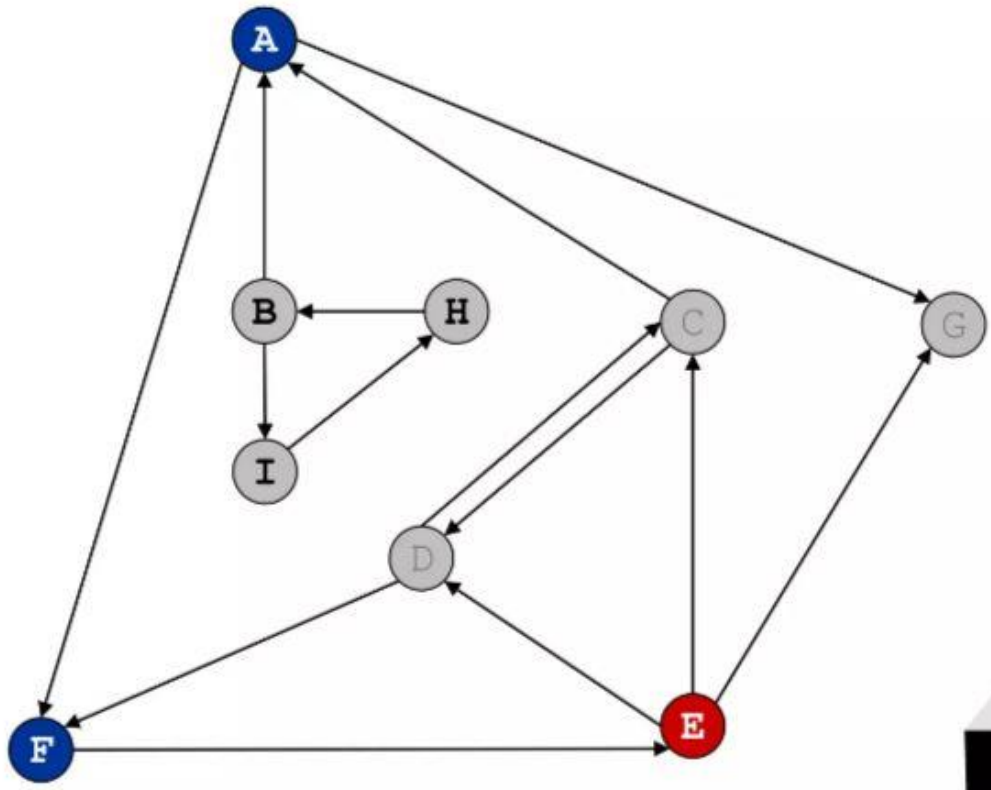


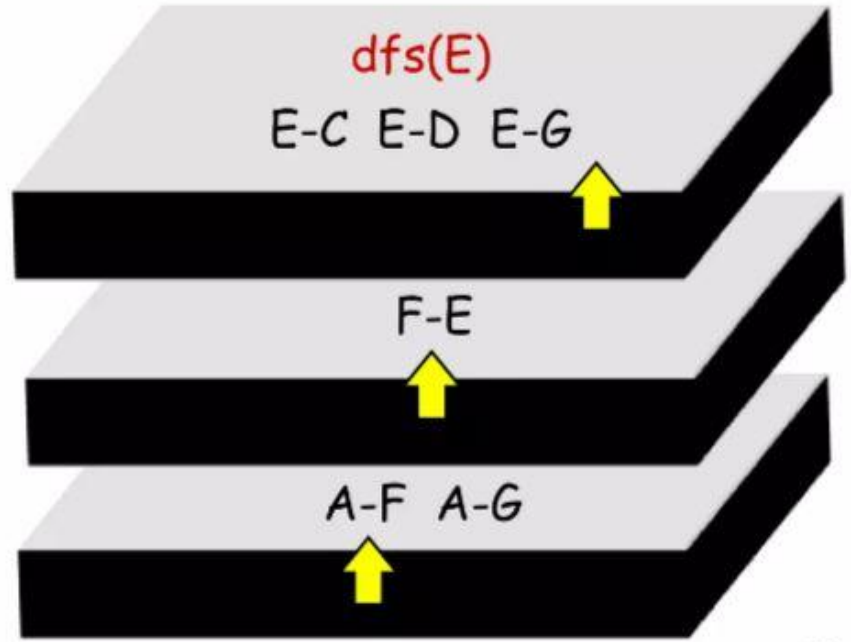
Function call stack:



