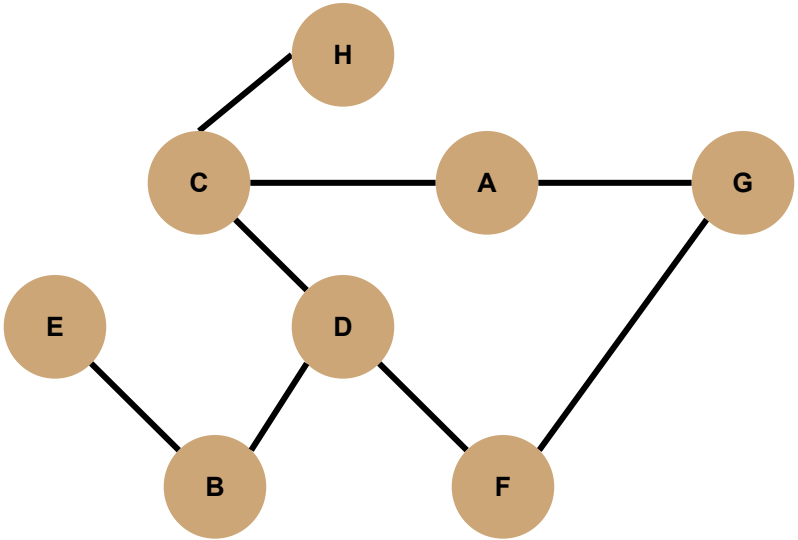
DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*



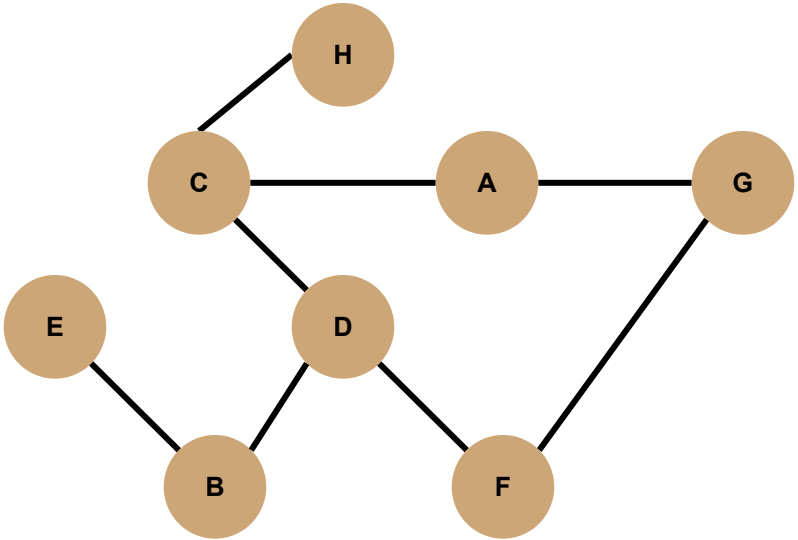
Order of DFS:

1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

Vertex	marked[]
A	F
B	F
C	F
D	F
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*



Order of DFS:

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	F
B	F
C	F
D	F
E	F
F	F
G	F
H	F

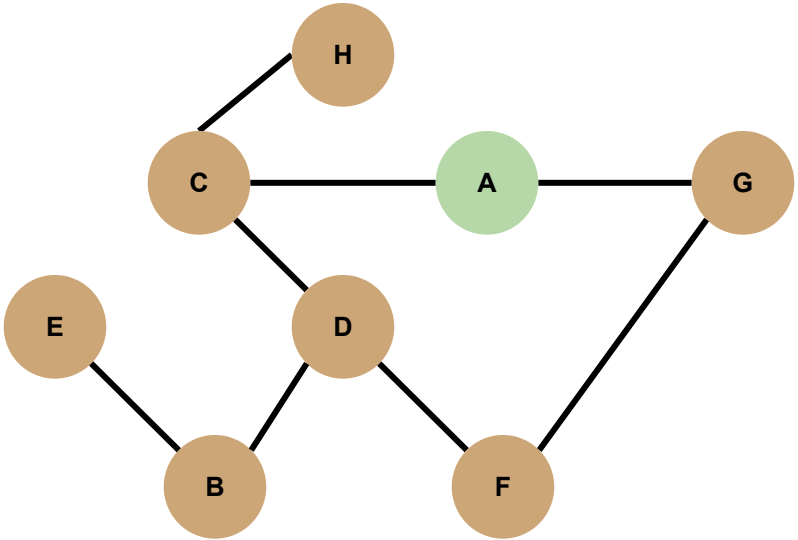
DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

DFS(A)



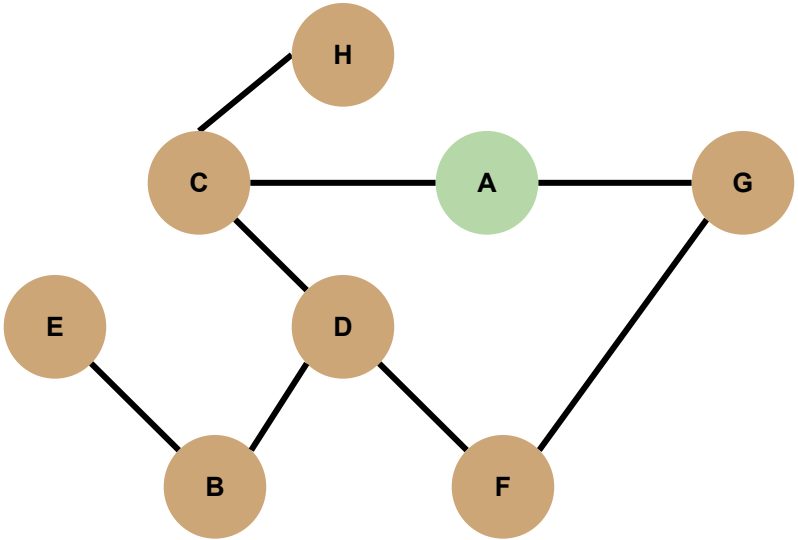
Vertex	marked[]
A	F
B	F
C	F
D	F
E	F
F	F
G	F
H	F

Order of DFS: A

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A)



Order of DFS: A

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	F
D	F
E	F
F	F
G	F
H	F

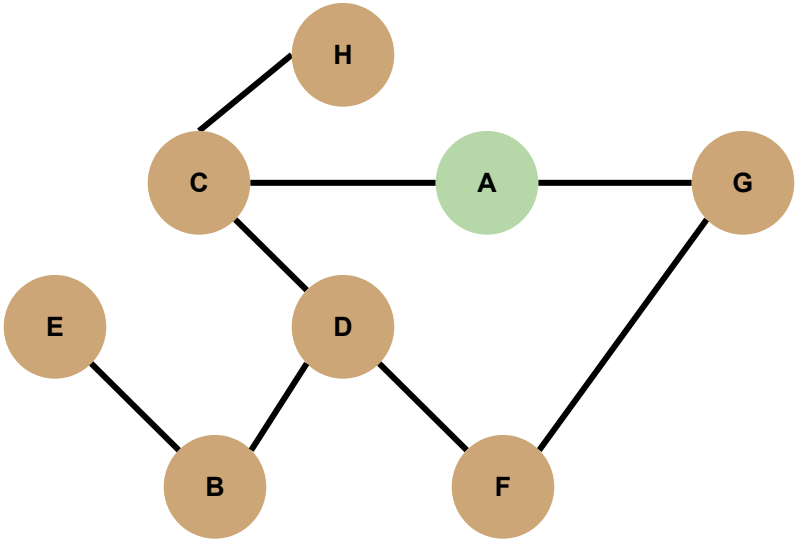
DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

DFS(A)



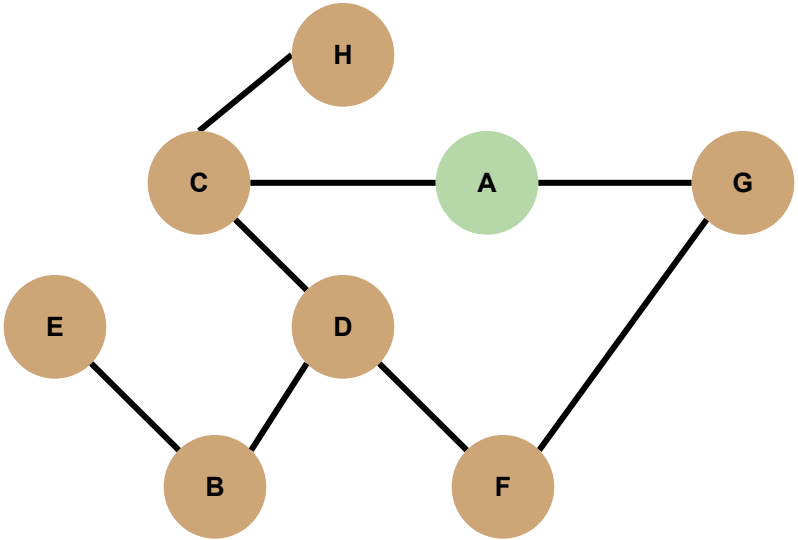
Vertex	marked[]
A	T
B	F
C	F
D	F
E	F
F	F
G	F
H	F

Order of DFS: A

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A)



