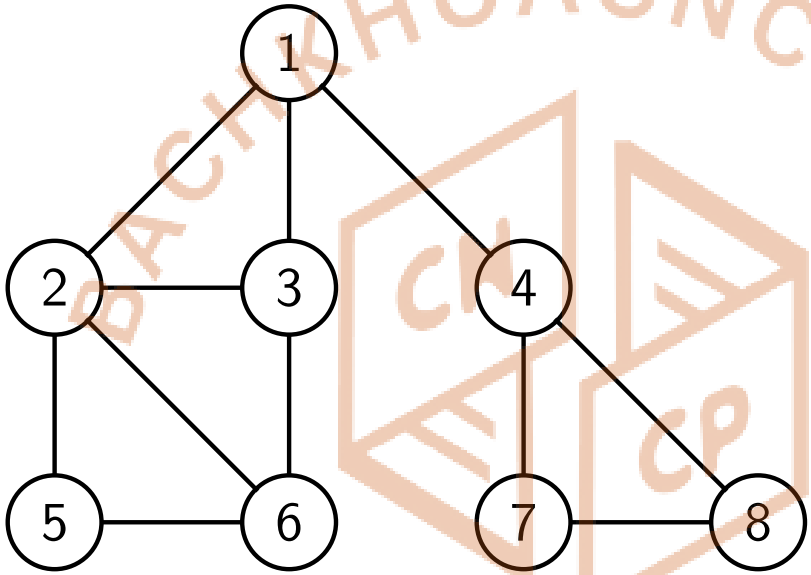
Chapter 11
Trees - DFS

Discrete Structures for Computing

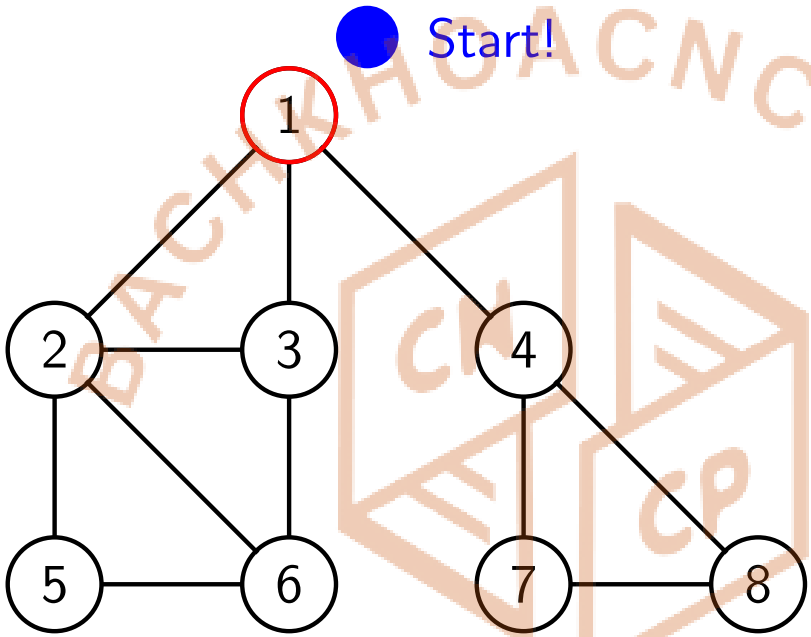
Huynh Tuong Nguyen, Nguyen Ngoc Le
Faculty of Computer Science and Engineering
University of Technology - VNUHCM
htnguyen@hcmut.edu.vn;ngngle@gmail.com

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



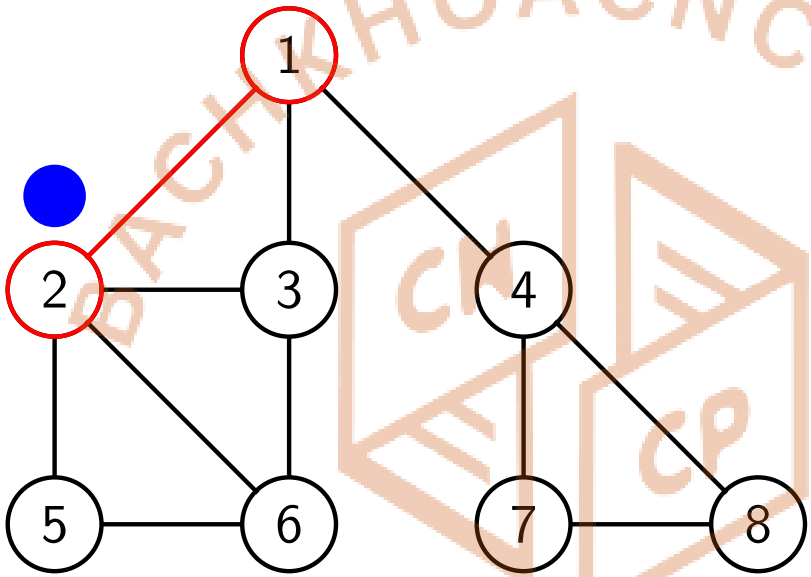
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



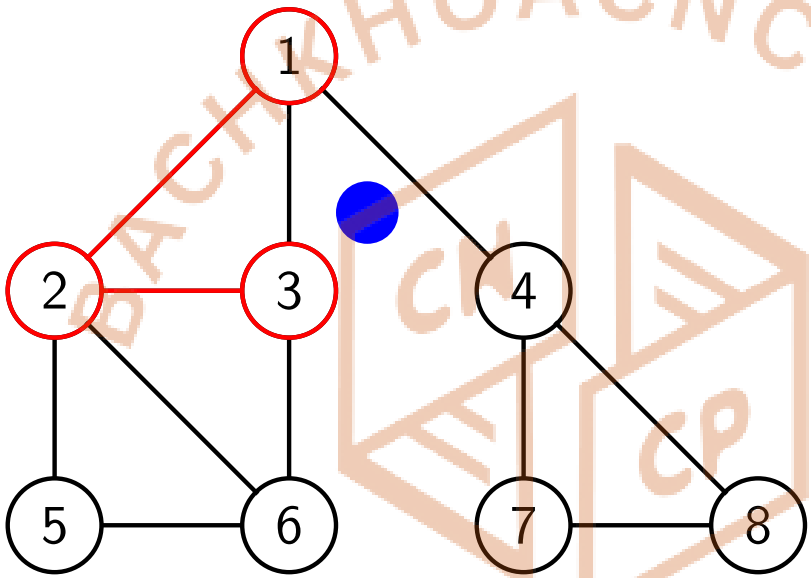
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



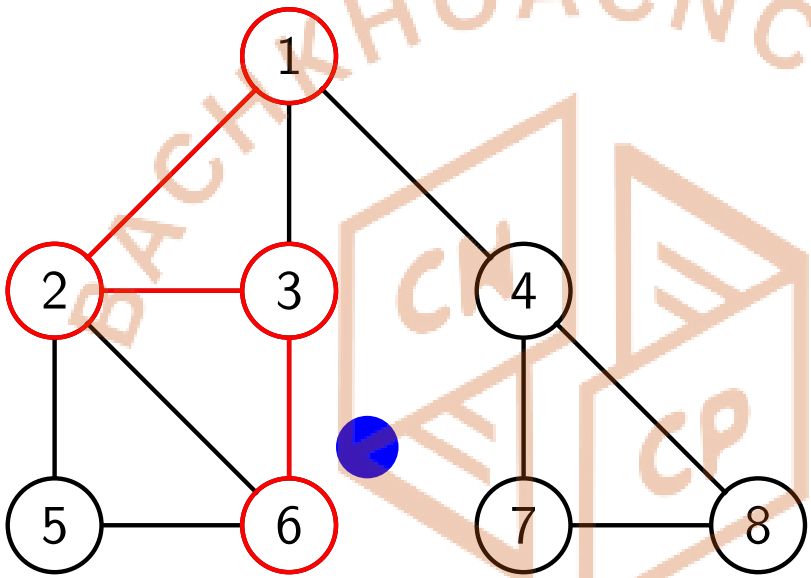
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



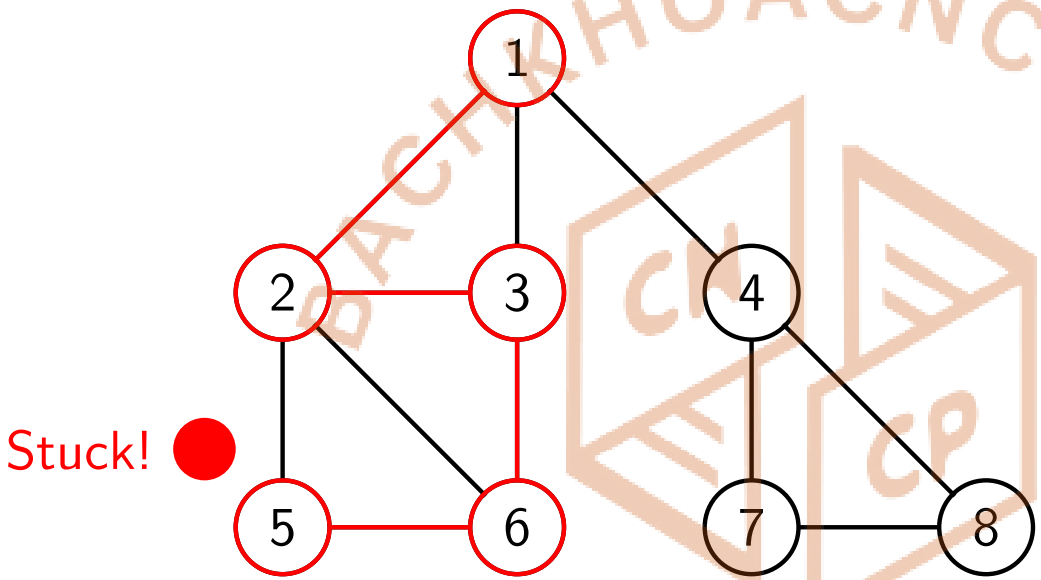
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



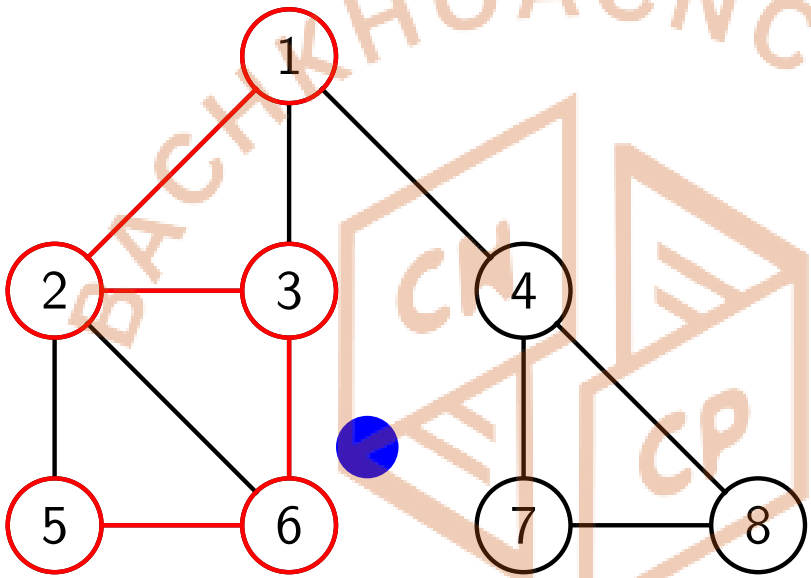
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



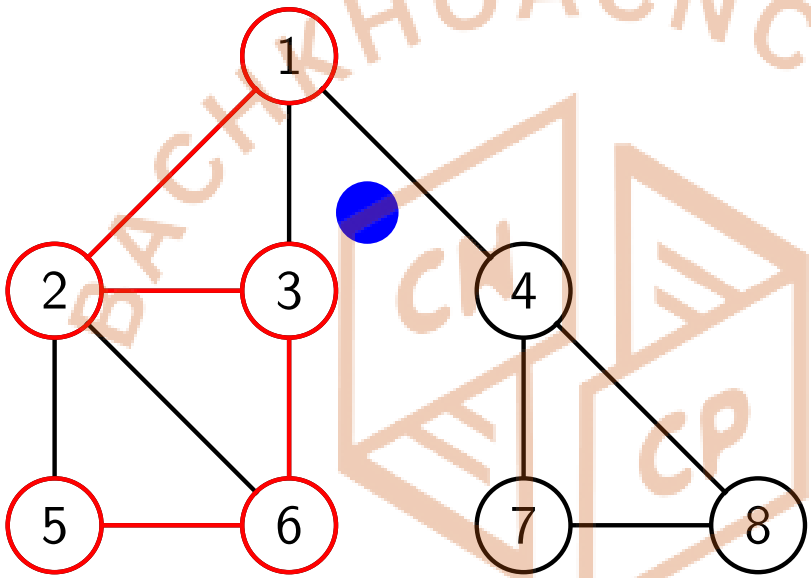
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



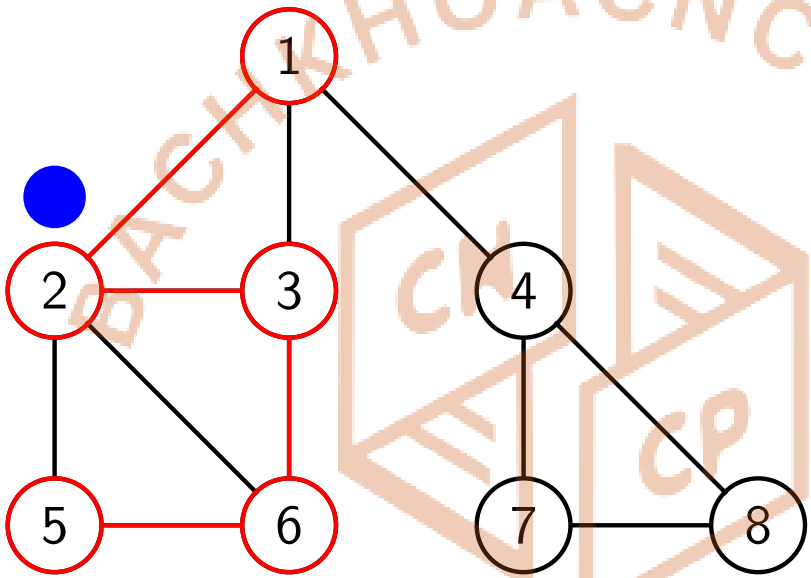
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



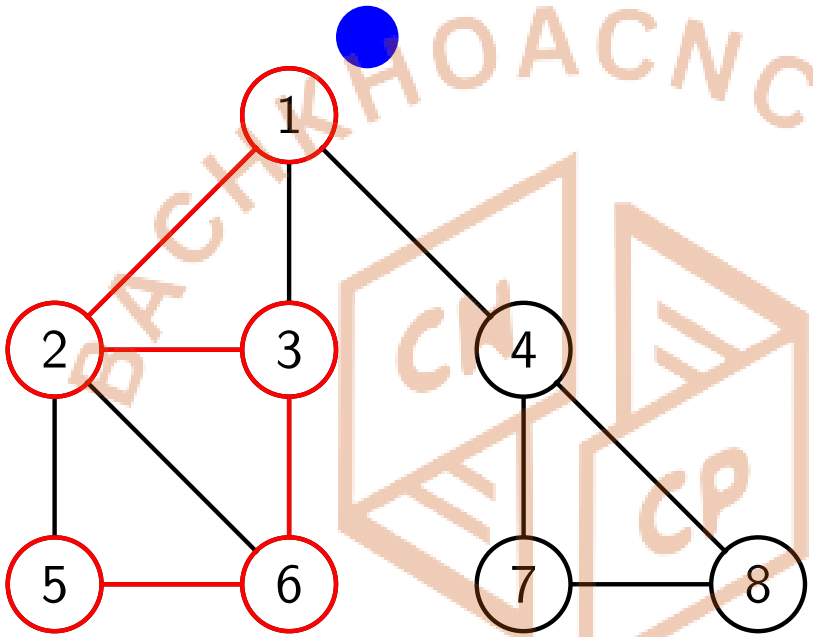
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