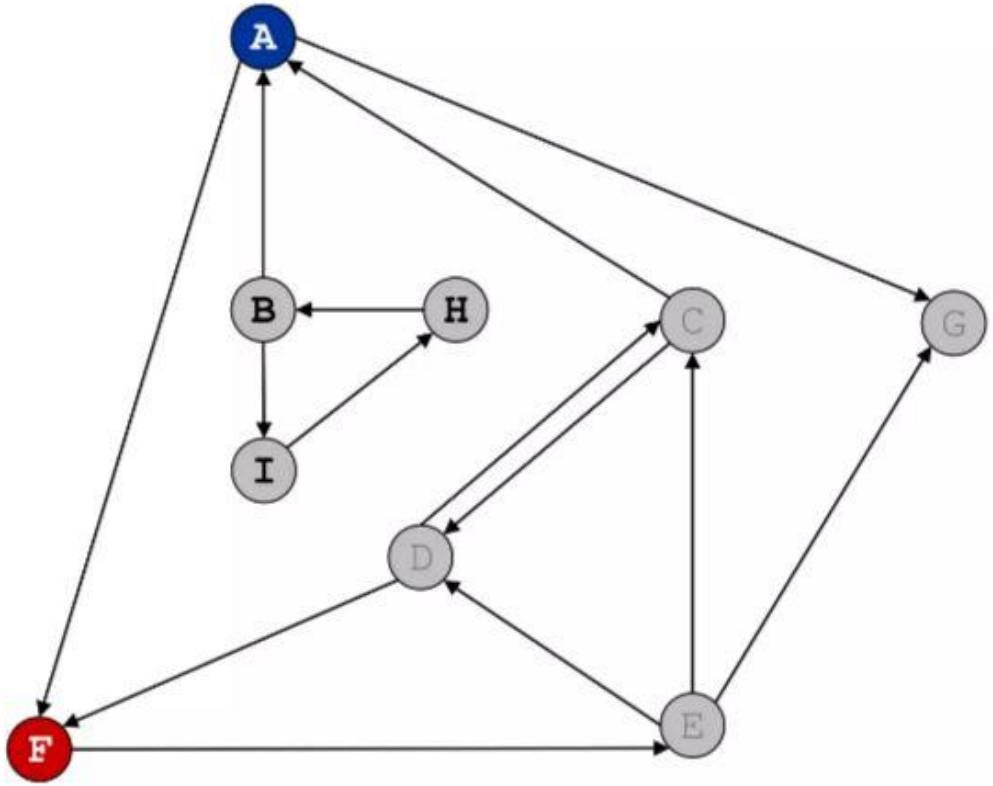


Function call stack:

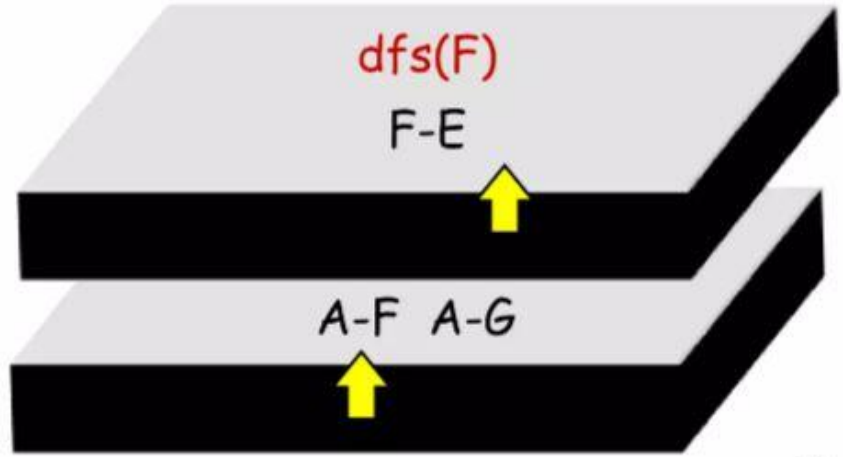


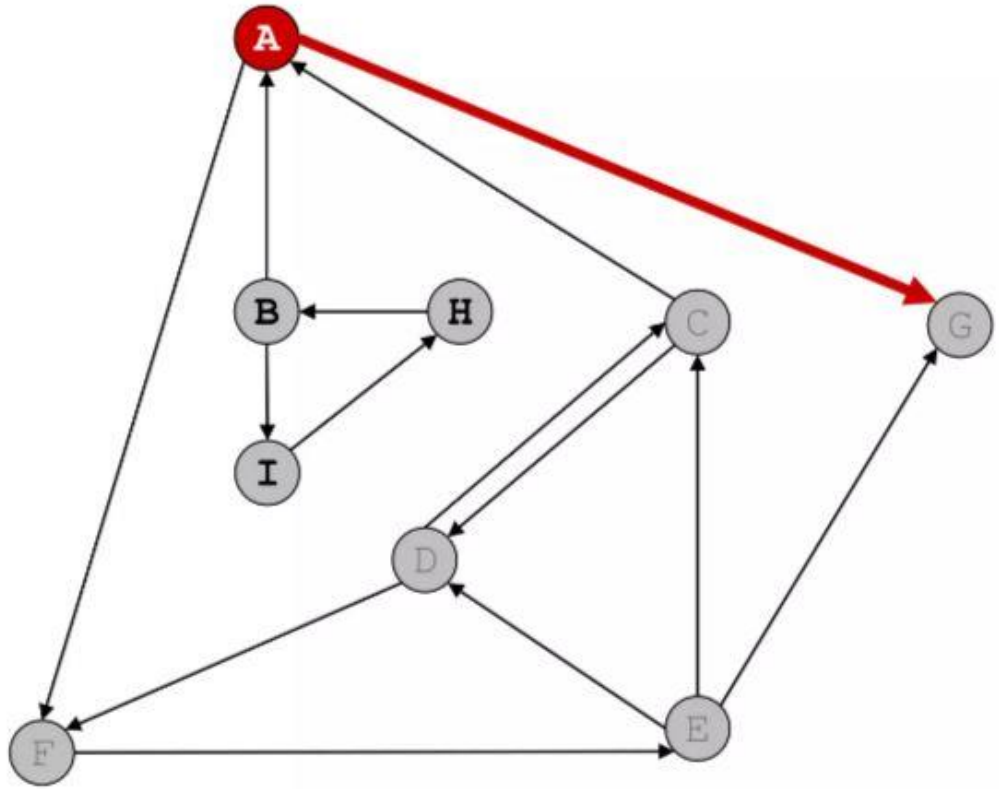


Function call stack:

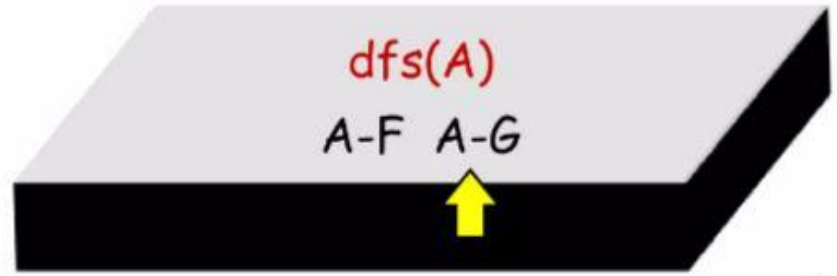


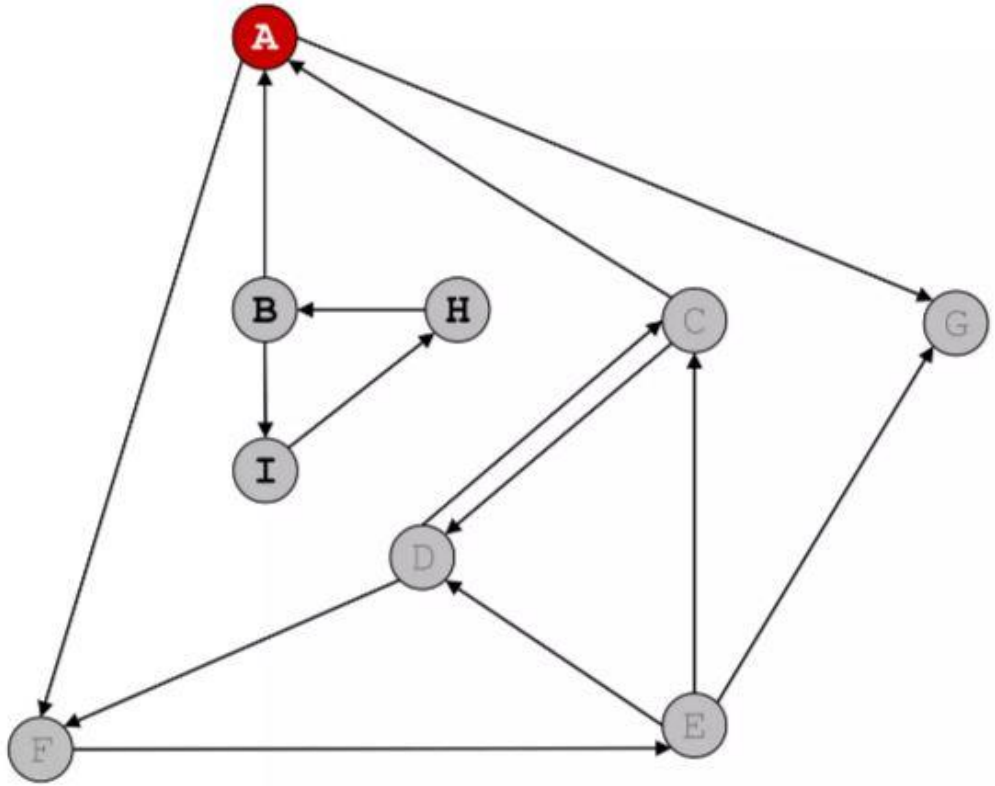
Function call stack:



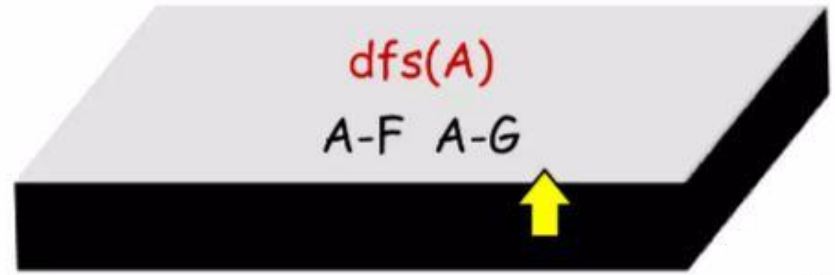


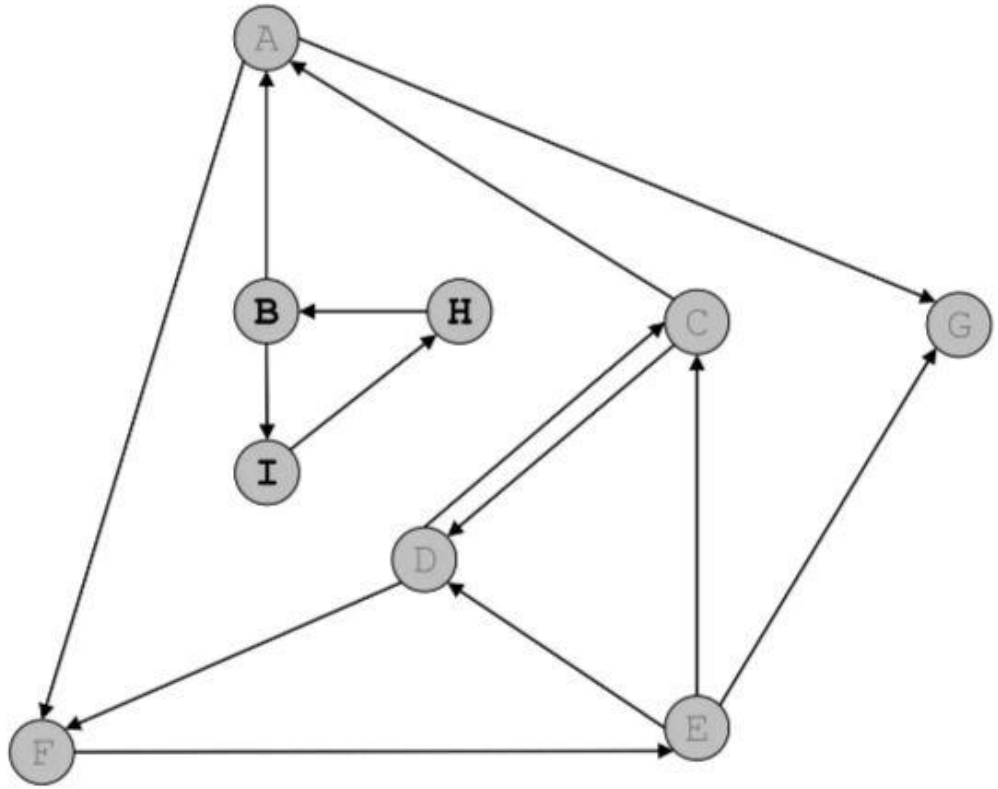
Function call stack:





Function call stack:





Nodes reachable from A: A, C, D, E, F, G