Order of DFS: A

1. DFS(v):
- *Mark v*

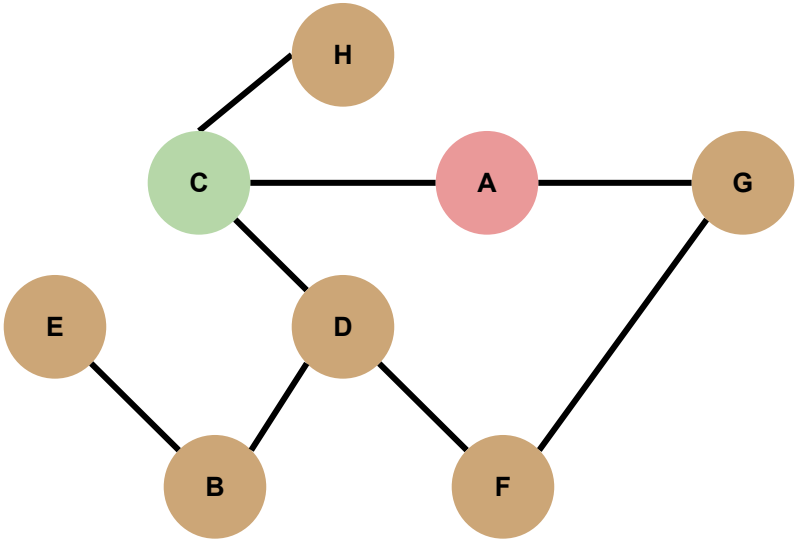
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	F
D	F
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C)



Order of DFS: A, C

1. DFS(v):
- *Mark v*

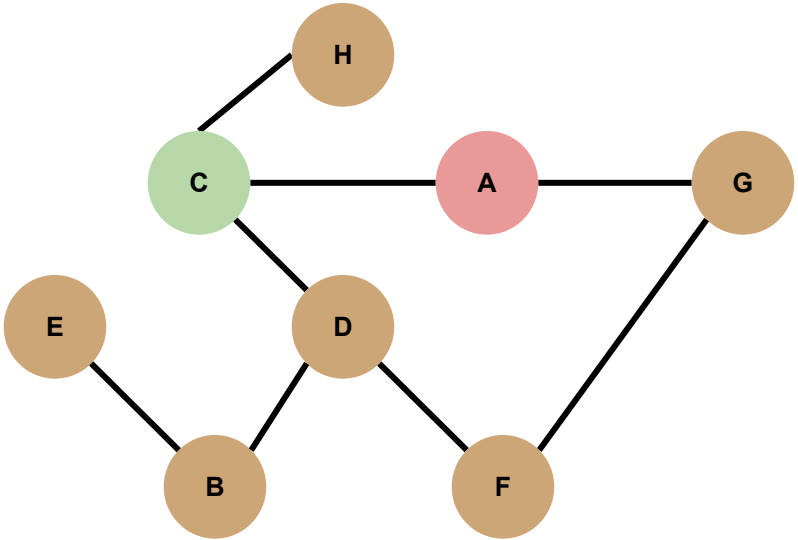
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	F
D	F
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C)



Order of DFS: A, C

1. DFS(v):
- *Mark v*

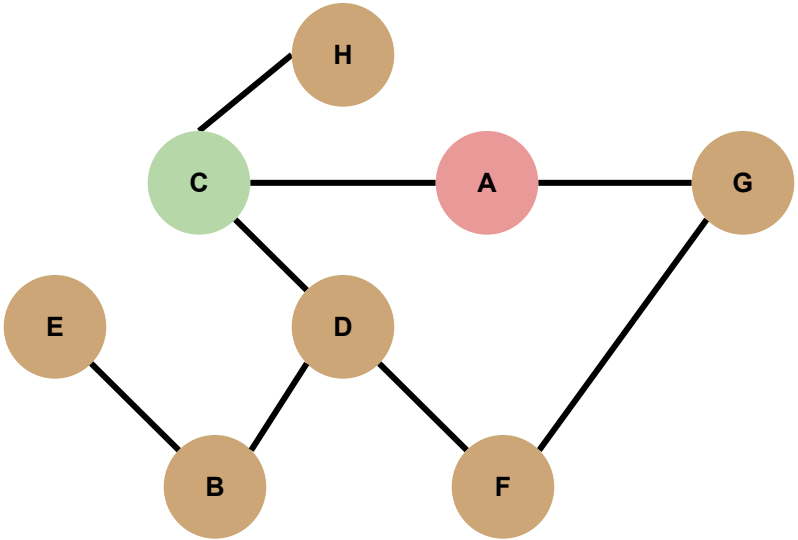
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	T
D	F
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C)



Order of DFS: A, C

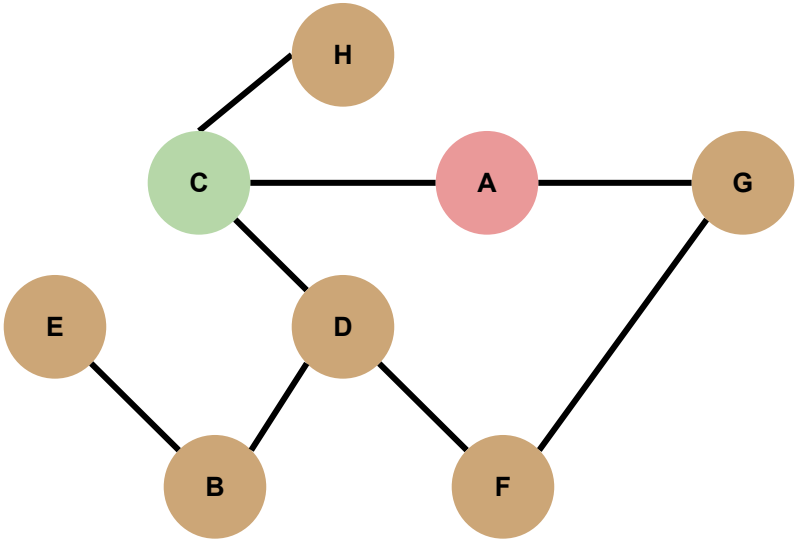
1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	T
D	F
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C)



Order of DFS: A, C

1. DFS(v):
- *Mark v*

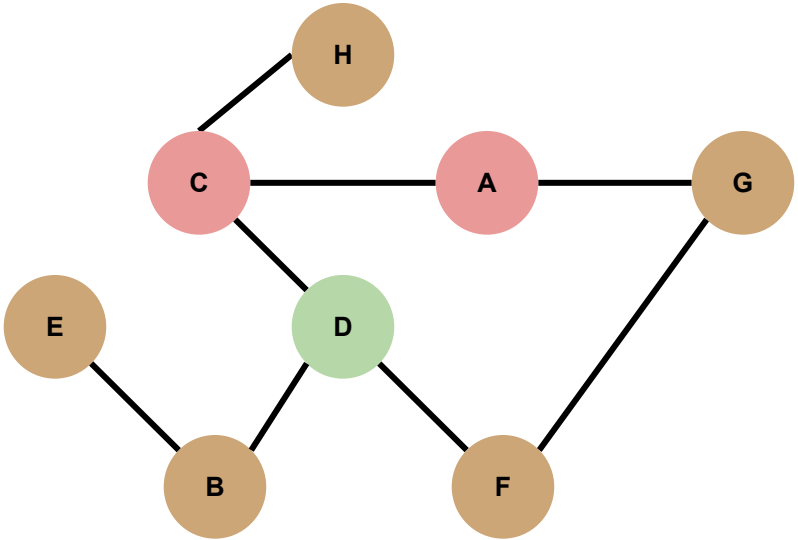
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	T
D	F
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D)



Order of DFS: A, C, D

1. DFS(v):
- *Mark v*

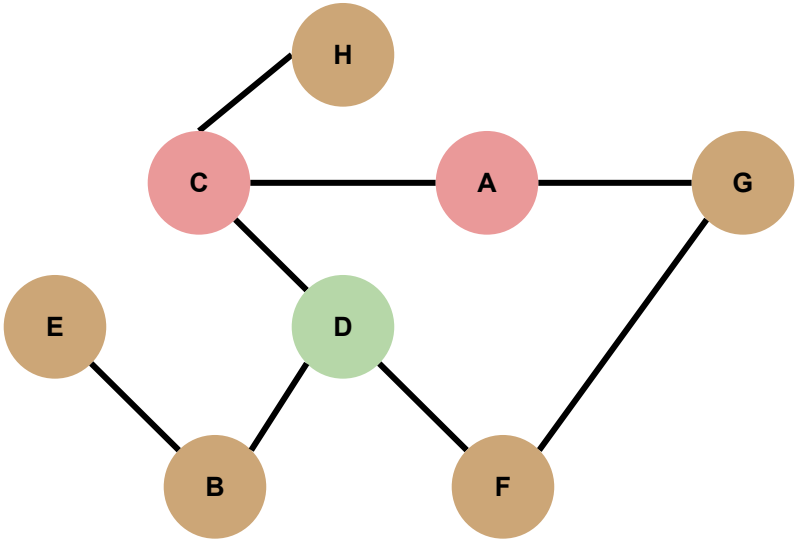
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	T
D	F
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D)