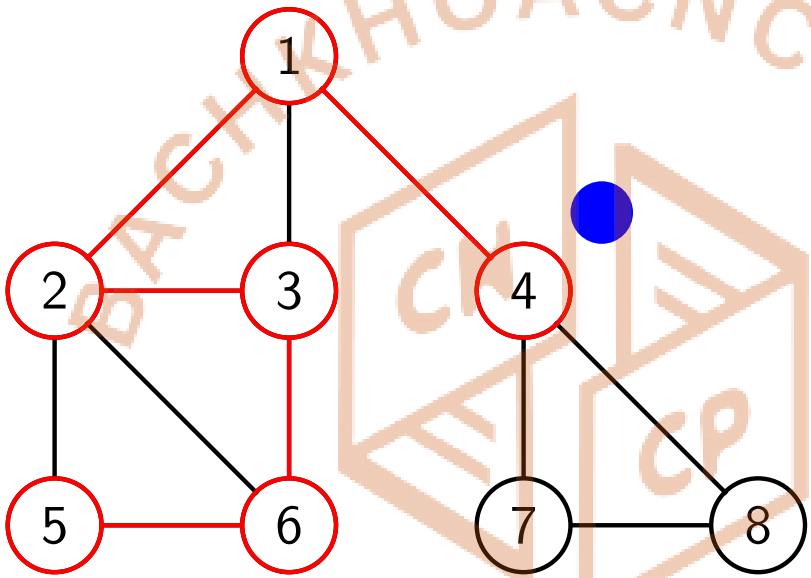
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



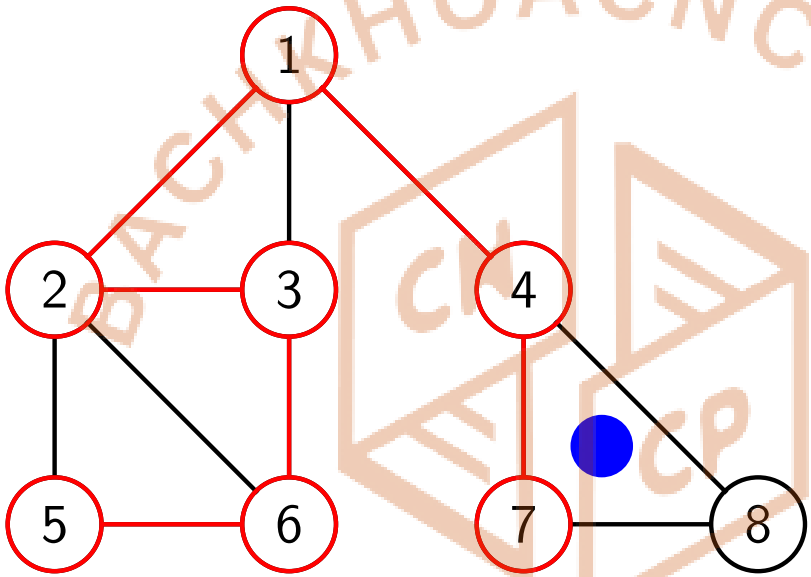
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



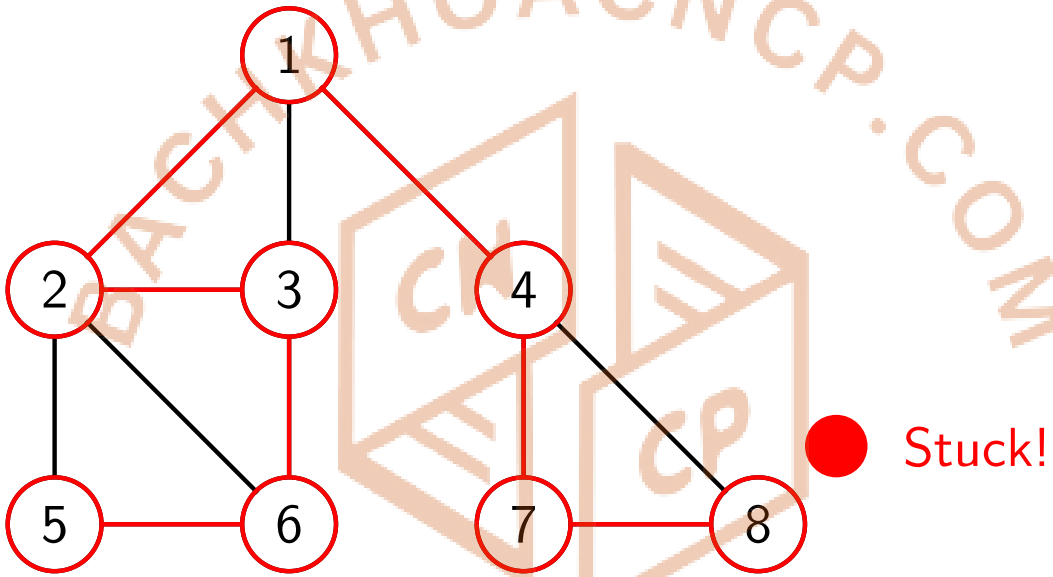
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



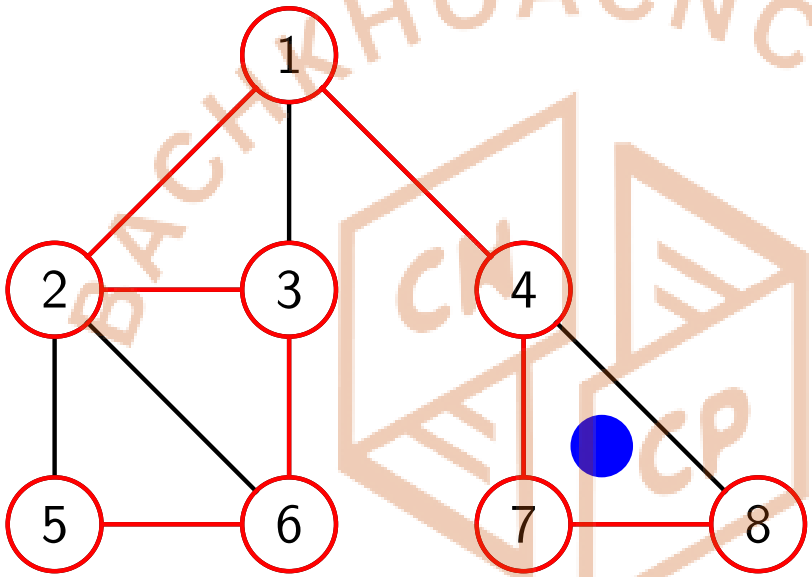
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



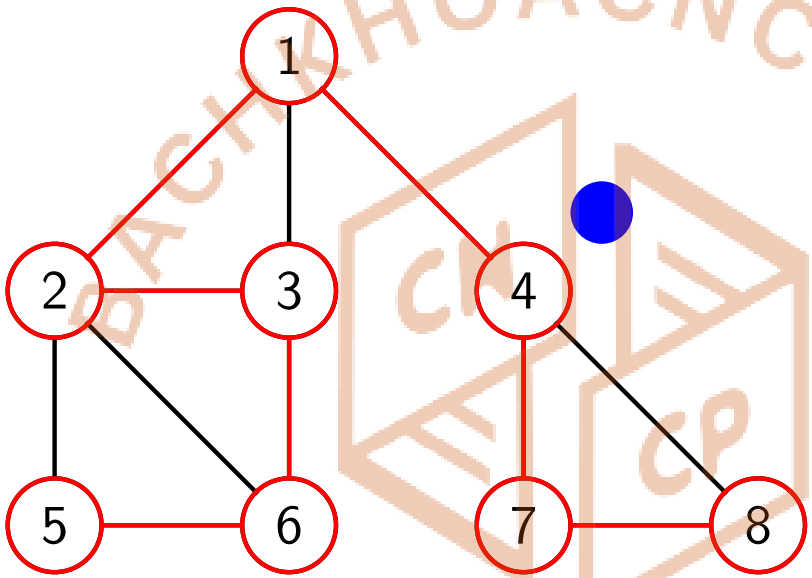
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



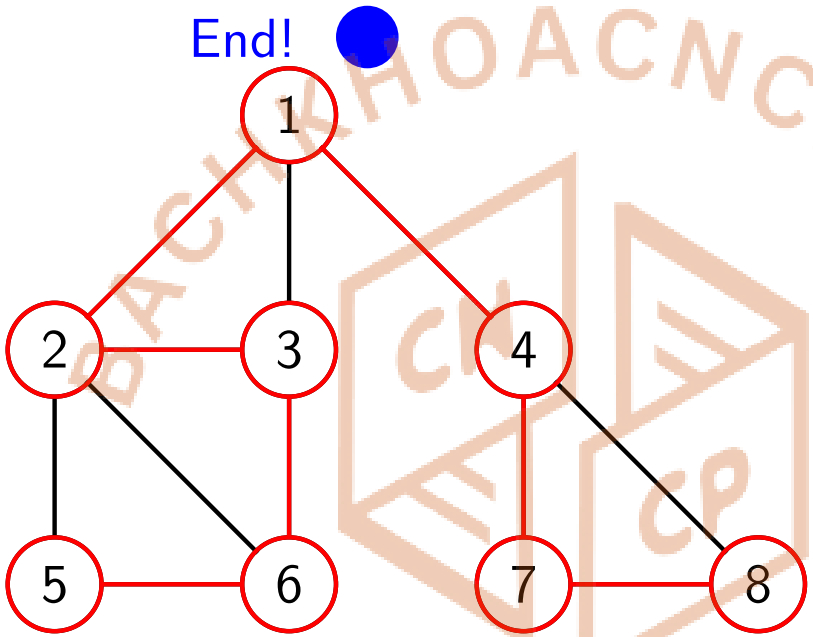
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



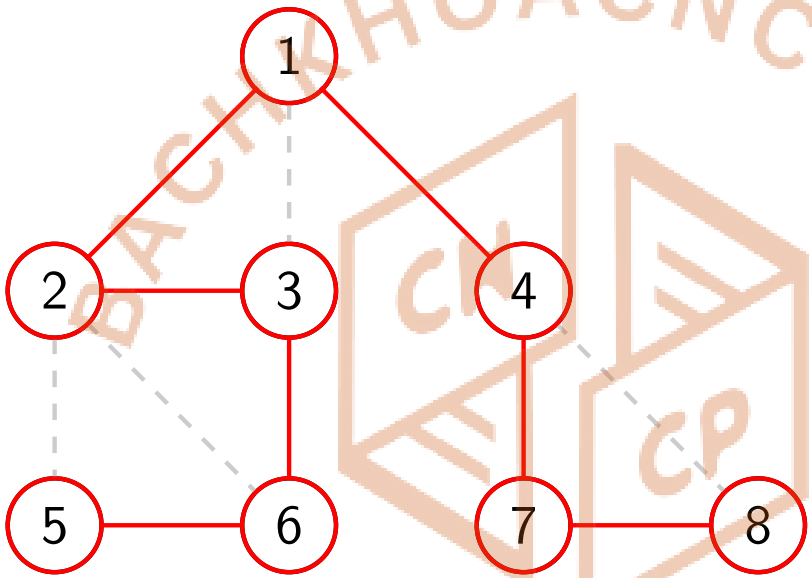
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



Huynh Tuong Nguyen,
Nguyen Ngoc Le

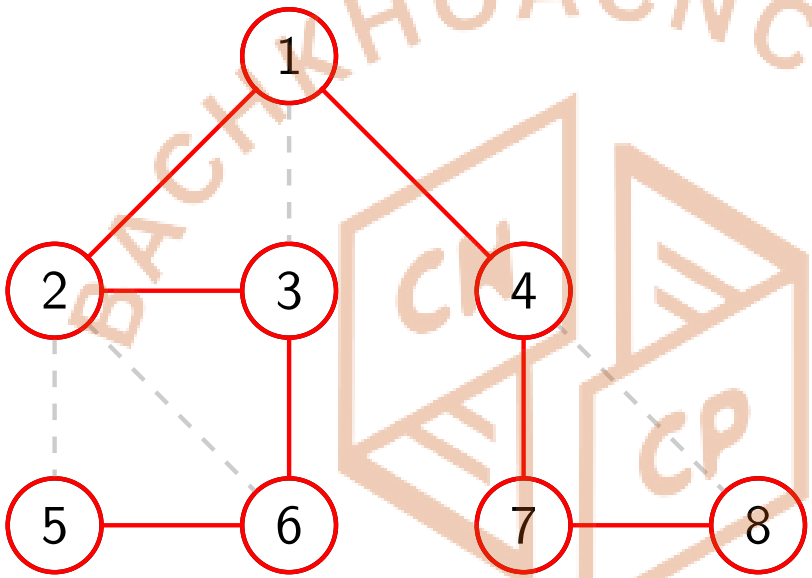
Depth-First Search (Tìm kiếm ưu tiên chiều sâu)



Huynh Tuong Nguyen,
Nguyen Ngoc Le

Depth-First Search (Tìm kiếm ưu tiên chiều sâu)

Huynh Tuong Nguyen,
Nguyen Ngoc Le



Property

- Go deeper as you can
- Backtrack (*quay lui*) to possible branch when you are stuck.
- $O(e)$ or $O(n^2)$

Depth-First Search

Algorithm

procedure *DFS* (G)
    $T :=$ tree consisting only vertex v_1
    visit(v_1)

procedure *visit*(v: vertex of G) /* recursive */
    for each vertex w adjacent to v and not in T
        add w and edge $\{v, w\}$ to T
        visit(w)

Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

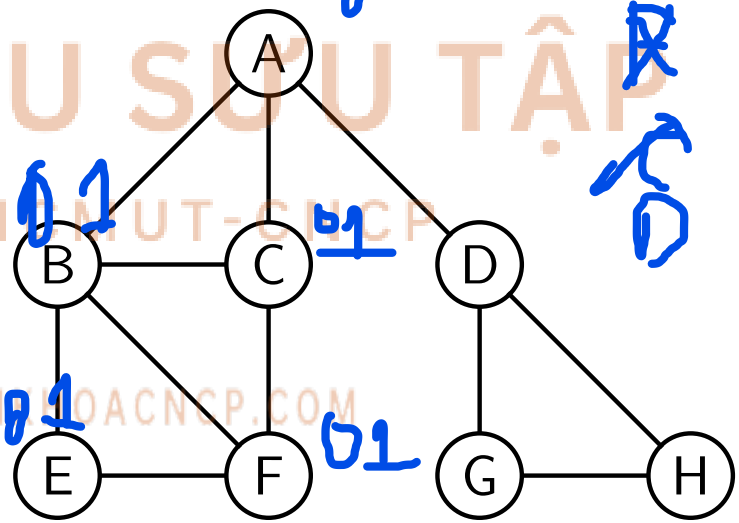
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	W	•	•
B	-	W	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 0



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

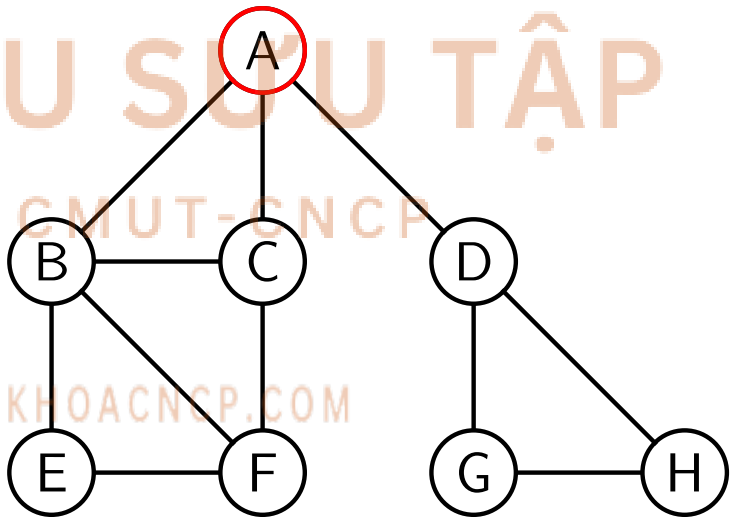
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	W	•	•
B	-	W	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 0



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

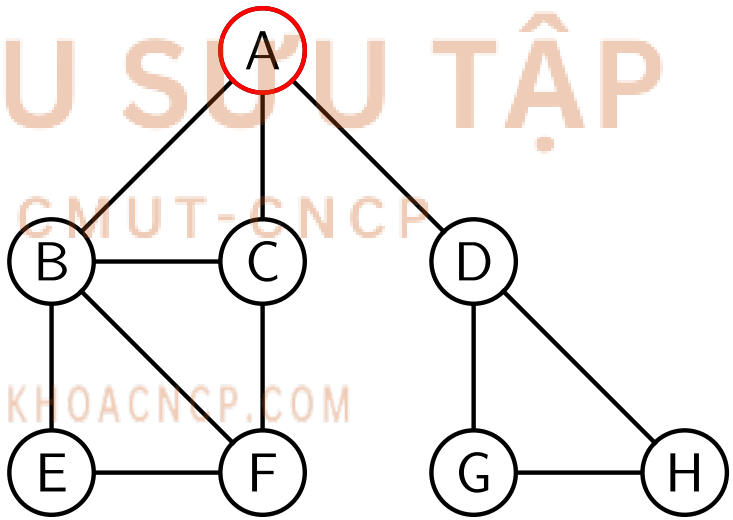
```
1.loop (more vertex v in G)
   1.color[v]= White
   2.father[v]=null
2.time=0
3.loop (more vertex v in G )
   1.if (color[v]== White)
      1.DFSVisit(G, v)
```

void DFSVisit (G, v)

```
1.color[v]=Gray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
   1.if(color[u]==White)
      1. father[u]=v
      2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time
```

vertex	father	color	d	f
A	-	WG	•	•
B	-	W	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 0



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

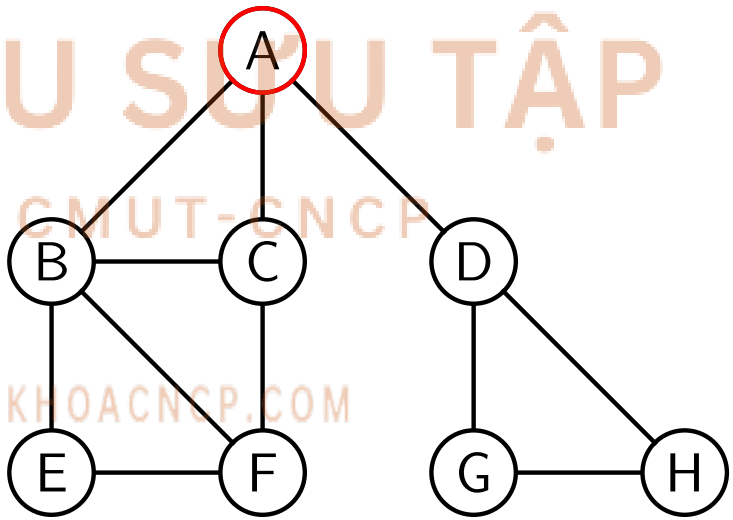
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	•	•
B	-	W	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 1



A pseudocode of DFS

void DFS(G)

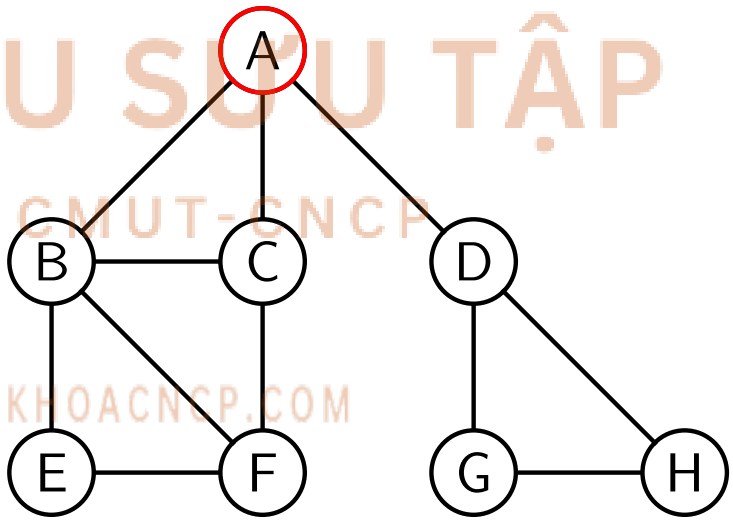
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	-	W	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 1



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

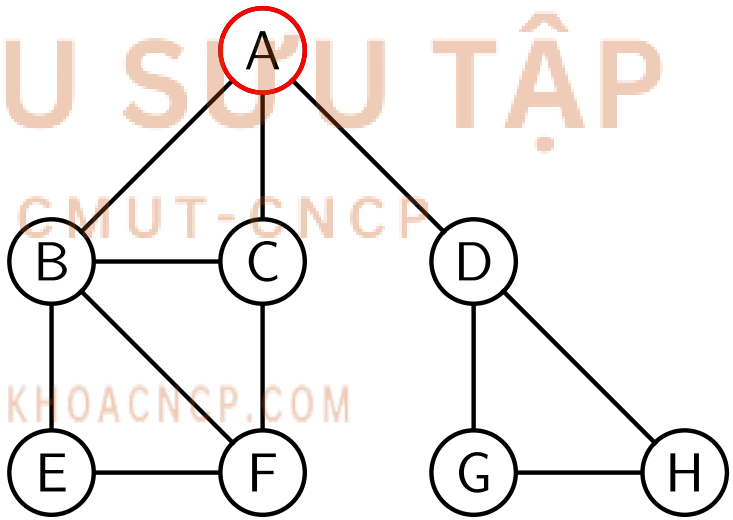
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	W	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 1



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

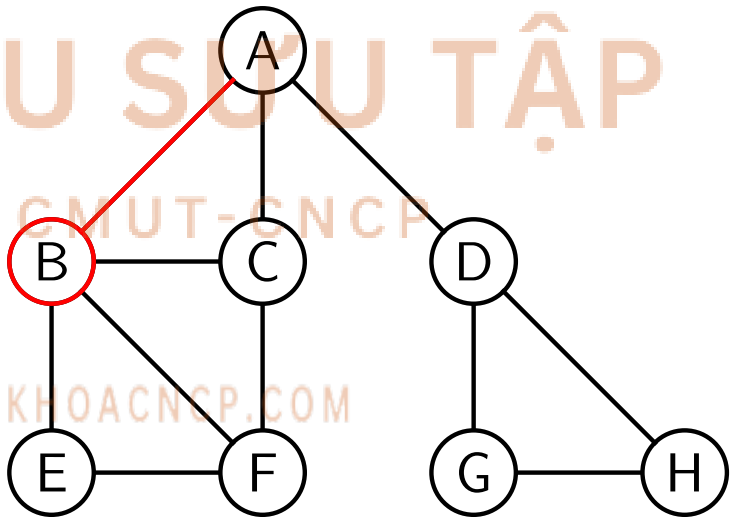
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	W	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 1



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

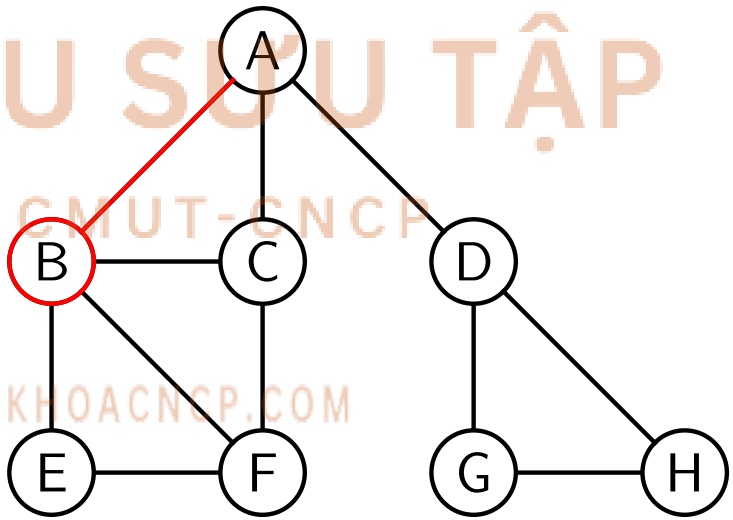
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**Gray**
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 1



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

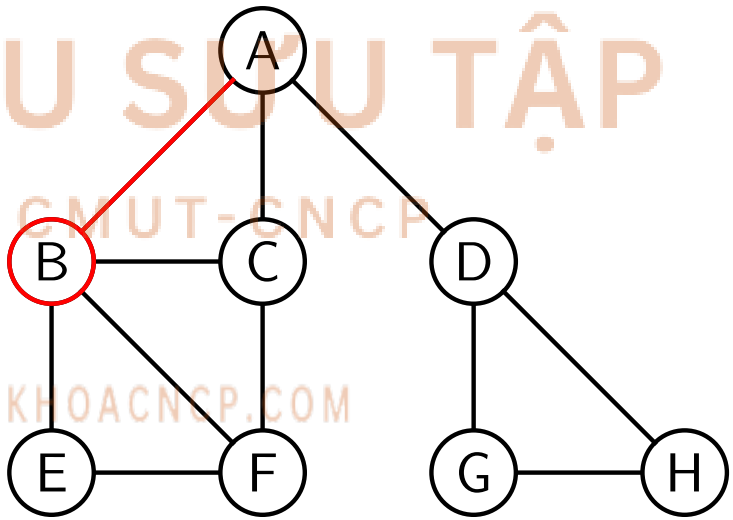
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	•	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 2



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

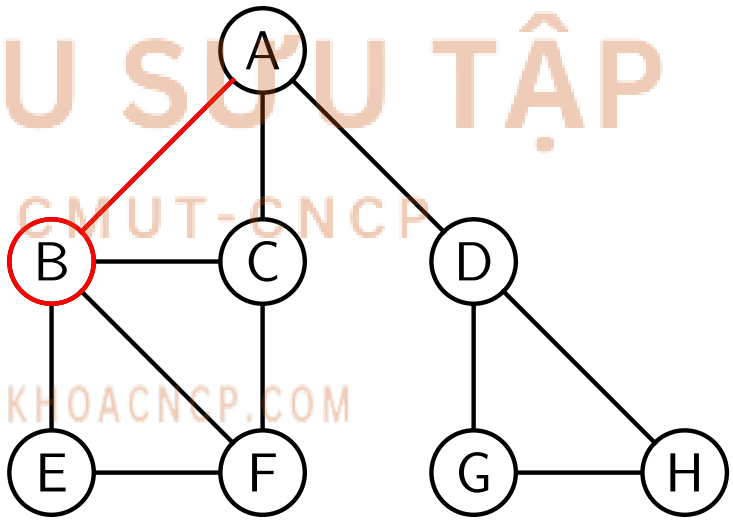
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	-	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 2



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

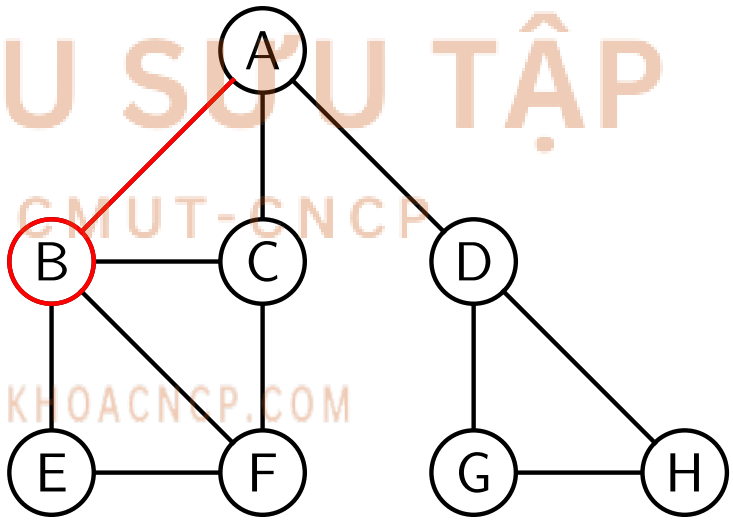
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 2



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

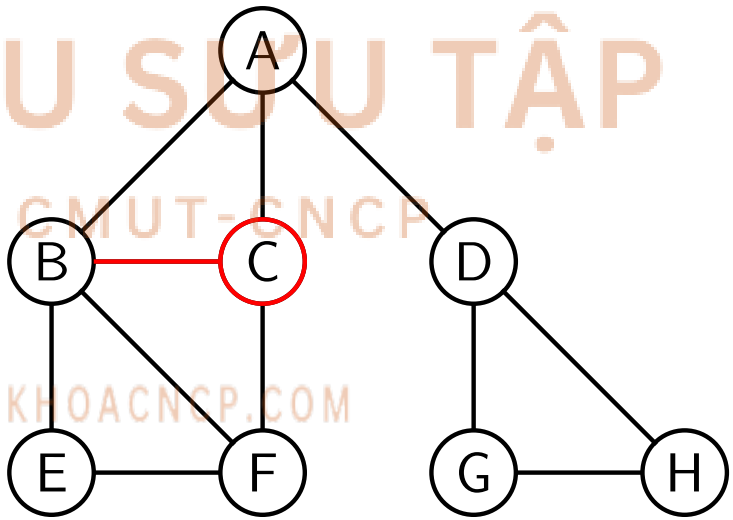
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	W	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 2



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

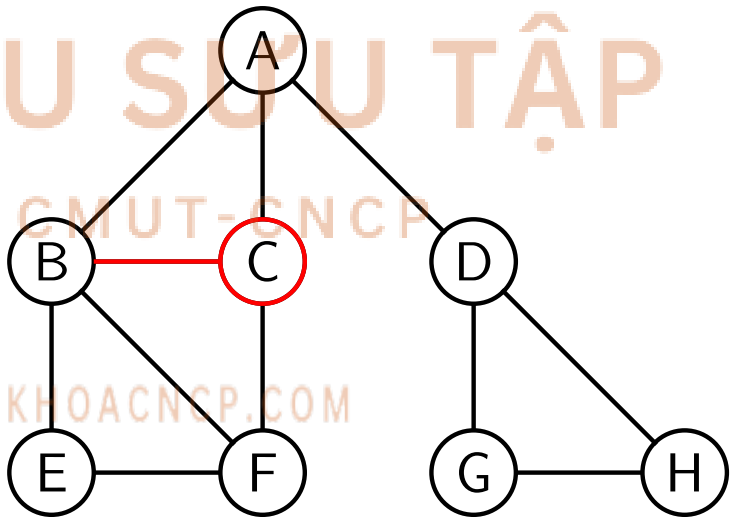
```
1.loop (more vertex v in G)
   1.color[v]= White
   2.father[v]=null
2.time=0
3.loop (more vertex v in G )
   1.if (color[v]== White)
      1.DFSVisit(G, v)
```

void DFSVisit (G, v)

```
1.color[v]=Gray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
   1.if(color[u]==White)
      1. father[u]=v
      2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time
```