Order of DFS: A, C, D

1. DFS(v):
- *Mark v*

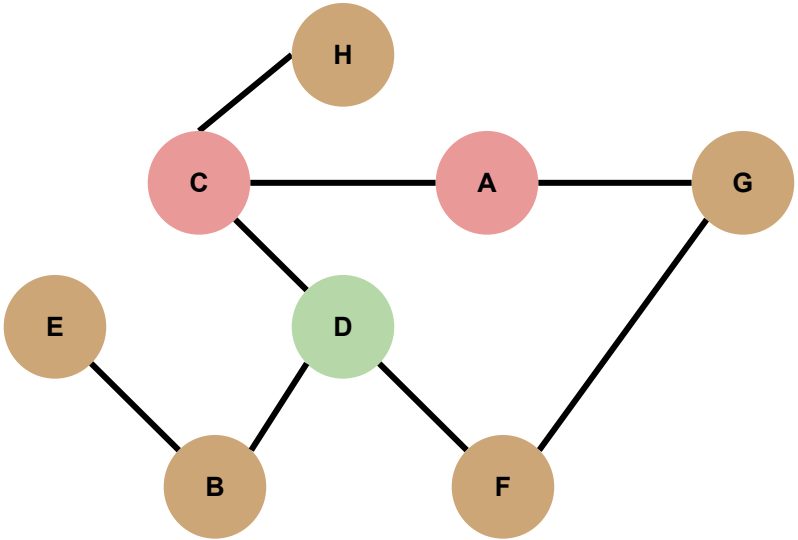
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	T
D	T
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D)



Order of DFS: A, C, D

1. DFS(v):
- *Mark v*

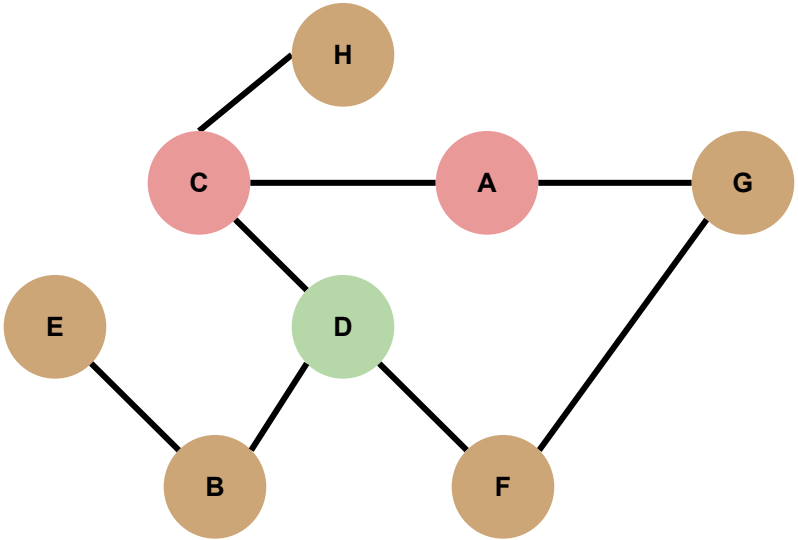
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	T
D	T
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D)



Order of DFS: A, C, D

1. DFS(v):
- *Mark v*

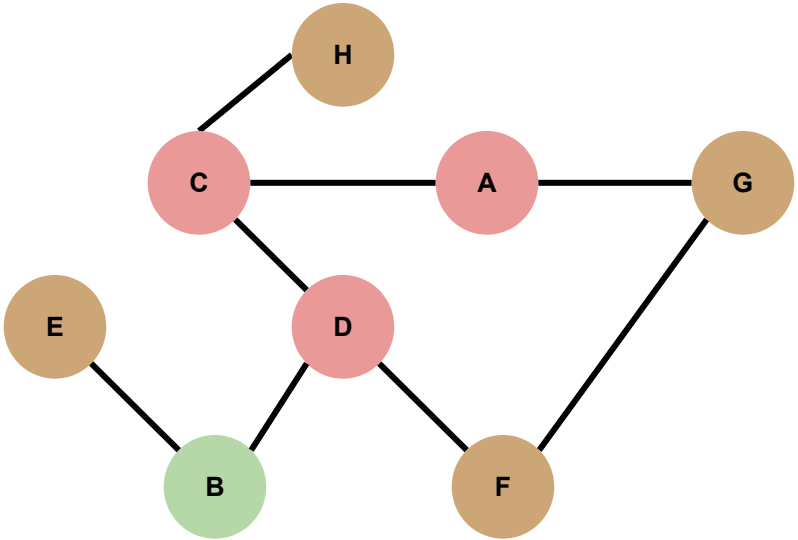
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	T
D	T
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B)



Order of DFS: A, C, D, B

1. DFS(v):
- *Mark v*

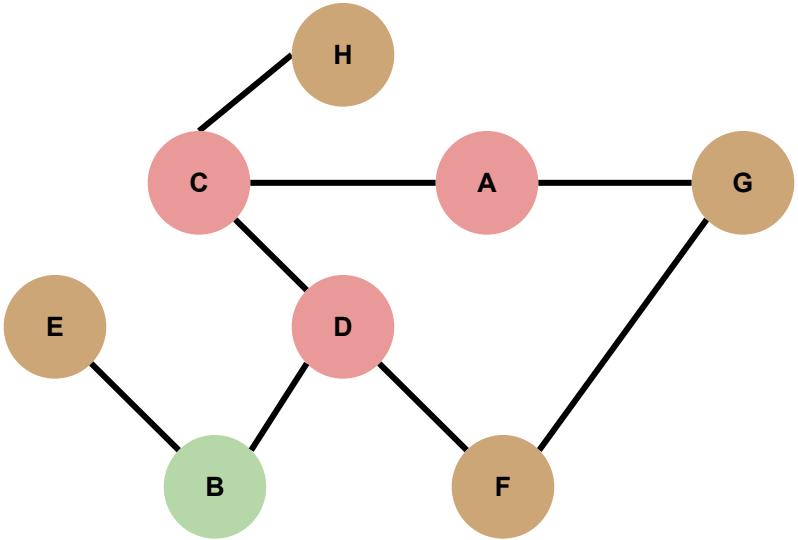
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	F
C	T
D	T
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B)



Order of DFS: A, C, D, B

1. DFS(v):
- *Mark v*

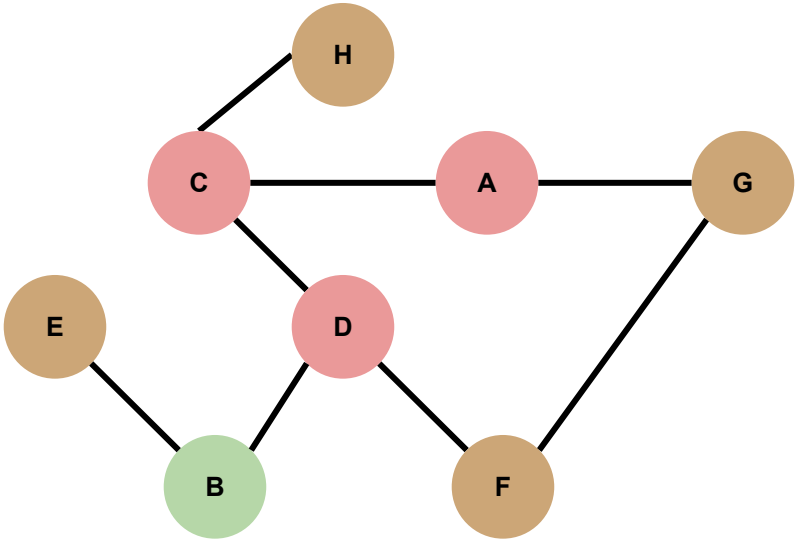
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B)



Order of DFS: A, C, D, B

1. DFS(v):
- *Mark v*

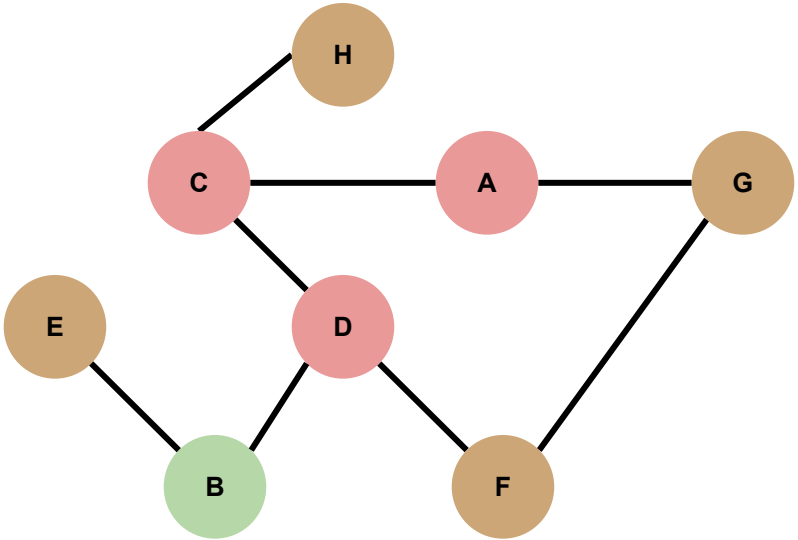
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B)



Order of DFS: A, C, D, B

1. DFS(v):
- *Mark v*

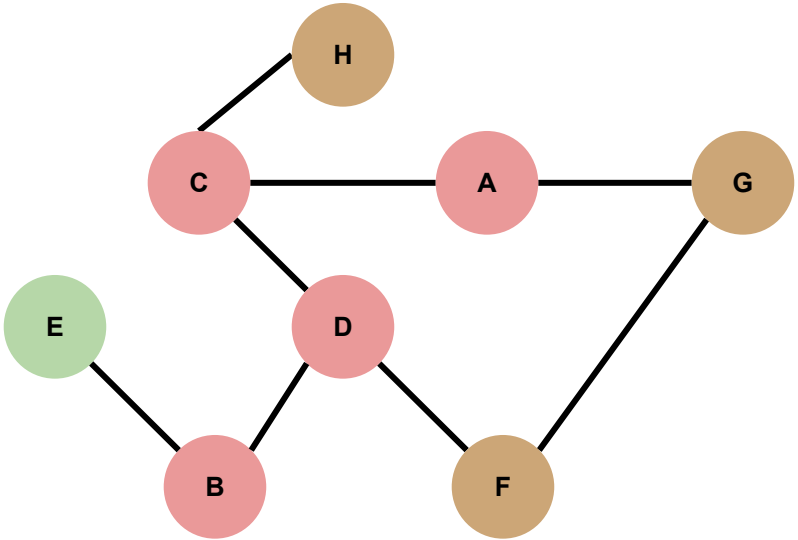
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B) → DFS(E)