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 2



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

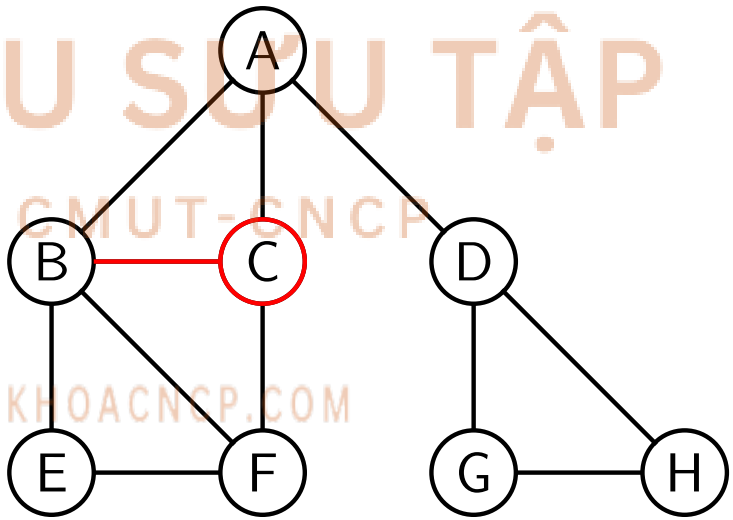
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	•	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 3



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

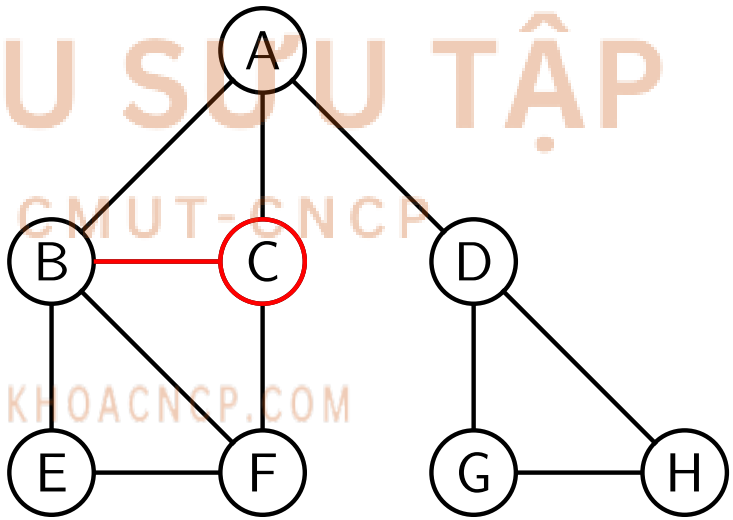
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	-	W	•	•
F	-	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 3



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

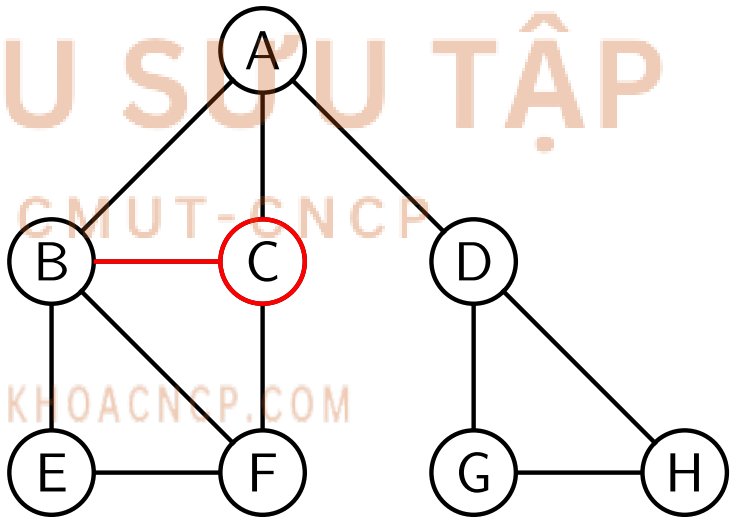
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	-	W	•	•
F	C	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 3



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

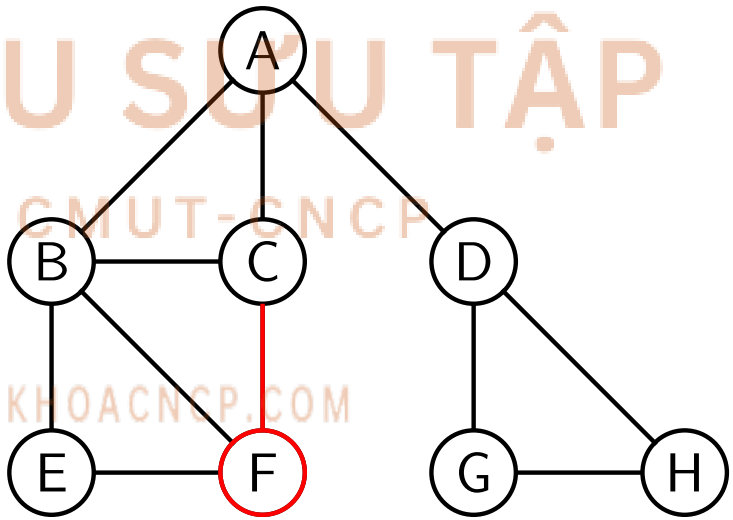
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	-	W	•	•
F	C	W	•	•
G	-	W	•	•
H	-	W	•	•

time = 3



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

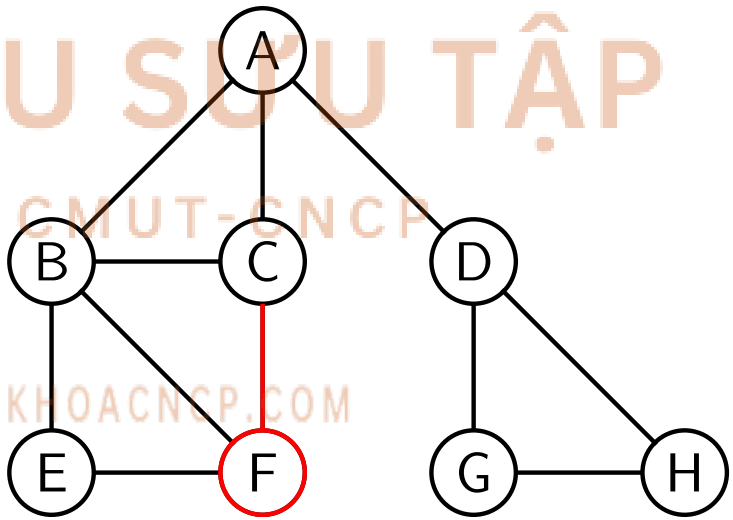
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**Gray**
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	-	W	•	•
F	C	WG	•	•
G	-	W	•	•
H	-	W	•	•

time = 3



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

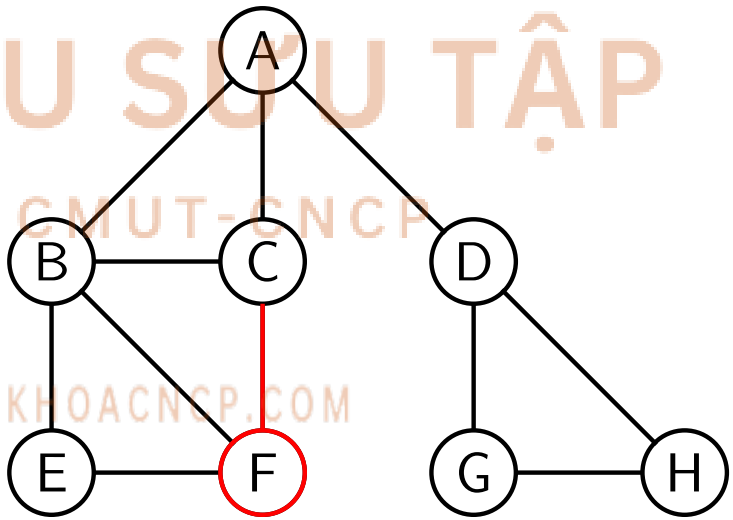
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	-	W	•	•
F	C	WG	•	•
G	-	W	•	•
H	-	W	•	•

time = 4



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

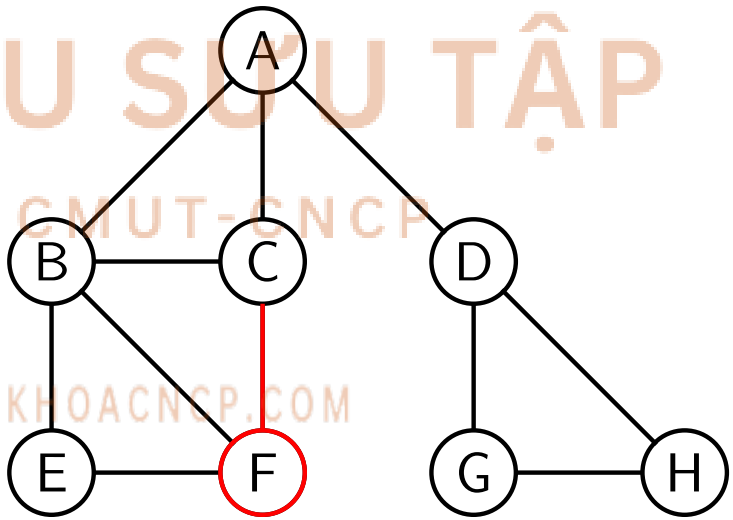
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	-	W	•	•
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 4



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

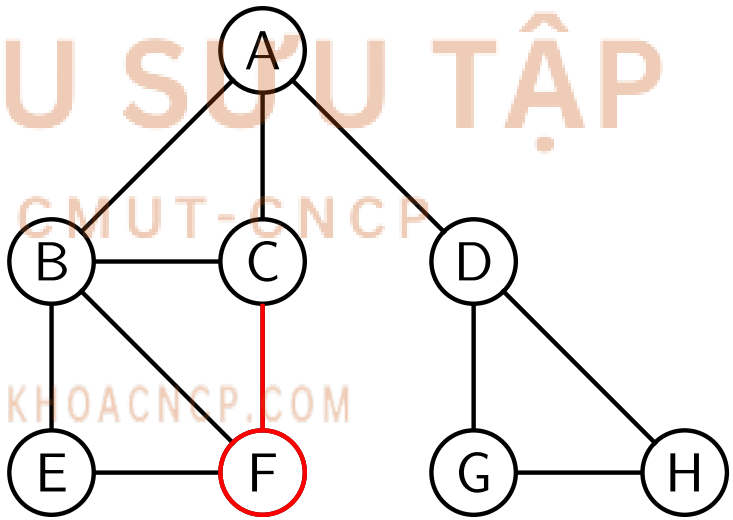
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	W	•	•
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 4



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

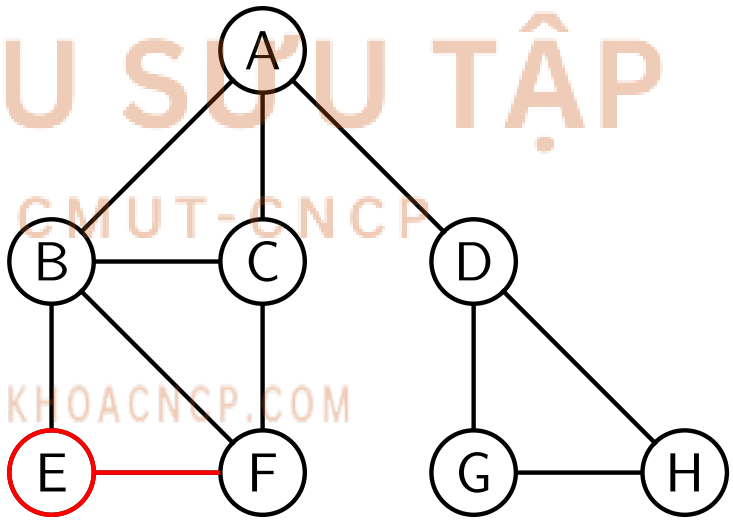
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	W	•	•
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 4



A pseudocode of DFS

void DFS(G)

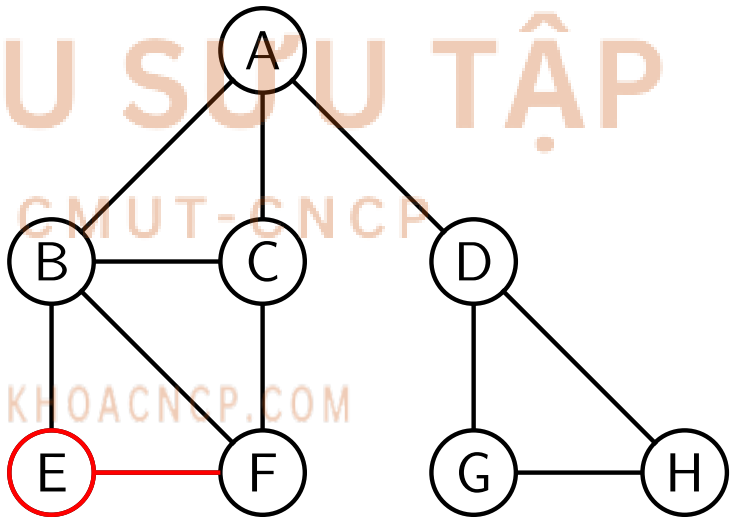
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**Gray**
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WG	•	•
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 4



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

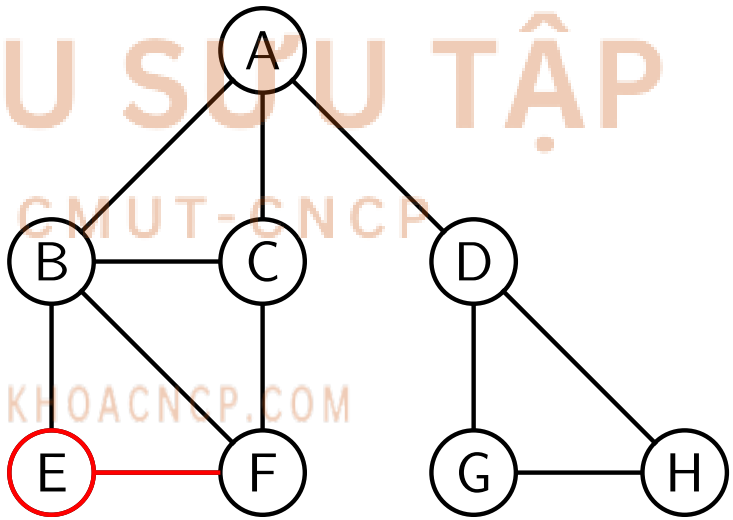
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WG	•	•
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 5



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

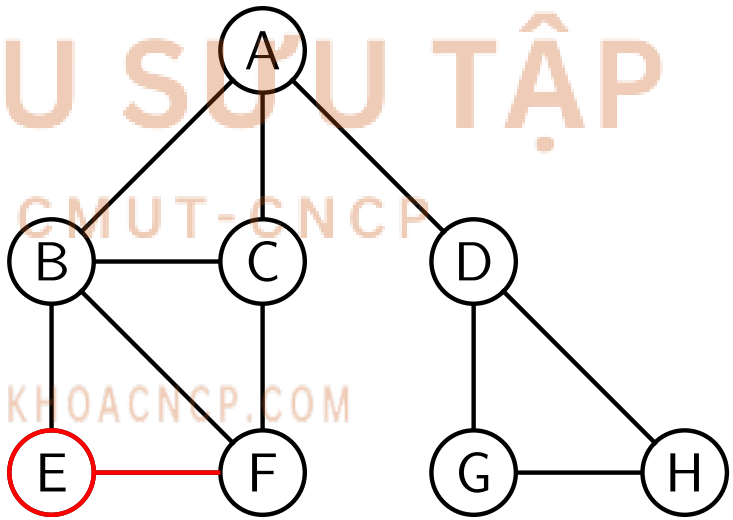
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WG	5	•
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 5