Order of DFS: A, C, D, B, E

1. DFS(v):
- *Mark v*

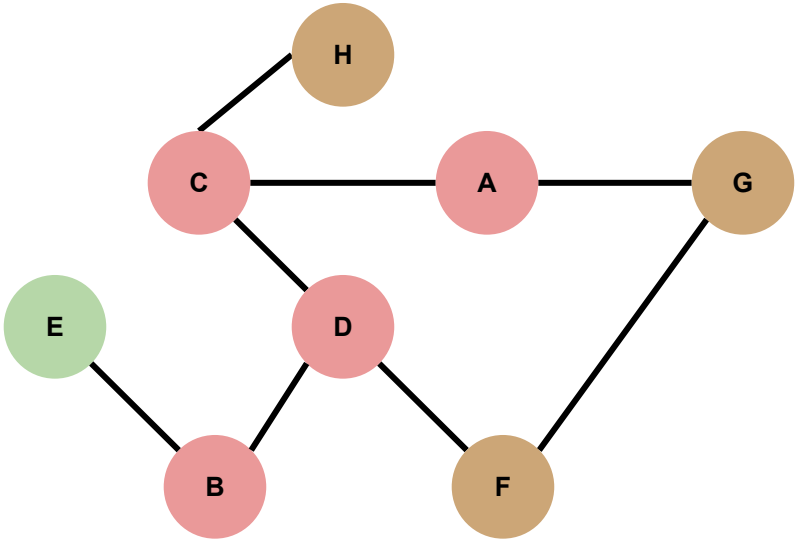
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	F
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B) → DFS(E)



Order of DFS: A, C, D, B, E

1. DFS(v):
- *Mark v*

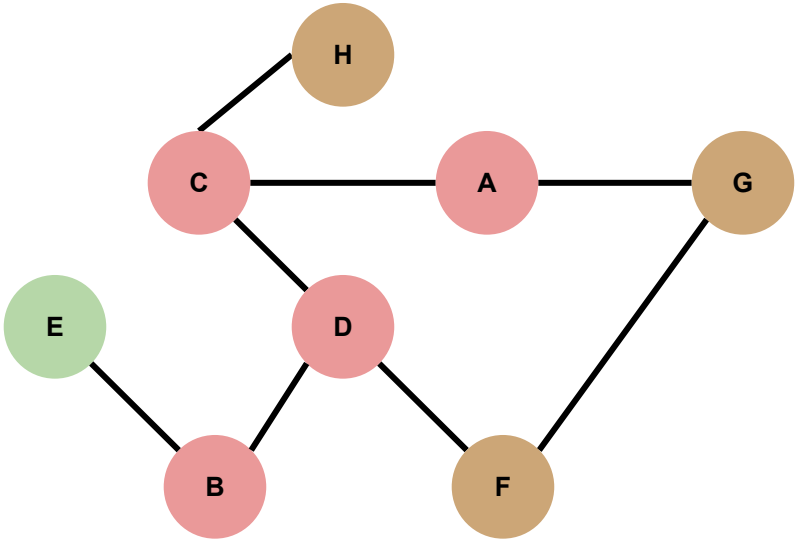
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B) → DFS(E)



Order of DFS: A, C, D, B, E

1. DFS(v):

- ***Mark v***

2. For every unmarked neighbor n:

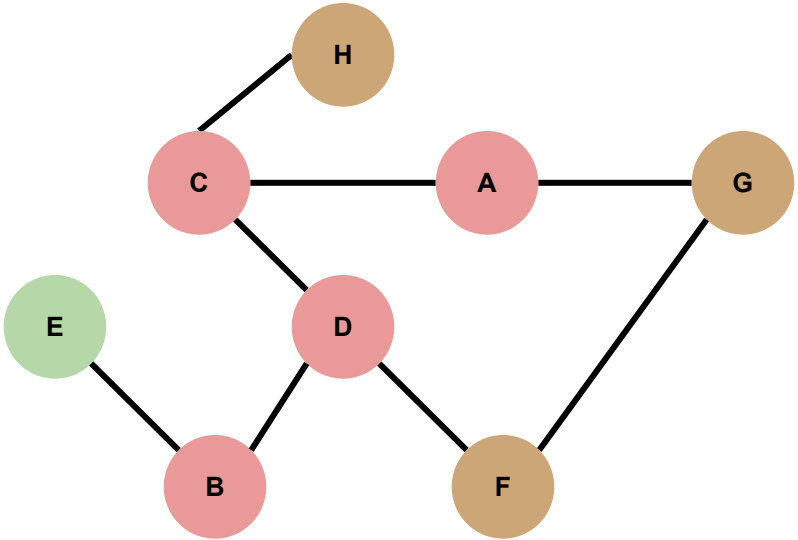
- ***Call DFS(n)***

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B) → ~~DFS(E)~~



Order of DFS: A, C, D, B, E

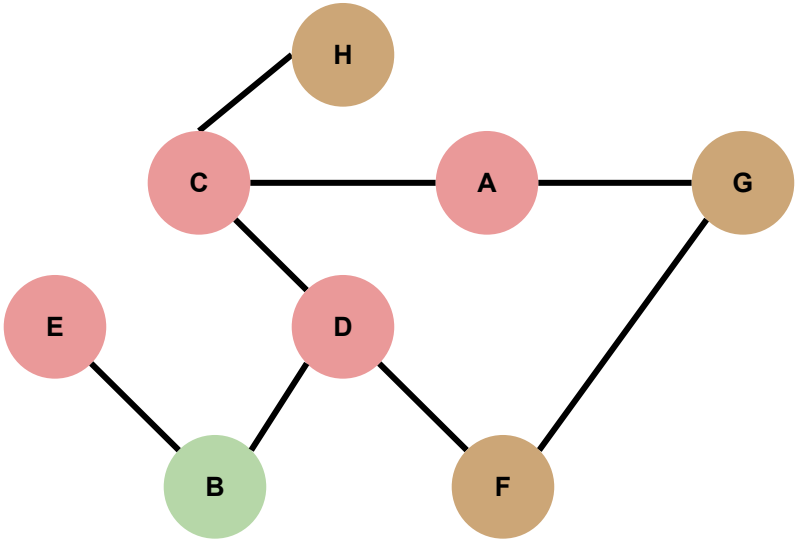
1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(B)



Order of DFS: A, C, D, B, E

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

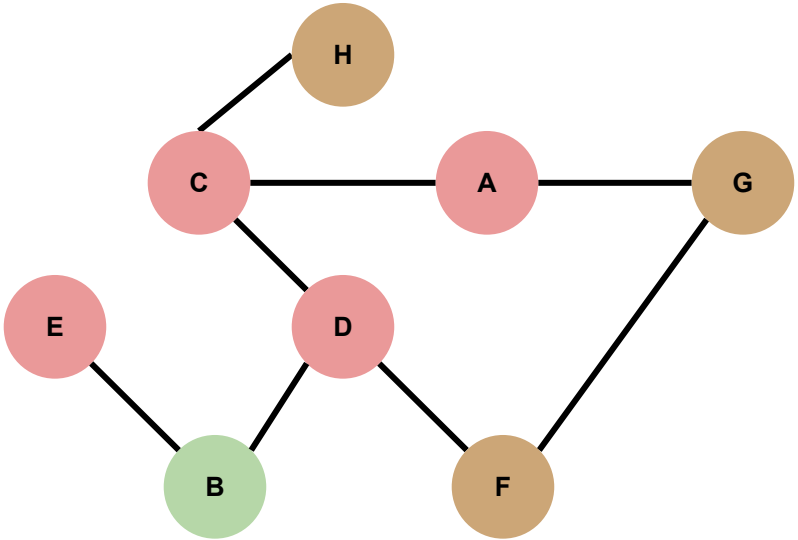
Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

DFS(A) → DFS(C) → DFS(D) → ~~DFS(B)~~



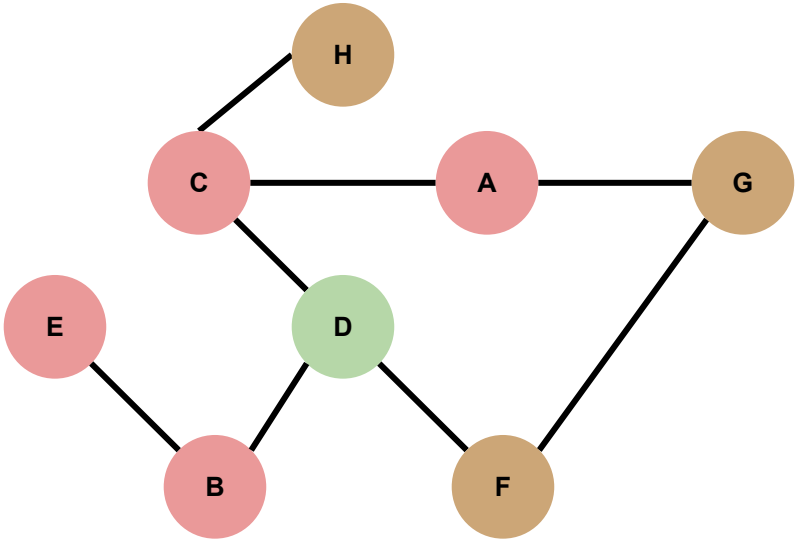
Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	F
G	F
H	F