A pseudocode of DFS

void DFS(G)

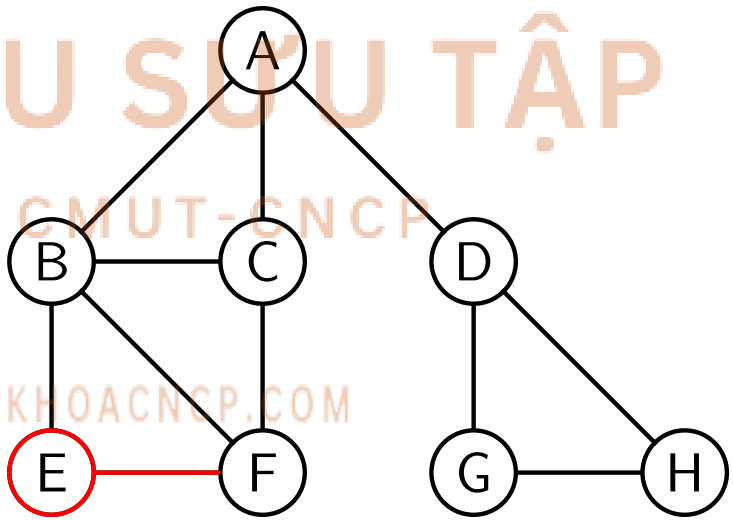
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WGB	5	•
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 5



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

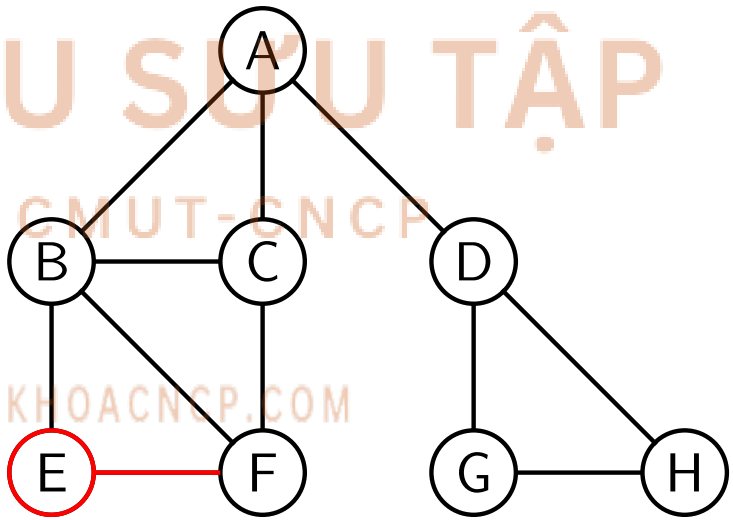
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WGB	5	•
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 6



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

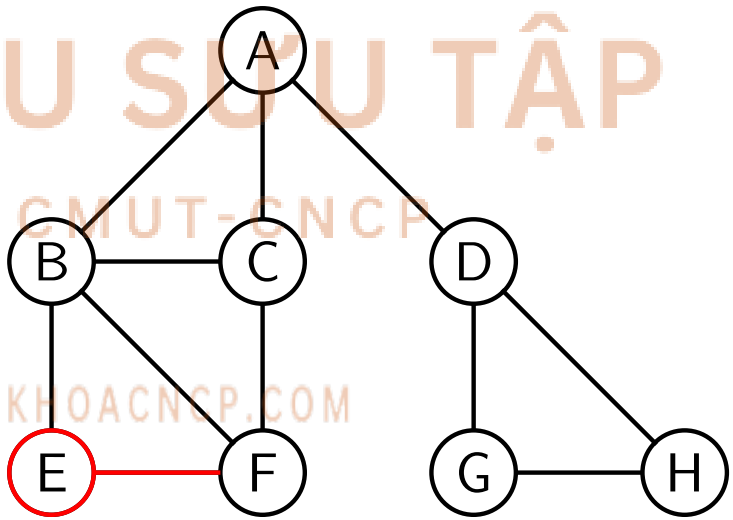
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WGB	5	6
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 6



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

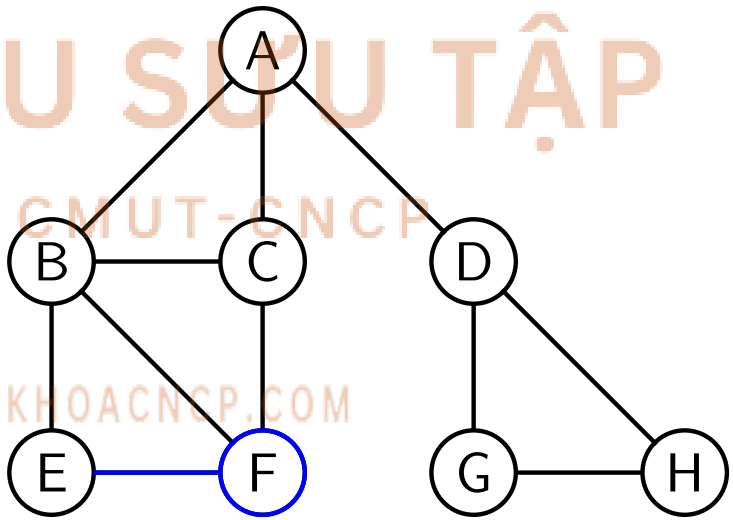
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WGB	5	6
F	C	WG	4	•
G	-	W	•	•
H	-	W	•	•

time = 6



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

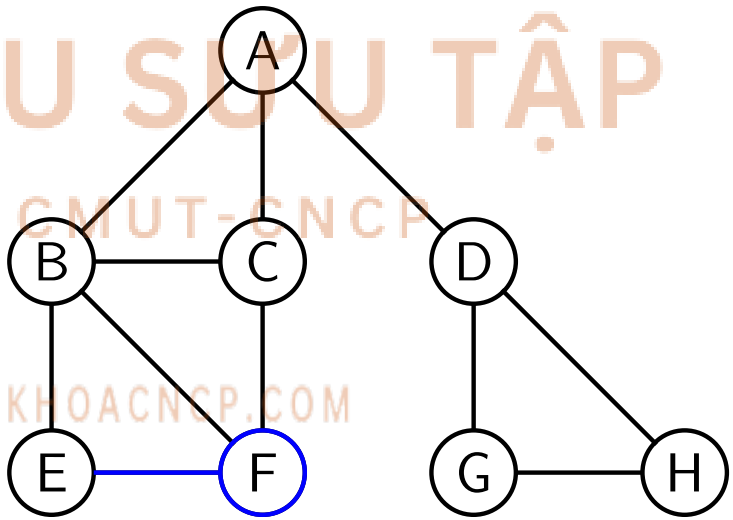
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	•
G	-	W	•	•
H	-	W	•	•

time = 6



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

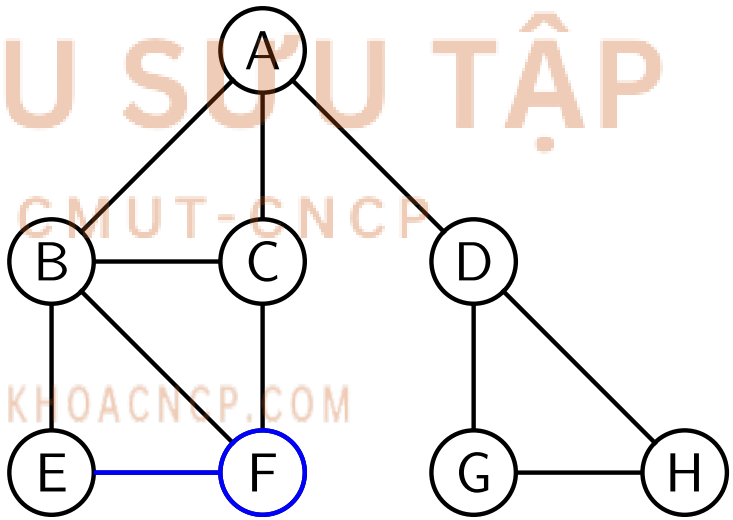
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	•
G	-	W	•	•
H	-	W	•	•

time = 7



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

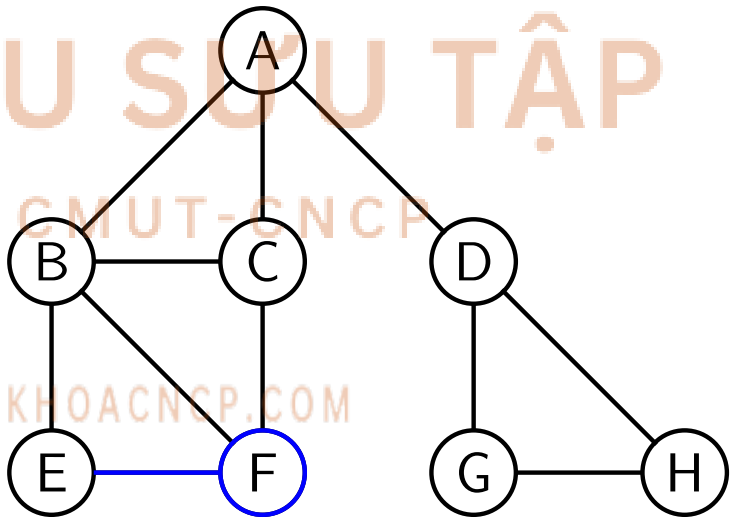
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 7



A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

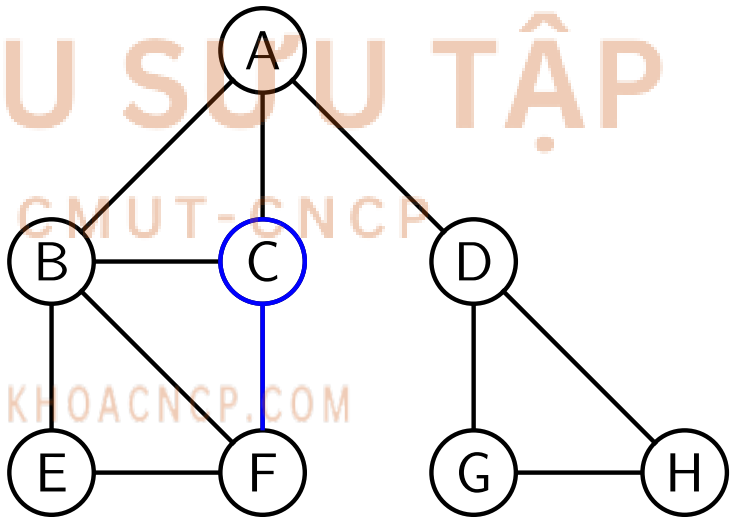
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WG	3	•
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 7



A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

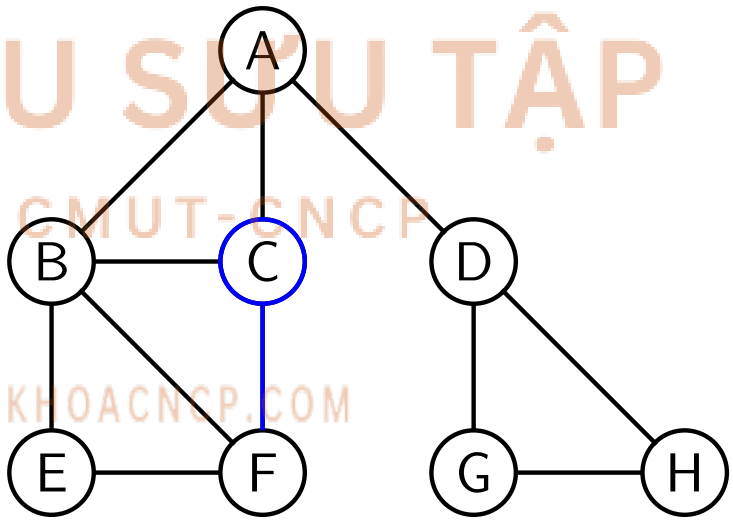
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WGB	3	•
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 7



A pseudocode of DFS

void DFS(G)

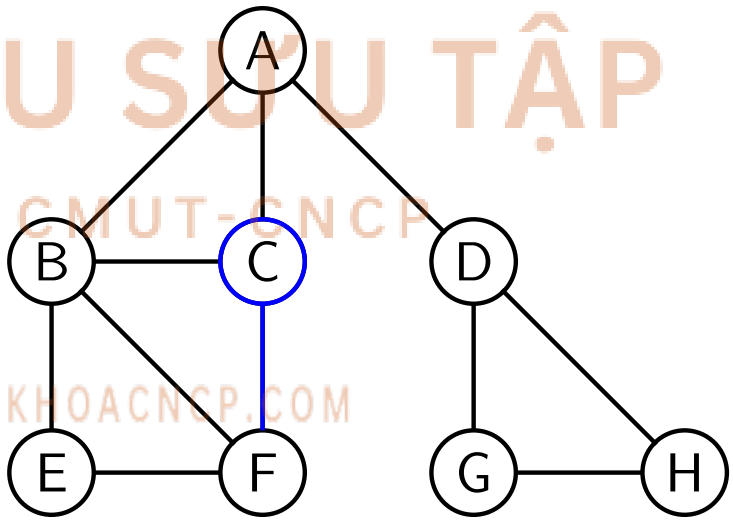
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WGB	3	•
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 8



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

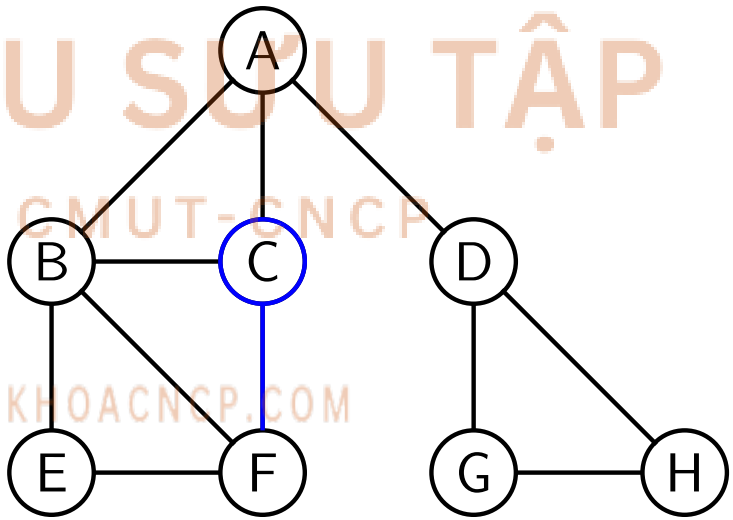
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WGB	3	8
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 8



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

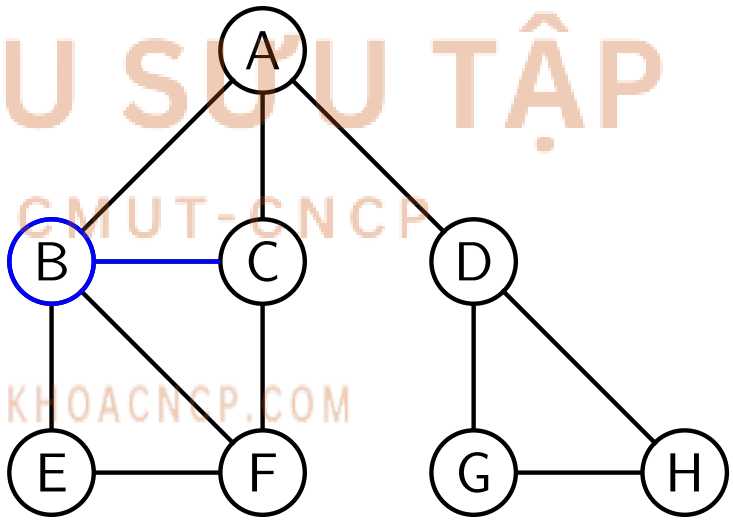
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WG	2	•
C	B	WGB	3	8
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 8