Order of DFS: A, C, D, B, E

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D)



Order of DFS: A, C, D, B, E

1. DFS(v):
- *Mark v*

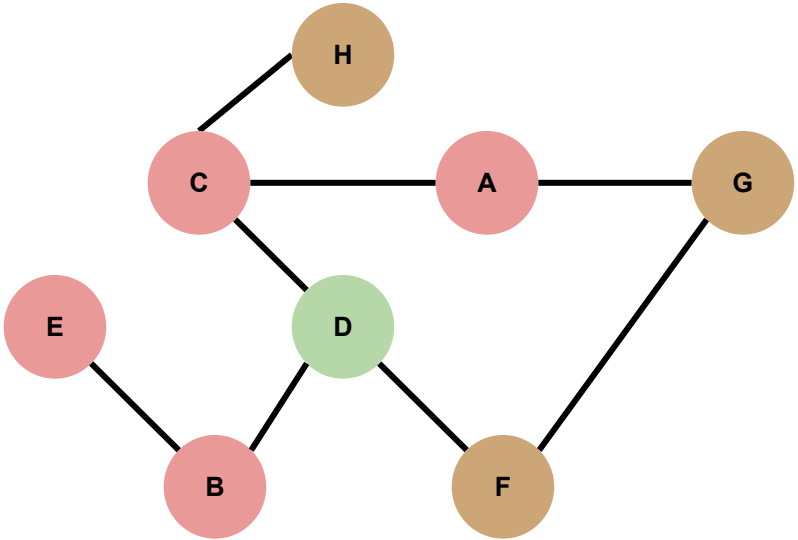
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D)



Order of DFS: A, C, D, B, E

1. DFS(v):
- *Mark v*

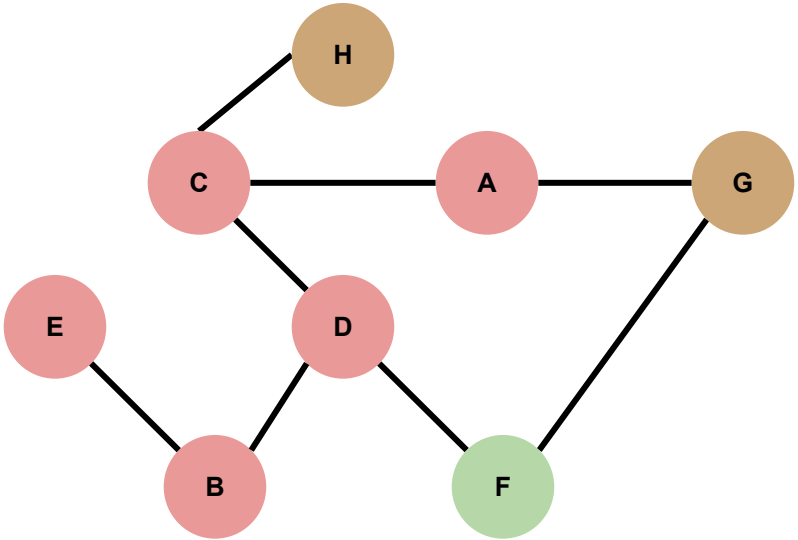
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F)



Order of DFS: A, C, D, B, E, F

1. DFS(v):
- *Mark v*

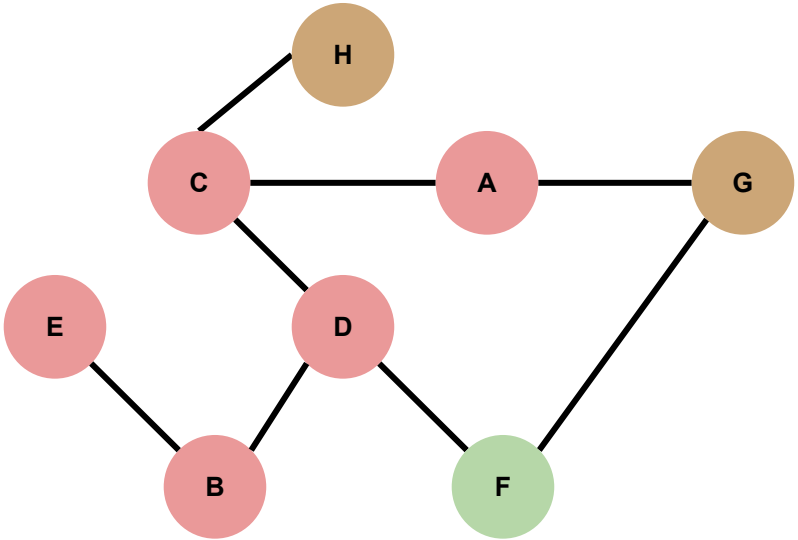
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	F
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F)



Order of DFS: A, C, D, B, E, F

1. DFS(v):
- *Mark v*

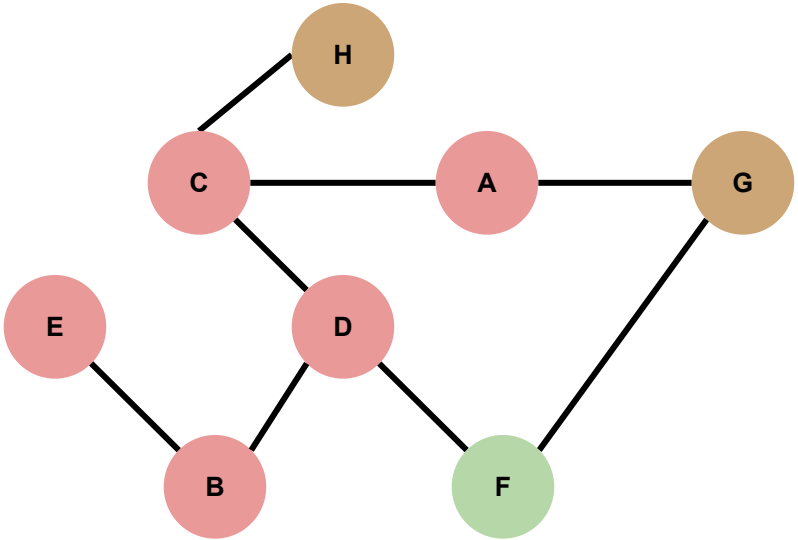
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F)



Order of DFS: A, C, D, B, E, F

1. DFS(v):
- *Mark v*

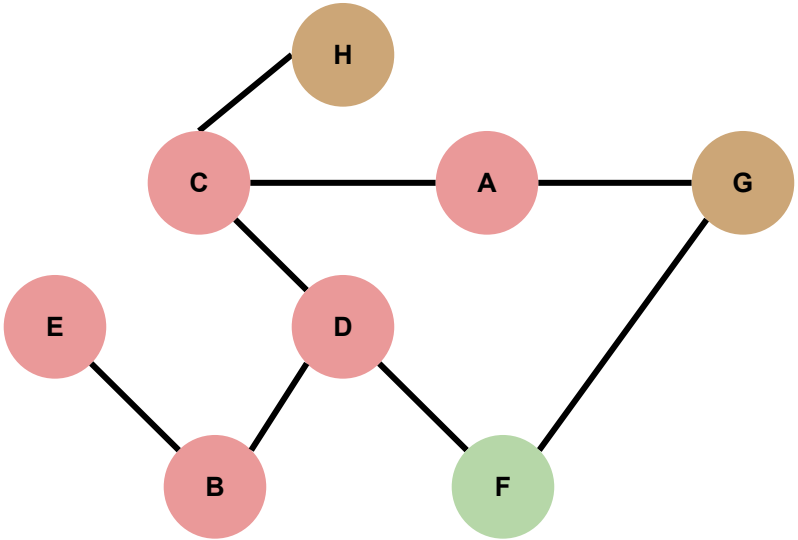
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F)



Order of DFS: A, C, D, B, E, F

1. DFS(v):
- *Mark v*

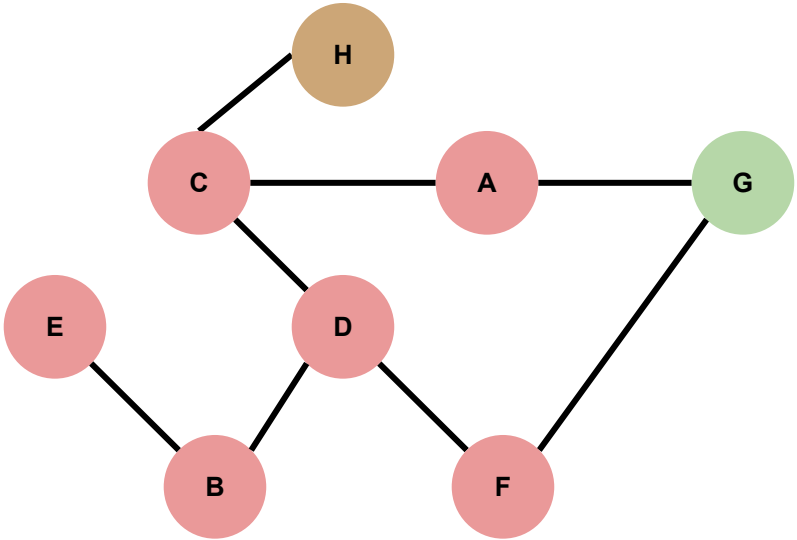
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F) → DFS(G)



Order of DFS: A, C, D, B, E, F, G

1. DFS(v):
- *Mark v*

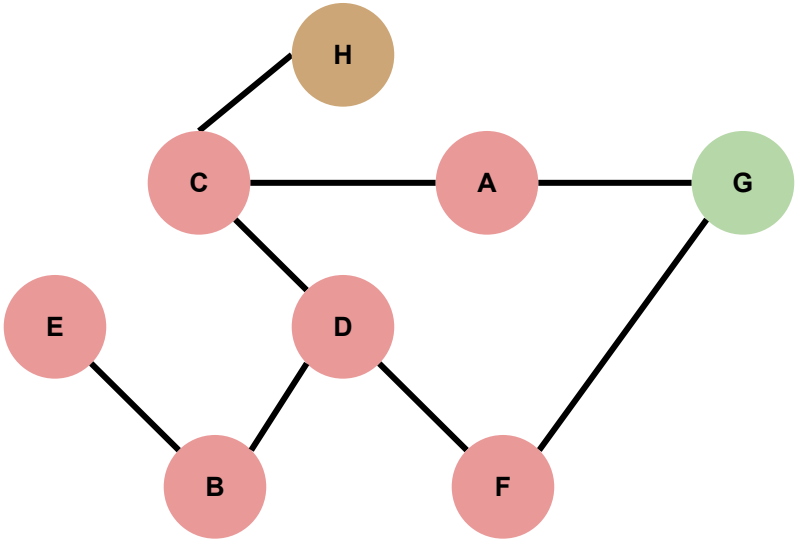
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	F
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F) → DFS(G)



Order of DFS: A, C, D, B, E, F, G

1. DFS(v):
- *Mark v*

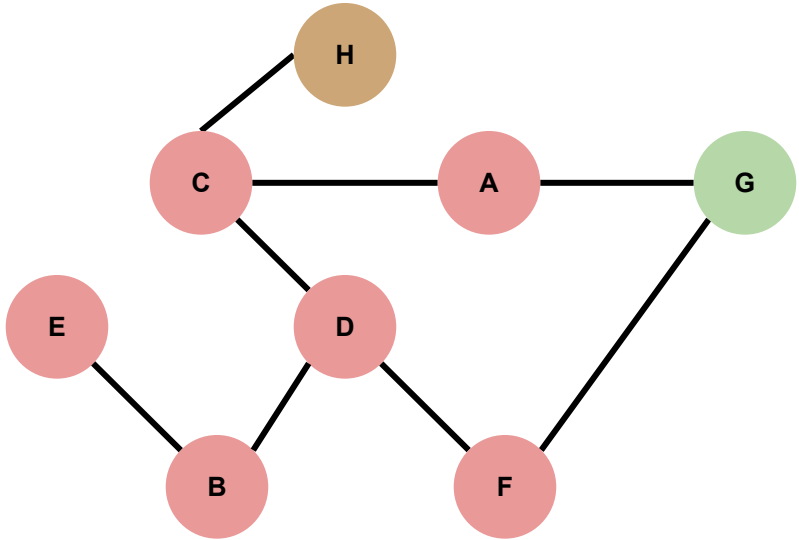
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F) → DFS(G)



Order of DFS: A, C, D, B, E, F, G

1. DFS(v):
- *Mark v*

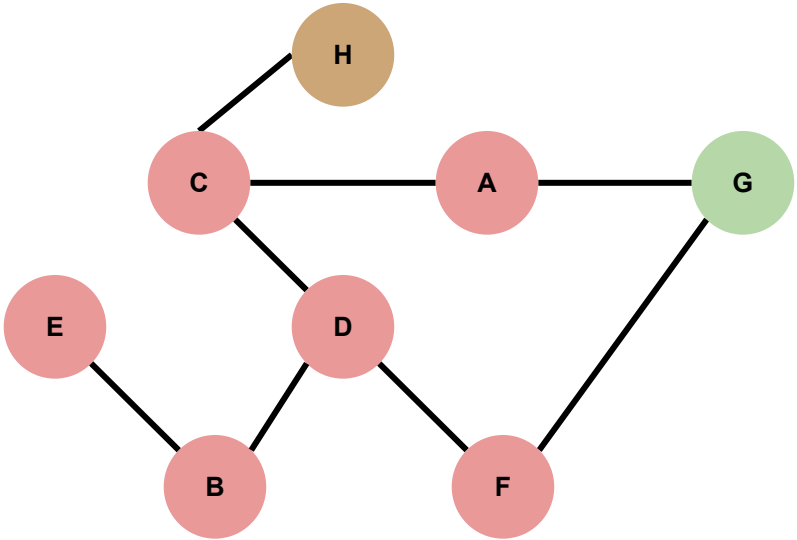
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F) → ~~DFS(G)~~



Order of DFS: A, C, D, B, E, F, G

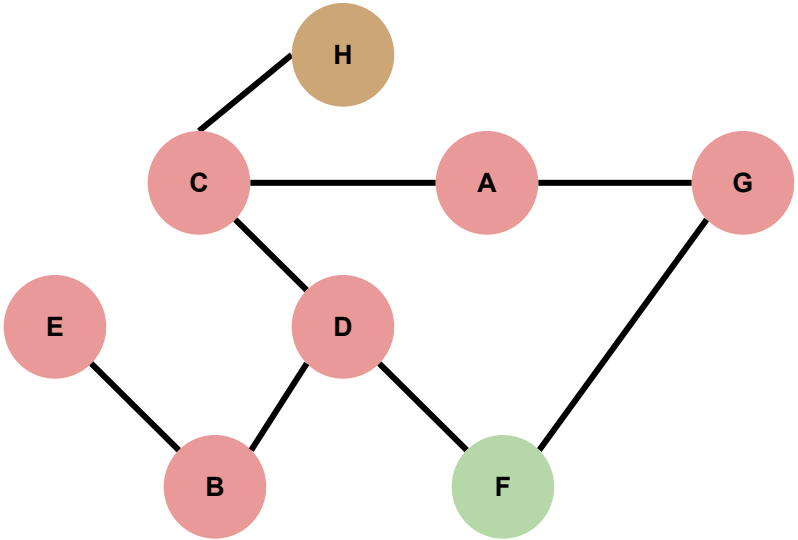
1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → DFS(F)



Order of DFS: A, C, D, B, E, F, G

1. DFS(v):
- *Mark v*

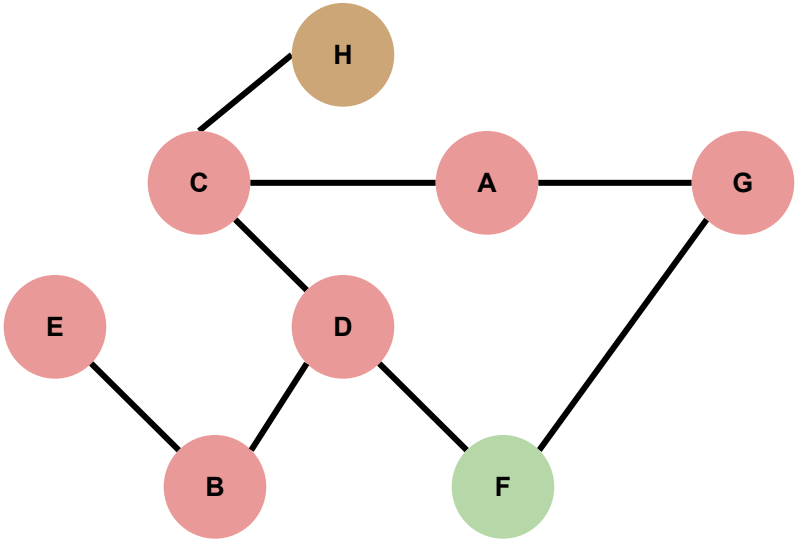
2. For every unmarked neighbor n:
- *Set n's edgeTo value to v*
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D) → ~~DFS(F)~~



Order of DFS: A, C, D, B, E, F, G

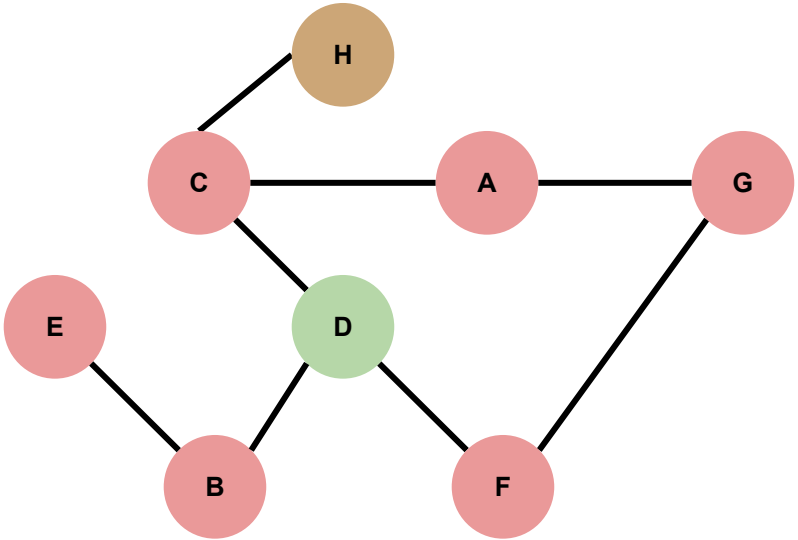
1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(D)