A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

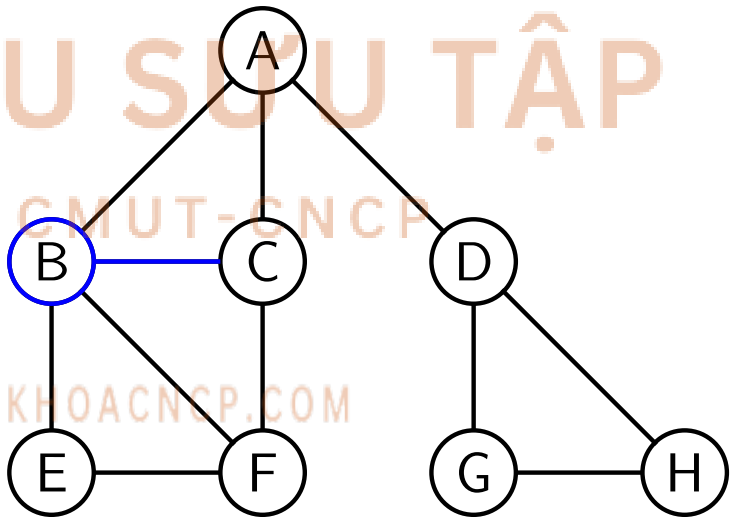
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	•
C	B	WGB	3	8
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 8



A pseudocode of DFS

void DFS(G)

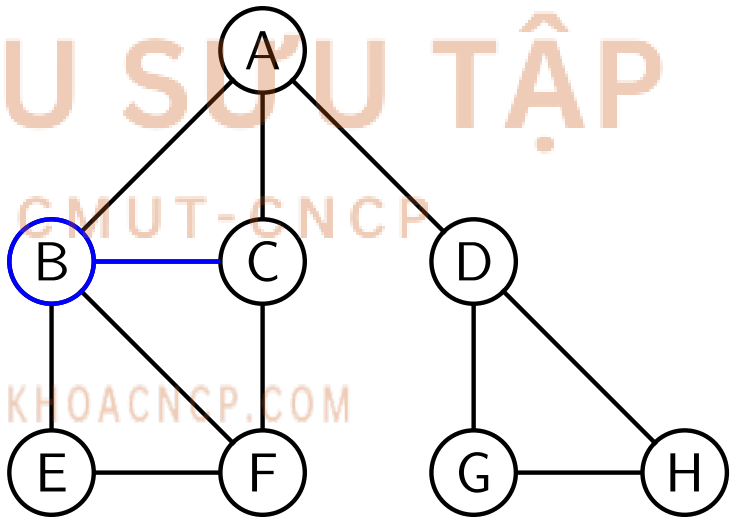
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	•
C	B	WGB	3	8
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 9



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

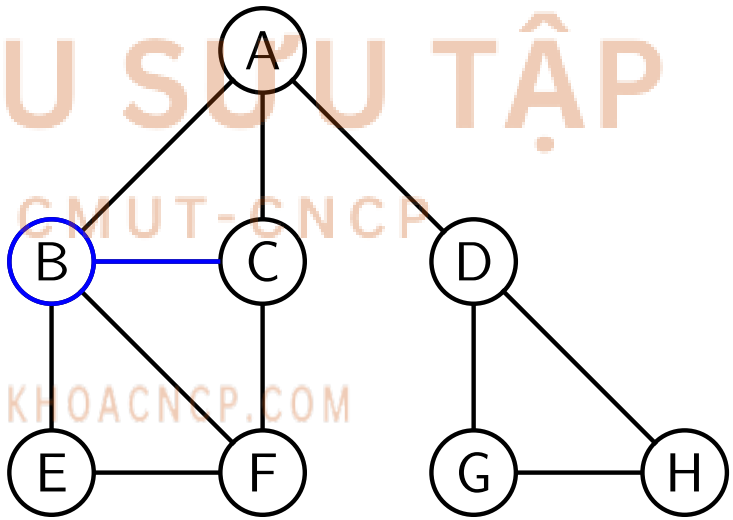
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 9



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

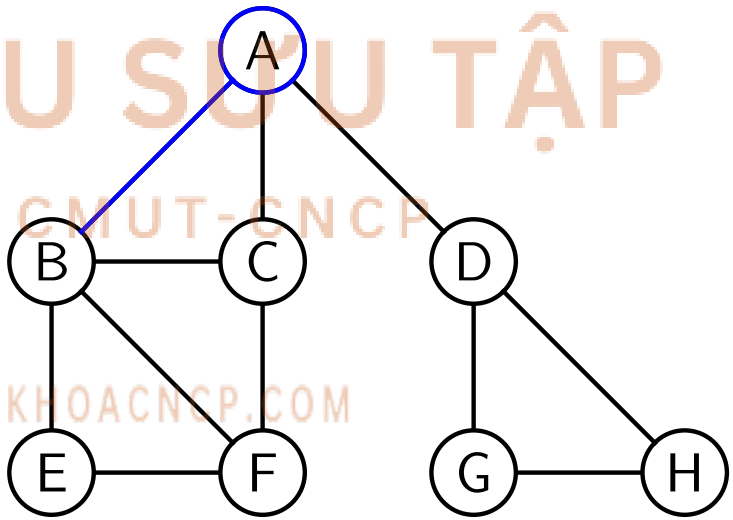
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	-	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 9



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

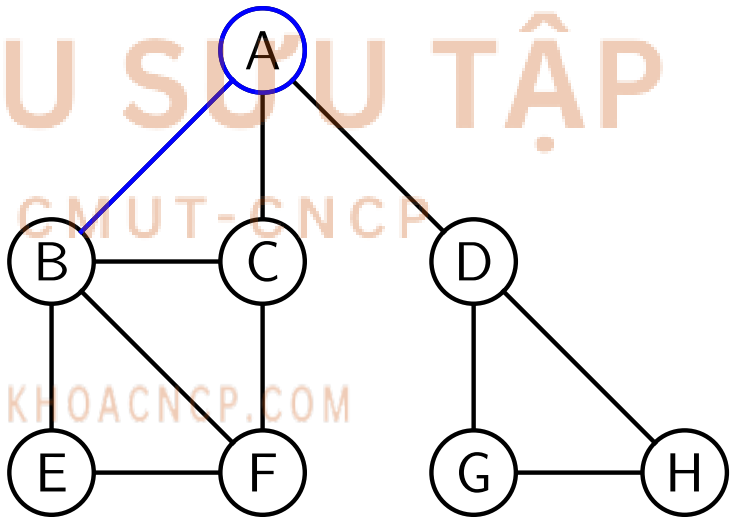
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 9



A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

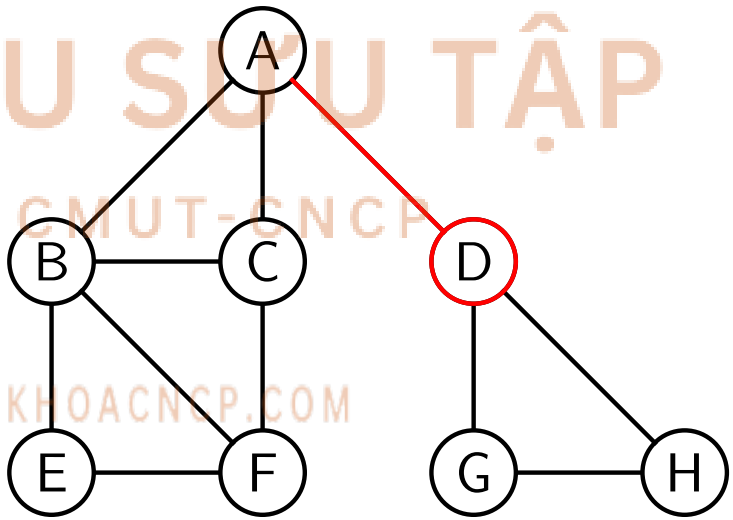
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	W	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 9



A pseudocode of DFS

void DFS(G)

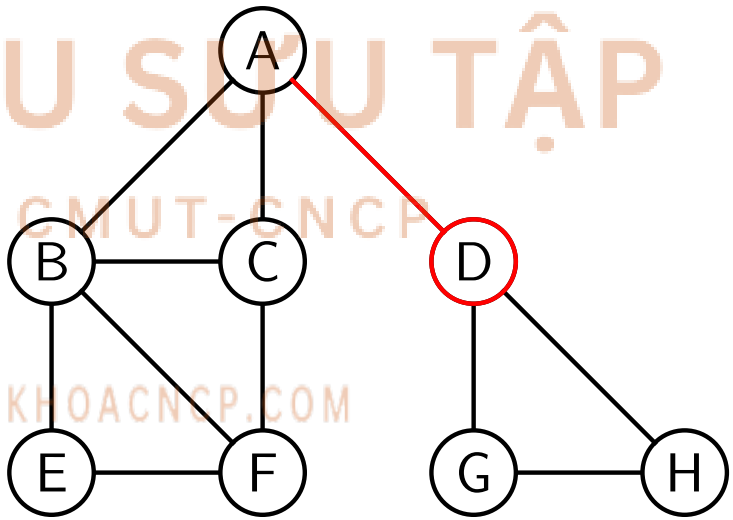
```
1.loop (more vertex v in G)
   1.color[v]= White
   2.father[v]=null
2.time=0
3.loop (more vertex v in G )
   1.if (color[v]== White)
      1.DFSVisit(G, v)
```

void DFSVisit (G, v)

```
1.color[v]=Gray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
   1.if(color[u]==White)
      1. father[u]=v
      2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time
```

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 9



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

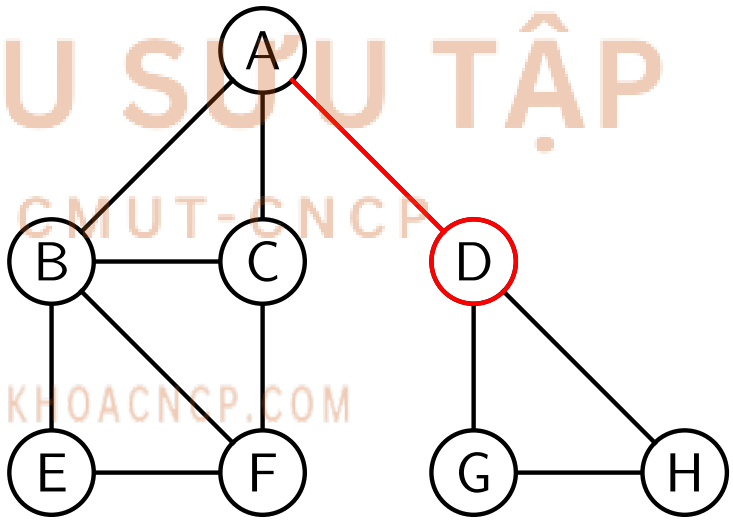
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	•	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 10



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

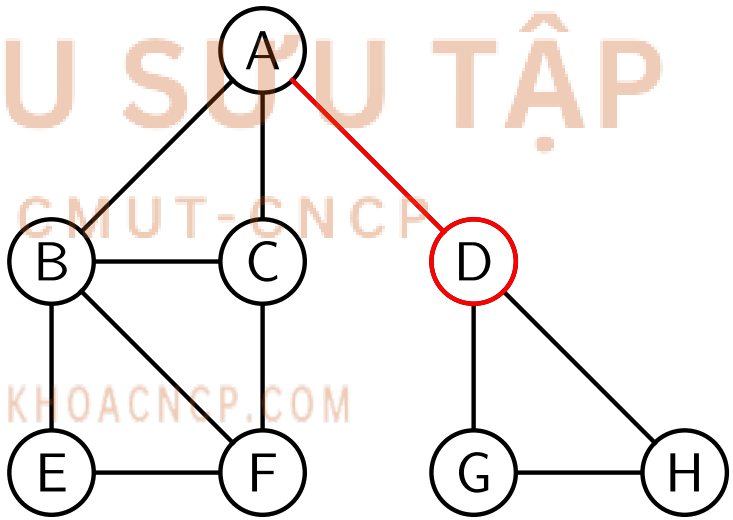
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	-	W	•	•
H	-	W	•	•

time = 10



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

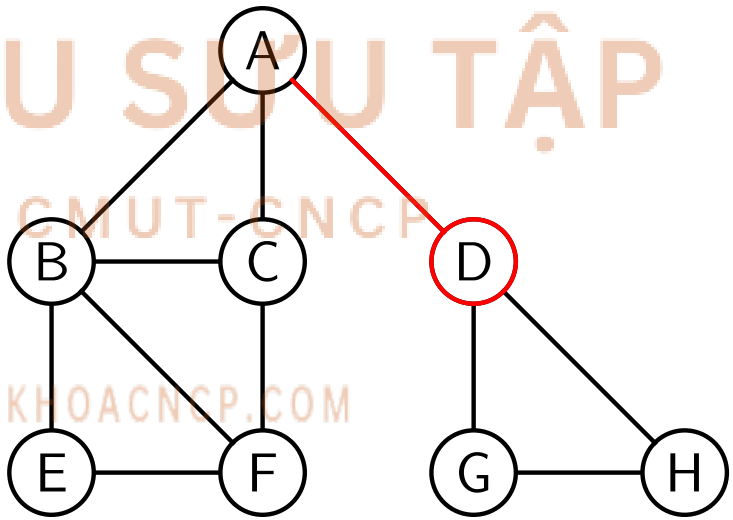
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	W	•	•
H	-	W	•	•

time = 10



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

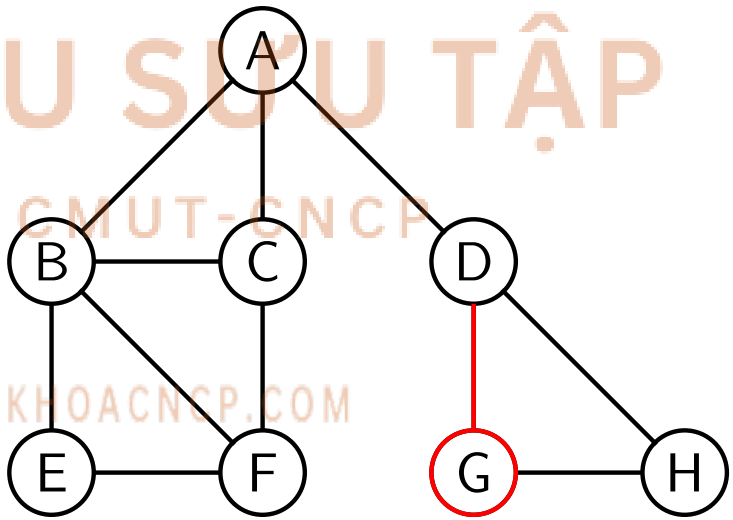
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	W	•	•
H	-	W	•	•

time = 10



A pseudocode of DFS

void DFS(G)

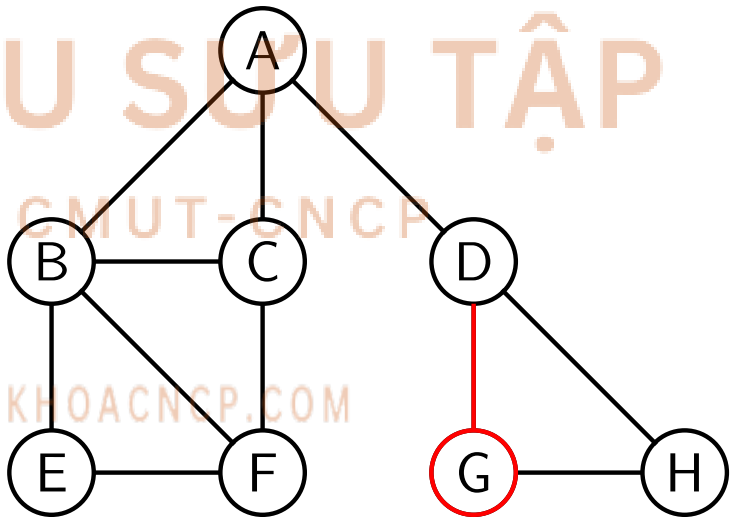
```
1.loop (more vertex v in G)
   1.color[v]= White
   2.father[v]=null
2.time=0
3.loop (more vertex v in G )
   1.if (color[v]== White)
      1.DFSVisit(G, v)
```

void DFSVisit (G, v)

```
1.color[v]=Gray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
   1.if(color[u]==White)
      1. father[u]=v
      2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time
```