Order of DFS: A, C, D, B, E, F, G

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

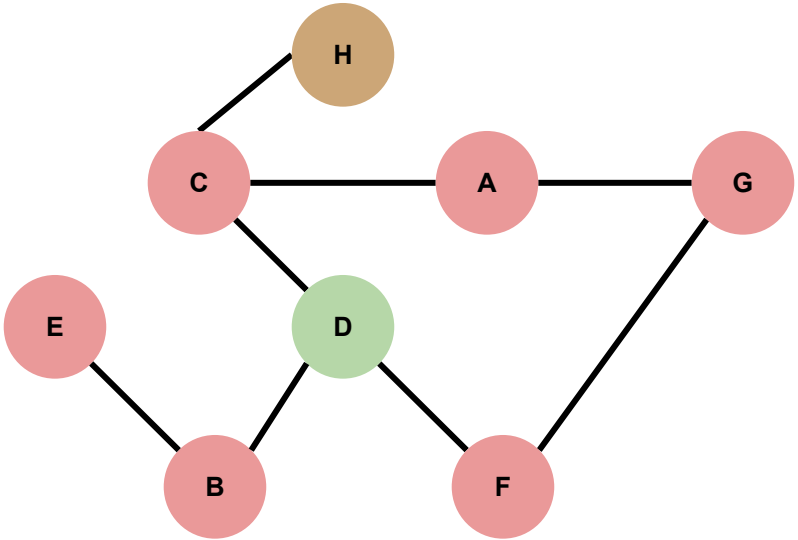
Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

DFS(A) → DFS(C) → ~~DFS(D)~~



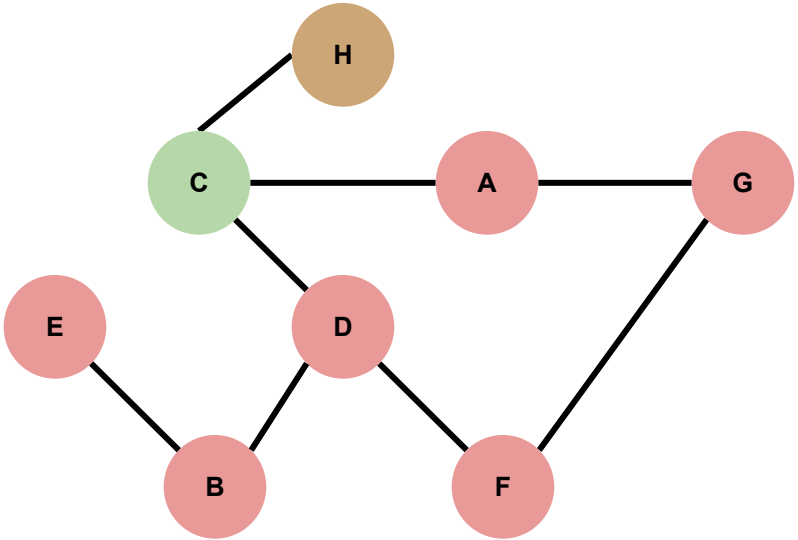
Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

Order of DFS: A, C, D, B, E, F, G

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C)



Order of DFS: A, C, D, B, E, F, G

1. DFS(v):
- *Mark v*

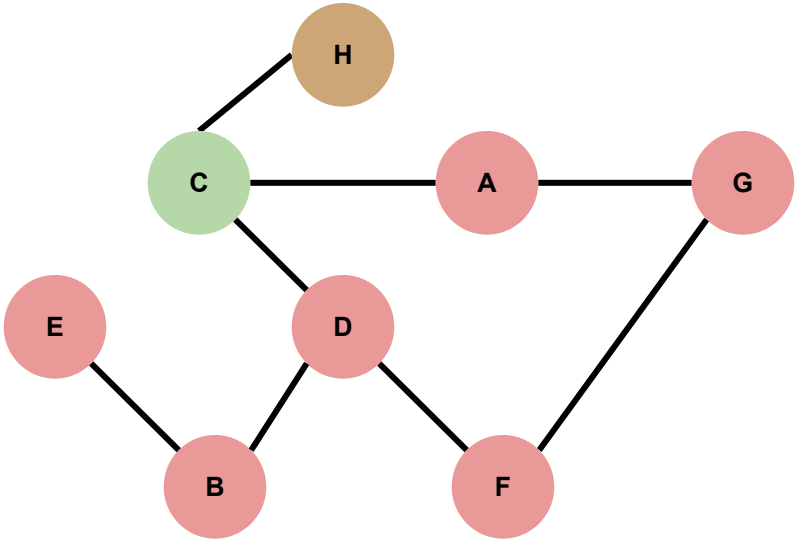
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C)



Order of DFS: A, C, D, B, E, F, G

1. DFS(v):
- *Mark v*

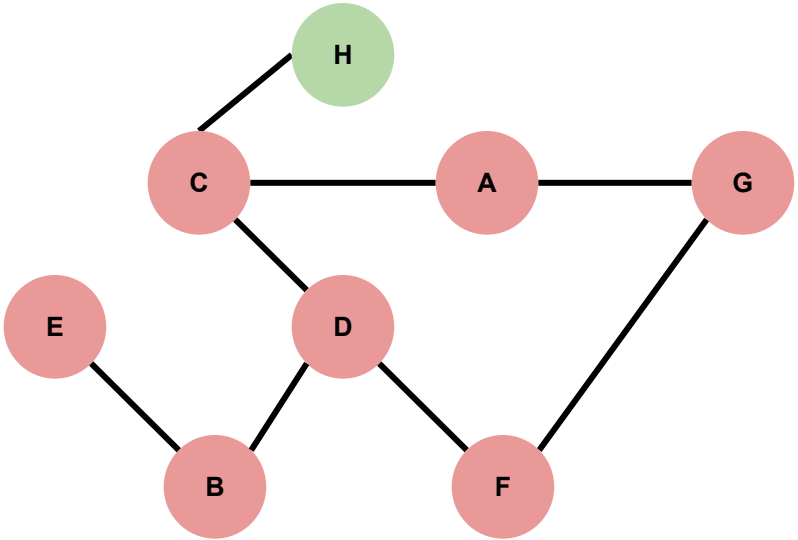
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(H)



Order of DFS: A, C, D, B, E, F, G, H

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	F

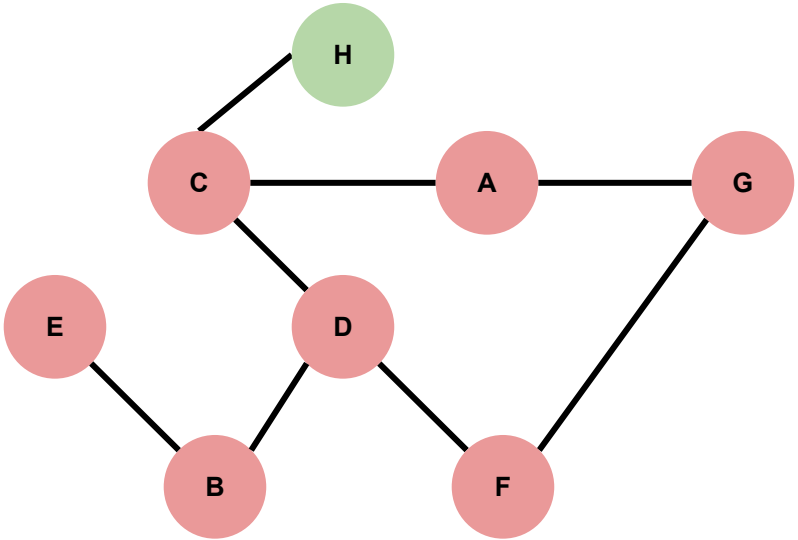
DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

DFS(A) → DFS(C) → DFS(H)



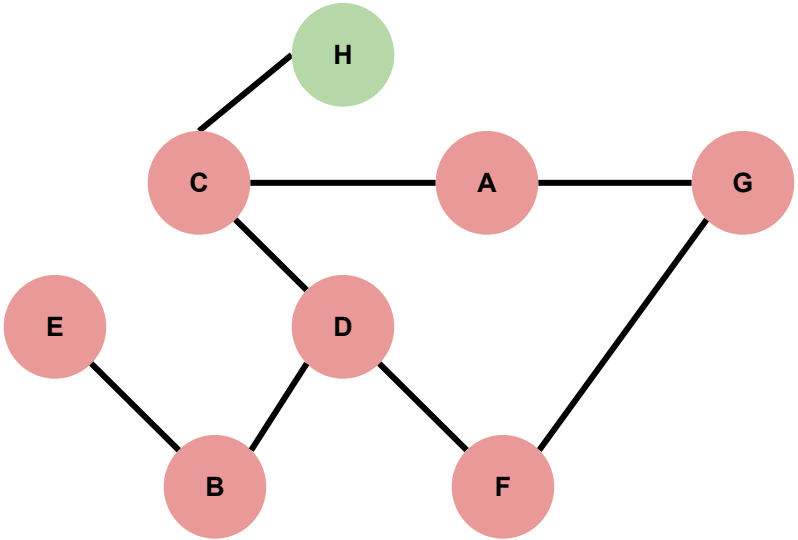
Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	T

Order of DFS: A, C, D, B, E, F, G, H

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → DFS(H)



Order of DFS: A, C, D, B, E, F, G, H

1. DFS(v):
- *Mark v*

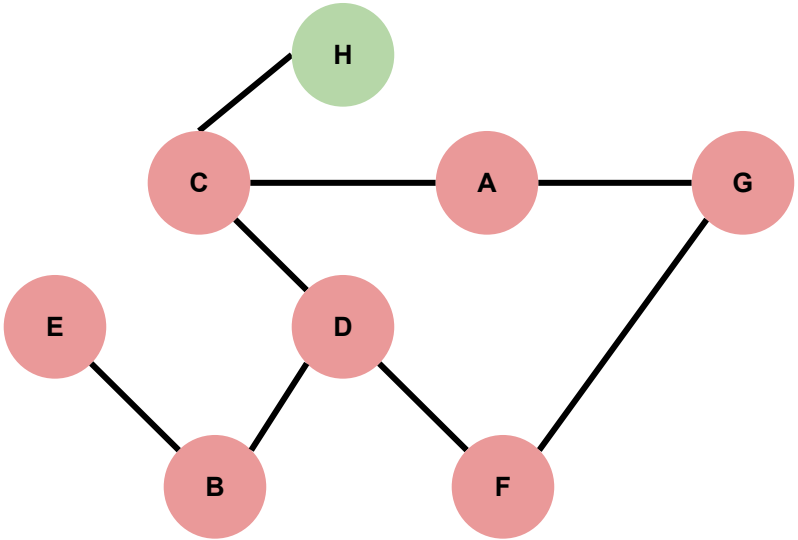
2. For every unmarked neighbor n:
- *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	T

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C) → ~~DFS(H)~~



Order of DFS: A, C, D, B, E, F, G, H

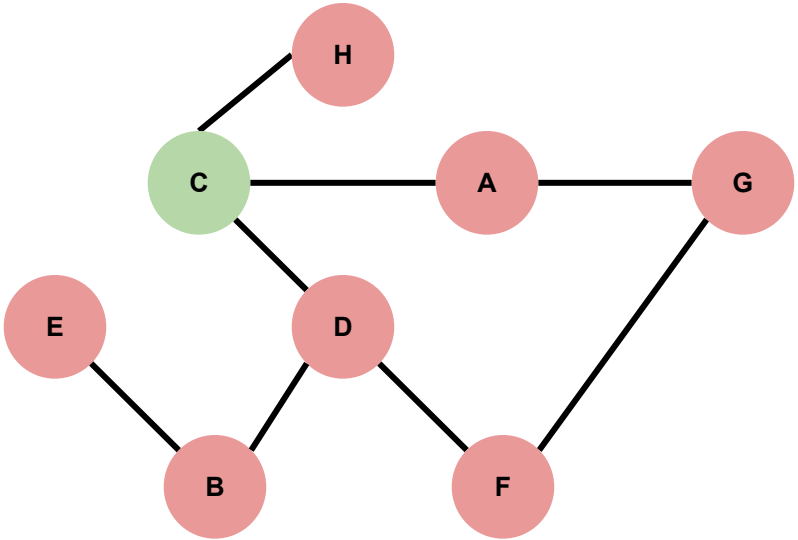
1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	T

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A) → DFS(C)



Order of DFS: A, C, D, B, E, F, G, H

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

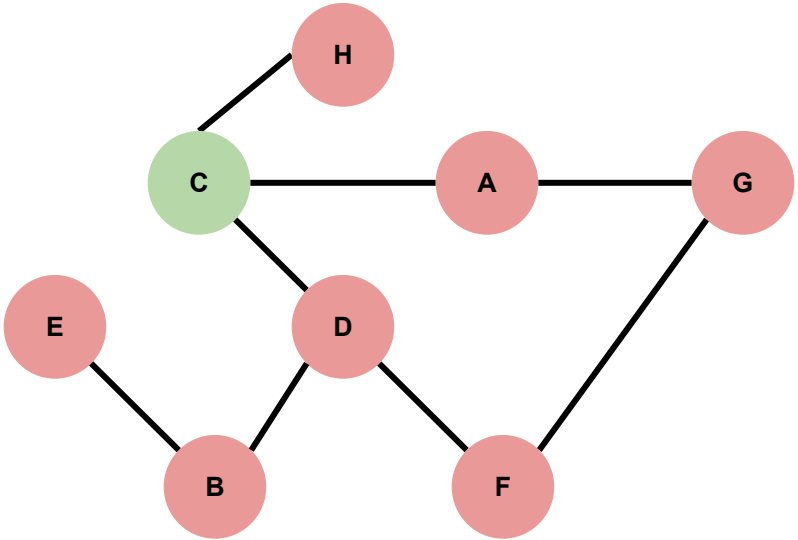
Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	T

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

DFS(A) → ~~DFS(C)~~



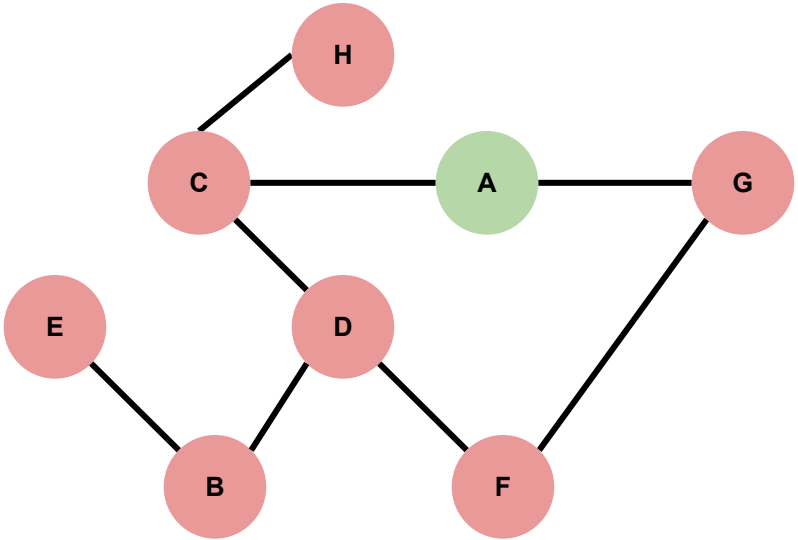
Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	T

Order of DFS: A, C, D, B, E, F, G, H

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

DFS(A)



Order of DFS: A, C, D, B, E, F, G, H

1. DFS(v):
- *Mark v*

2. For every unmarked neighbor n:
- *Call DFS(n)*

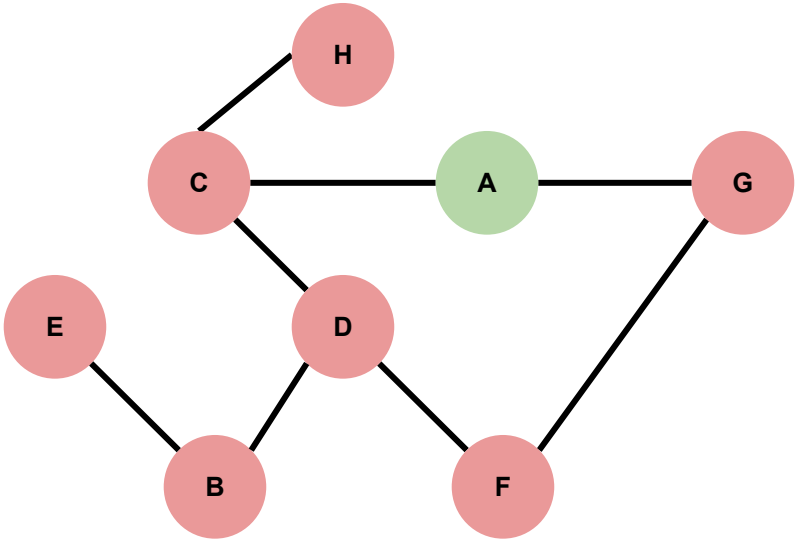
Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	T

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*

1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

~~***DFS(A)***~~

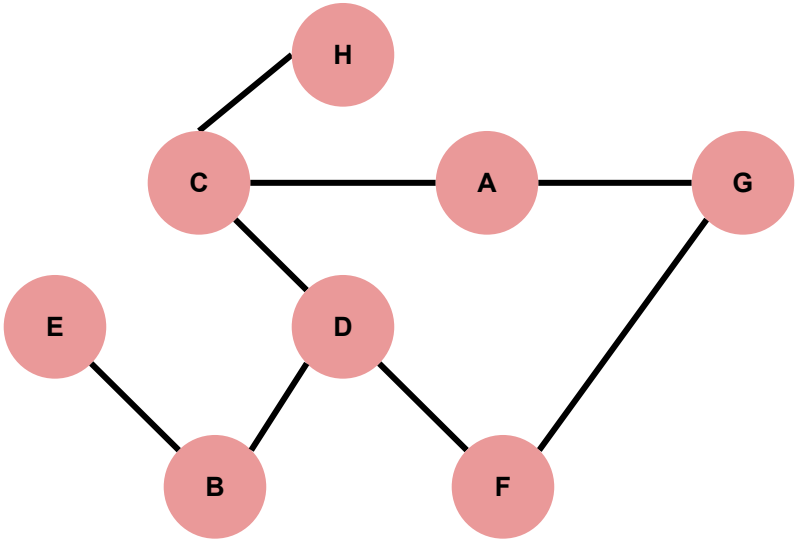


Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	T

Order of DFS: A, C, D, B, E, F, G, H

DFS

*Starting from A, write the order in which vertices are visited using **DFS**.*



Order of DFS: A, C, D, B, E, F, G, H

1. *DFS(v):*
   - *Mark v*
2. *For every unmarked neighbor n:*
   - *Call DFS(n)*

Vertex	marked[]
A	T
B	T
C	T
D	T
E	T
F	T
G	T
H	T