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	•	•
H	-	W	•	•

time = 10



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

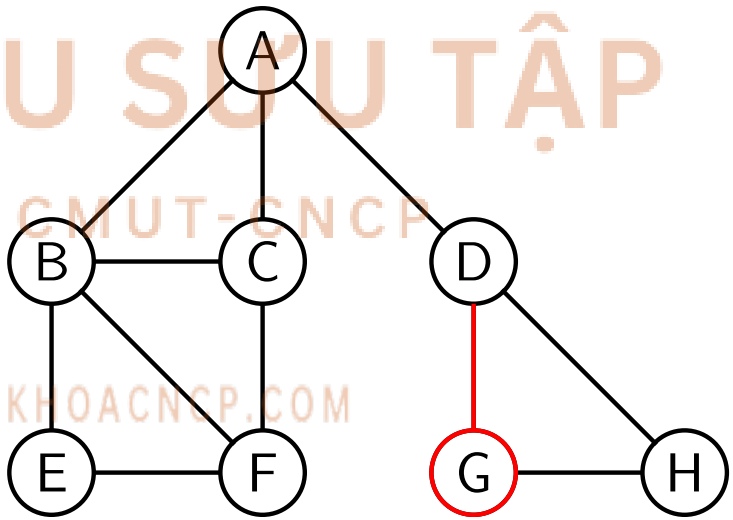
void DFS(G)

1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	•	•
H	-	W	•	•
time =		11		



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

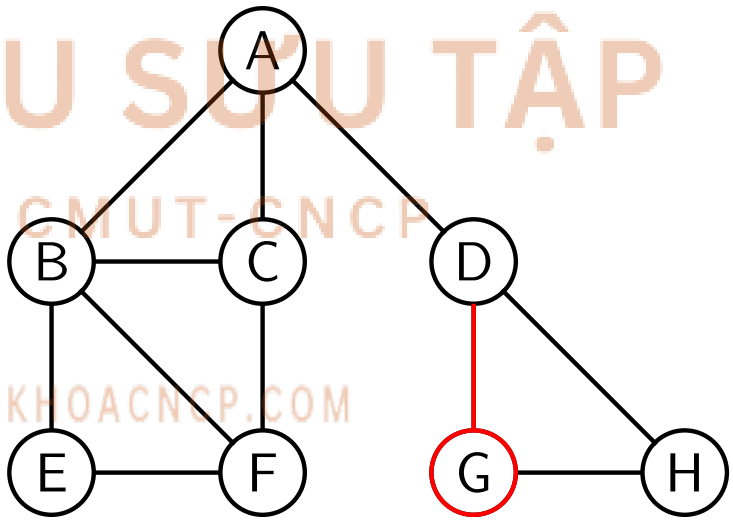
void DFS(G)

1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	-	W	•	•
time =		11		



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

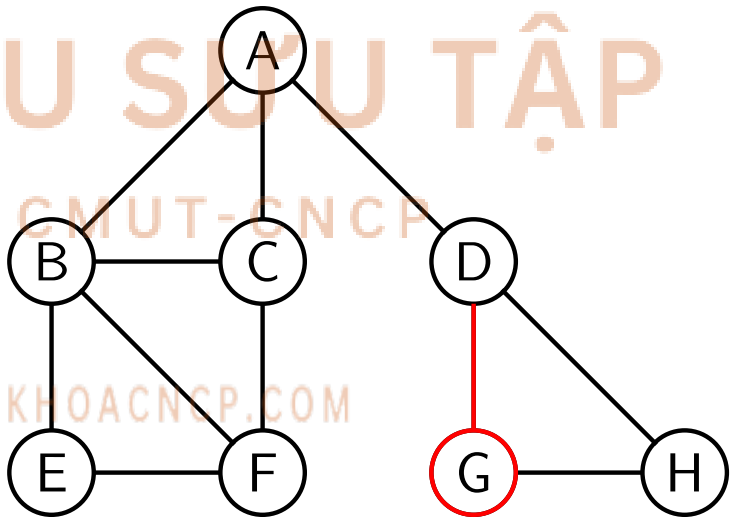
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	W	•	•

time = 11



A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

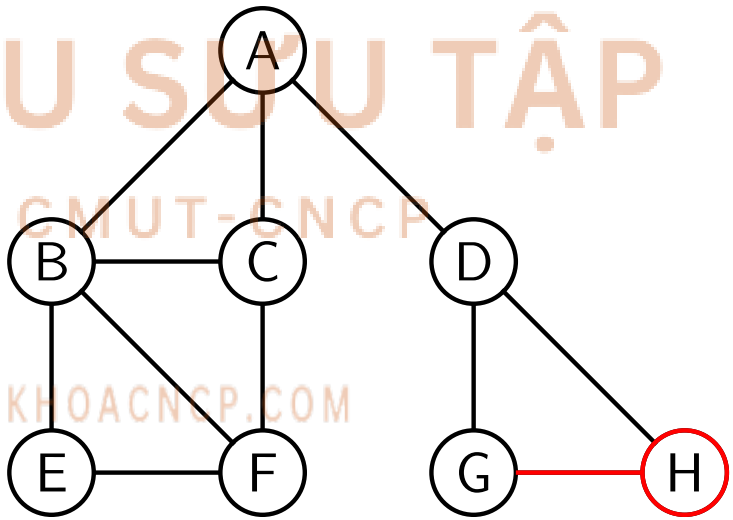
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	W	•	•

time = 11



A pseudocode of DFS

void DFS(G)

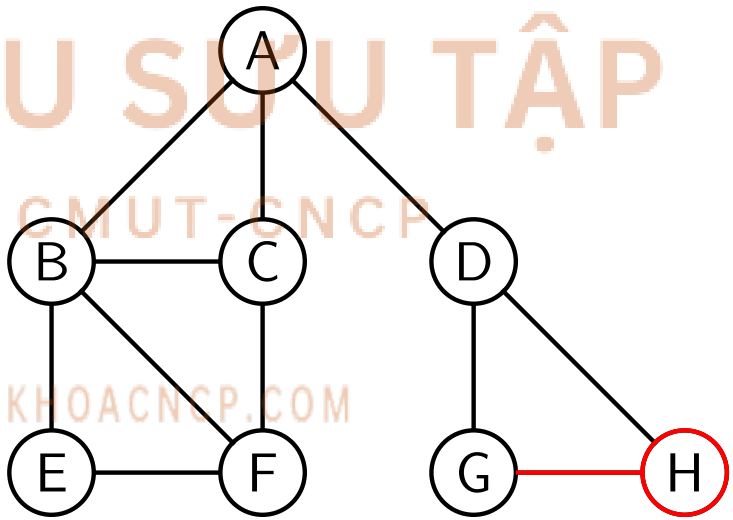
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**Gray**
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	WG	•	•

time = 11



A pseudocode of DFS

void DFS(G)

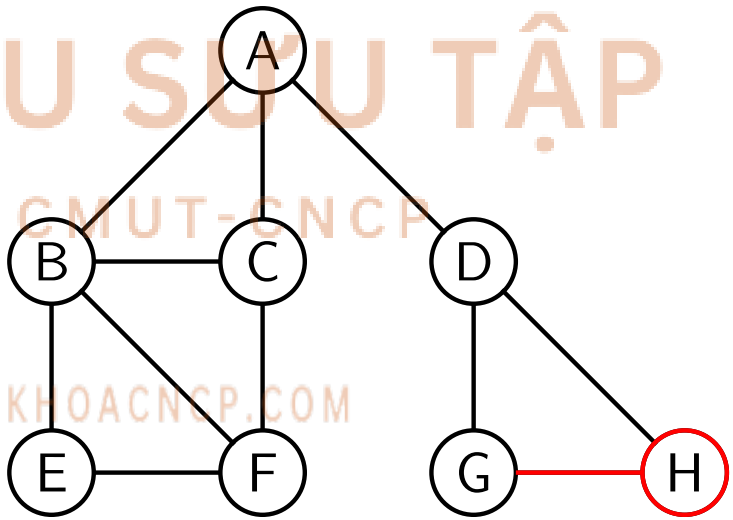
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	WG	•	•

time = 12



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

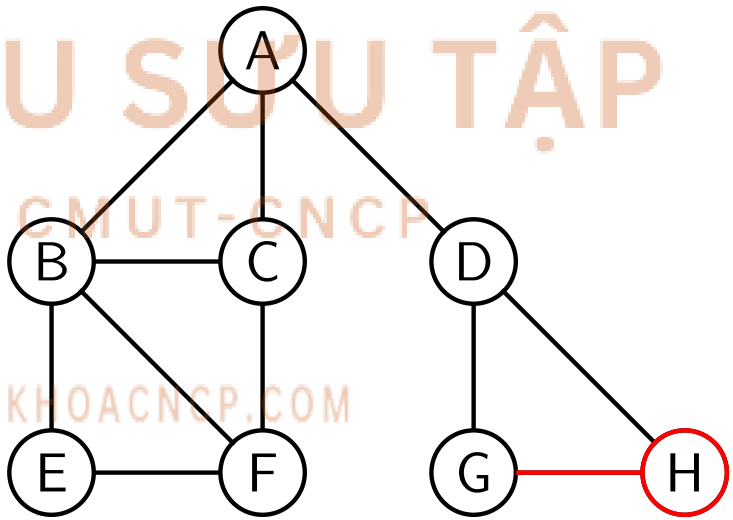
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	WG	12	•

time = 12



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

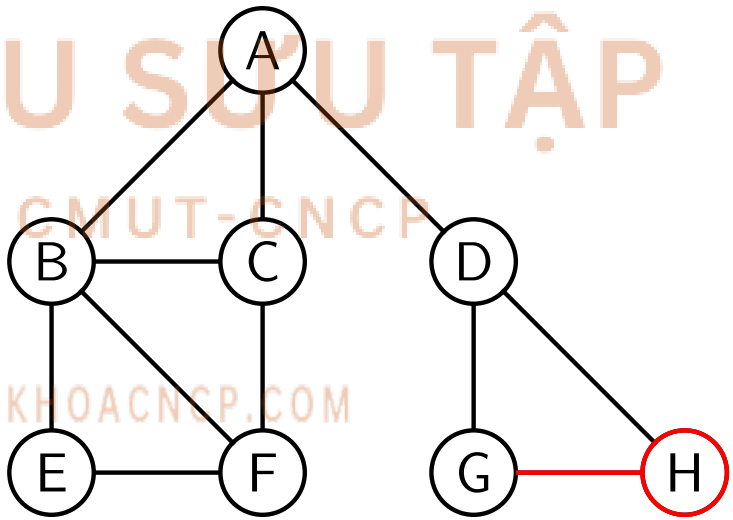
void DFS(G)

```
1.loop (more vertex v in G)
   1.color[v]= White
   2.father[v]=null
2.time=0
3.loop (more vertex v in G )
   1.if (color[v]== White)
      1.DFSVisit(G, v)
```

void DFSVisit (G, v)

```
1.color[v]=Gray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
   1.if(color[u]==White)
      1. father[u]=v
      2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time
```

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	WGB	12	•
time =		12		



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

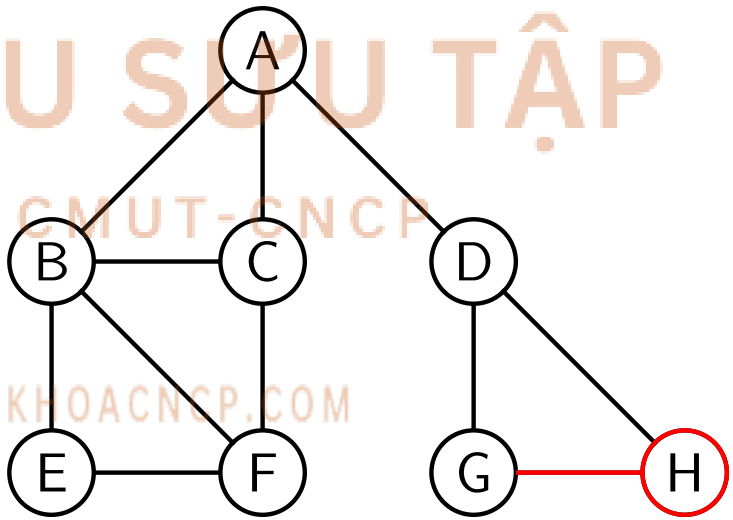
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	WGB	12	•

time = 13



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

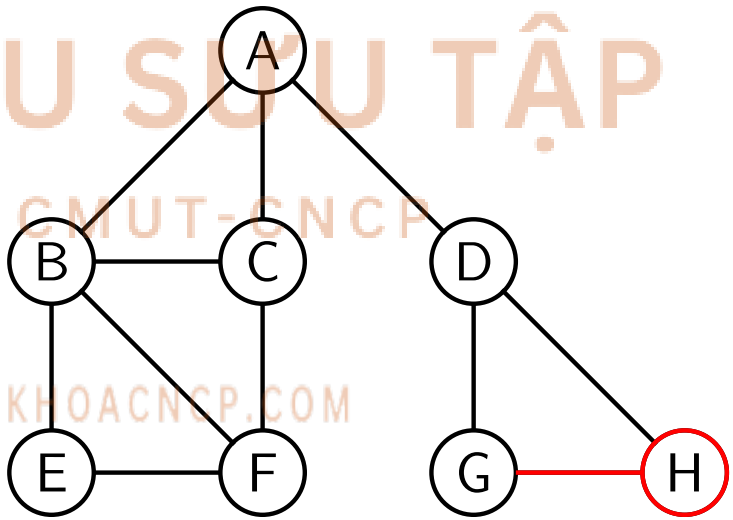
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	WGB	12	13

time = 13



A pseudocode of DFS

void DFS(G)

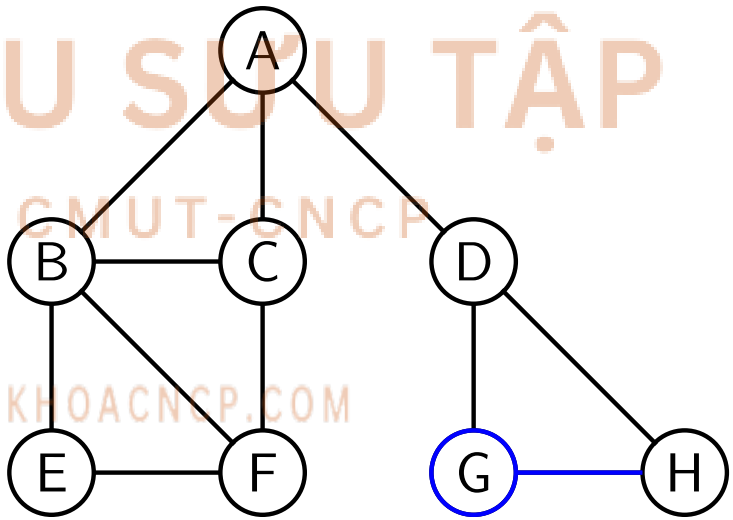
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WG	11	•
H	G	WGB	12	13

time = 13



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

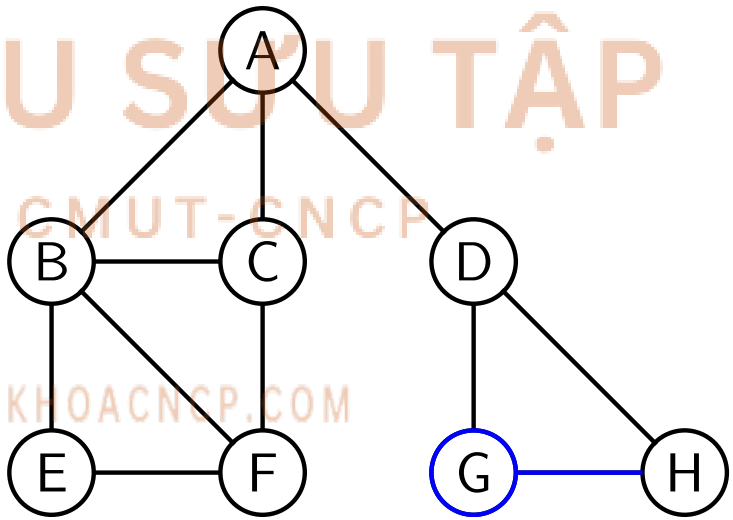
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	•
H	G	WGB	12	13

time = 13



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

Huynh Tuong Nguyen,
Nguyen Ngoc Le

void DFS(G)

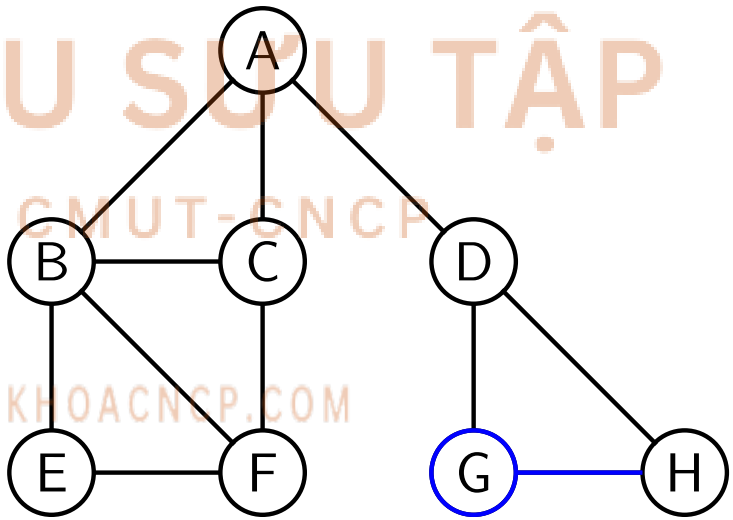
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	•
H	G	WGB	12	13

time = 14



A pseudocode of DFS

void DFS(G)

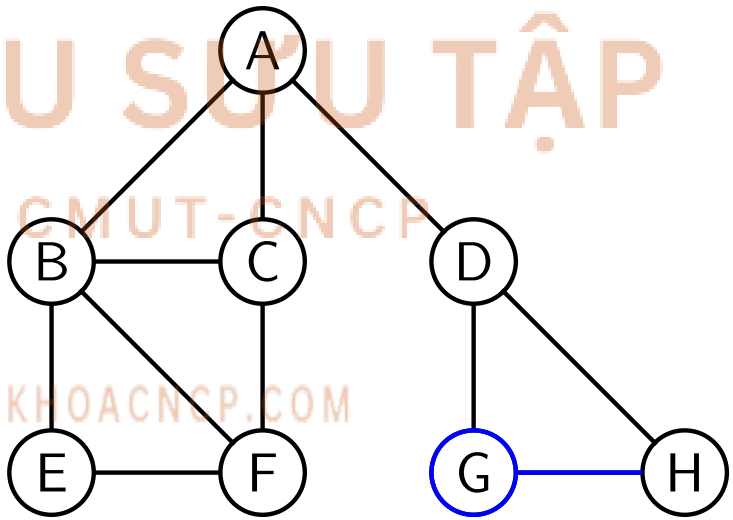
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 14



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

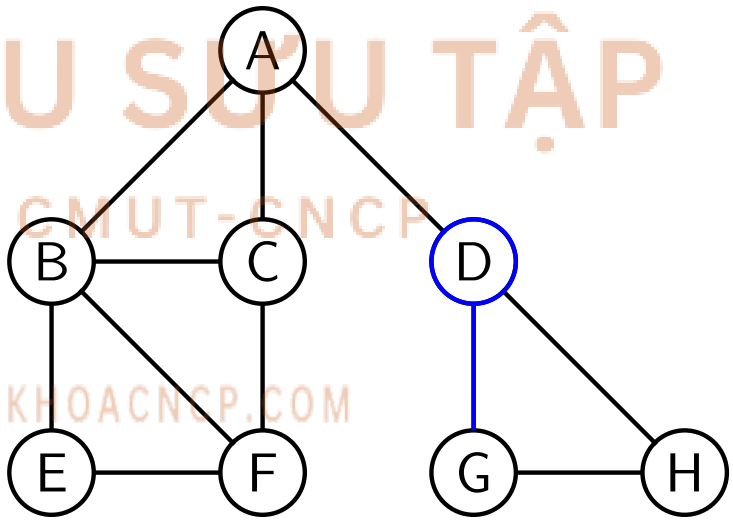
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WG	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 14



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

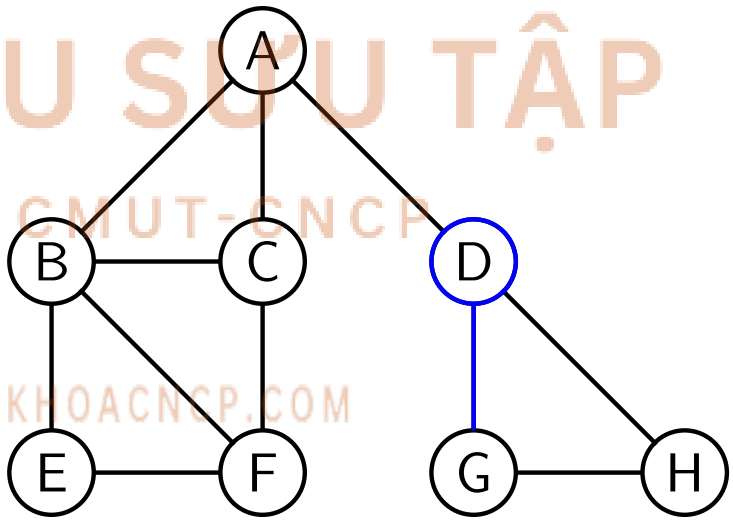
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WGB	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 14



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

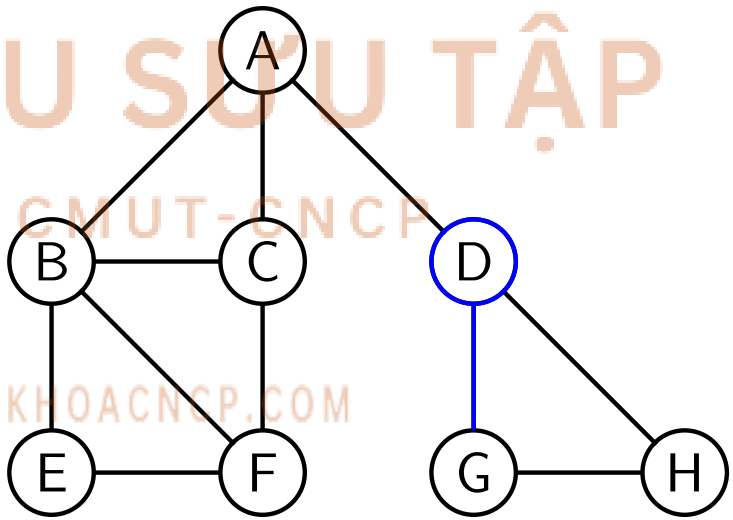
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WGB	10	•
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 15



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

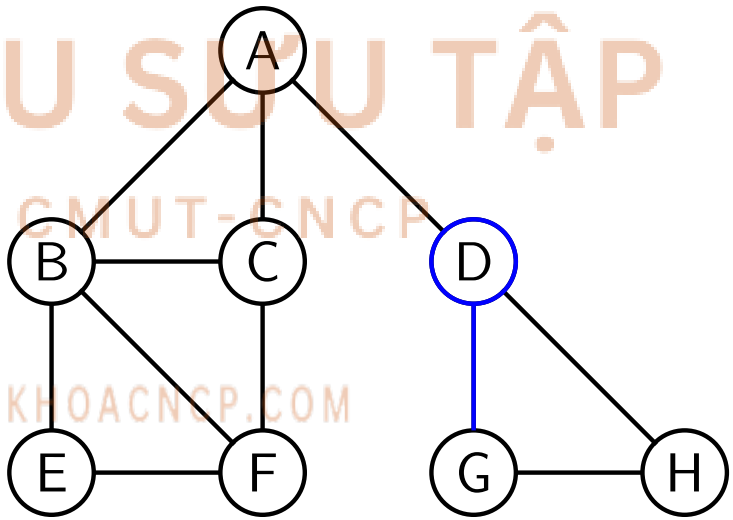
```
1.loop (more vertex v in G)
   1.color[v]= White
   2.father[v]=null
2.time=0
3.loop (more vertex v in G )
   1.if (color[v]== White)
      1.DFSVisit(G, v)
```

void DFSVisit (G, v)

```
1.color[v]=Gray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
   1.if(color[u]==White)
      1. father[u]=v
      2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time
```

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WGB	10	15
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 15



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

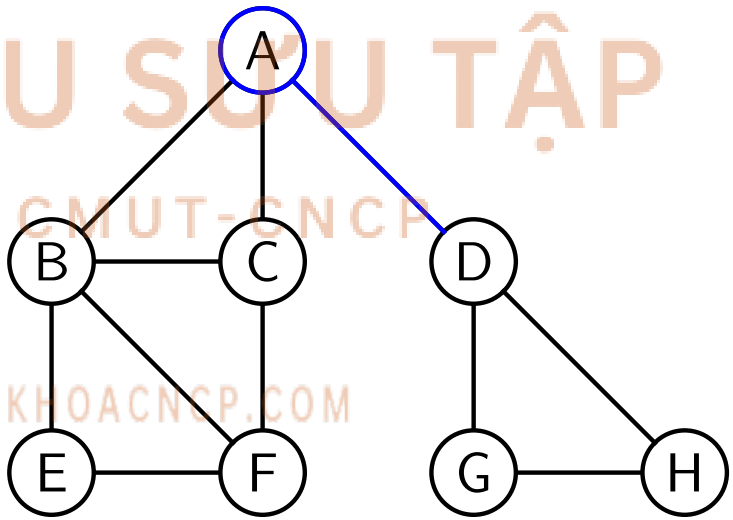
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WG	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WGB	10	15
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 15



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

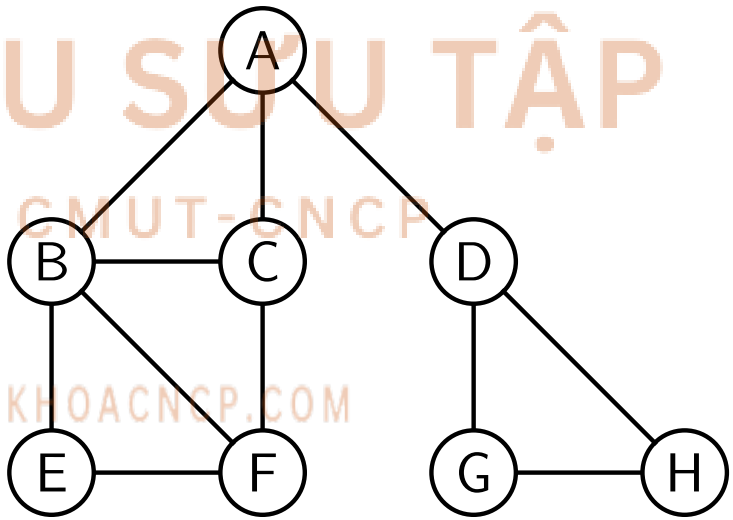
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=Black
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WGB	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WGB	10	15
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 15



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

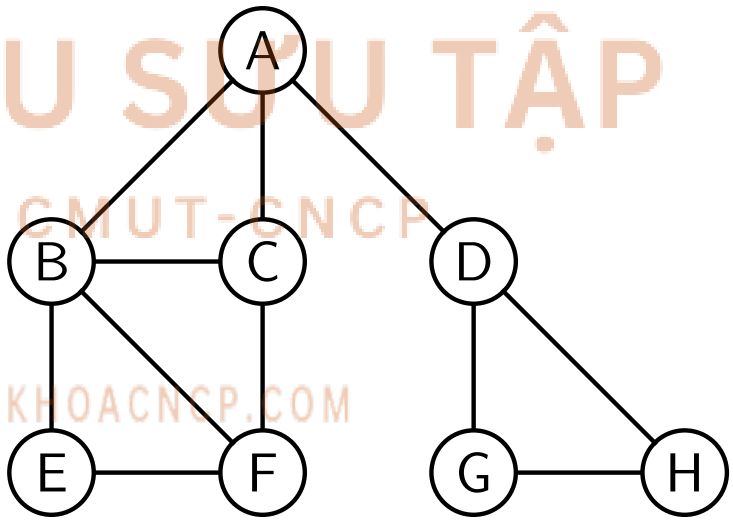
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WGB	1	•
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WGB	10	15
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 16



Huynh Tuong Nguyen,
Nguyen Ngoc Le

A pseudocode of DFS

void DFS(G)

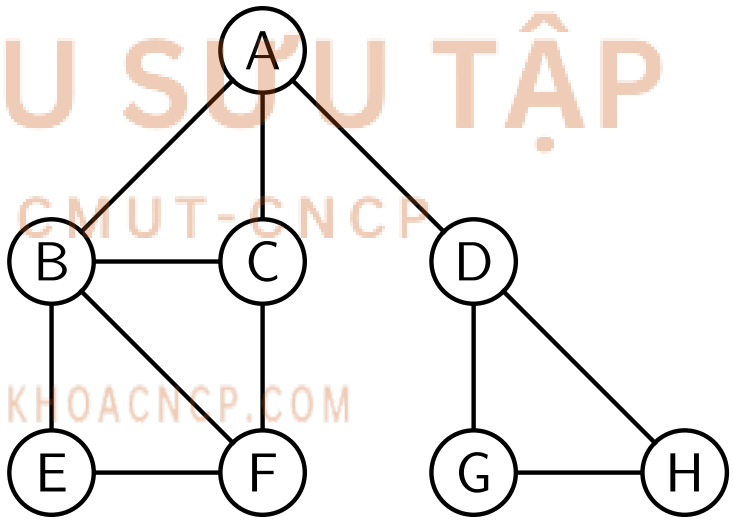
1.loop (more vertex v in G)
  1.color[v]= **W**hite
  2.father[v]=*null*
2.time=0
3.loop (more vertex v in G)
  1.if (color[v]== **W**hite)
    1.DFSVisit(G, v)

void DFSVisit (G, v)

1.color[v]=**G**ray
2.time= time+1
3.d[v]= time
4.loop(more u adjacent to v)
  1.if(color[u]==**W**hite)
    1. father[u]=v
    2.DFSVisit(G, u)
5.color[v]=**B**lack
6.time=time+1
7.f[v]=time

vertex	father	color	d	f
A	-	WGB	1	16
B	A	WGB	2	9
C	B	WGB	3	8
D	A	WGB	10	15
E	F	WGB	5	6
F	C	WGB	4	7
G	D	WGB	11	14
H	G	WGB	12	13

time = 16



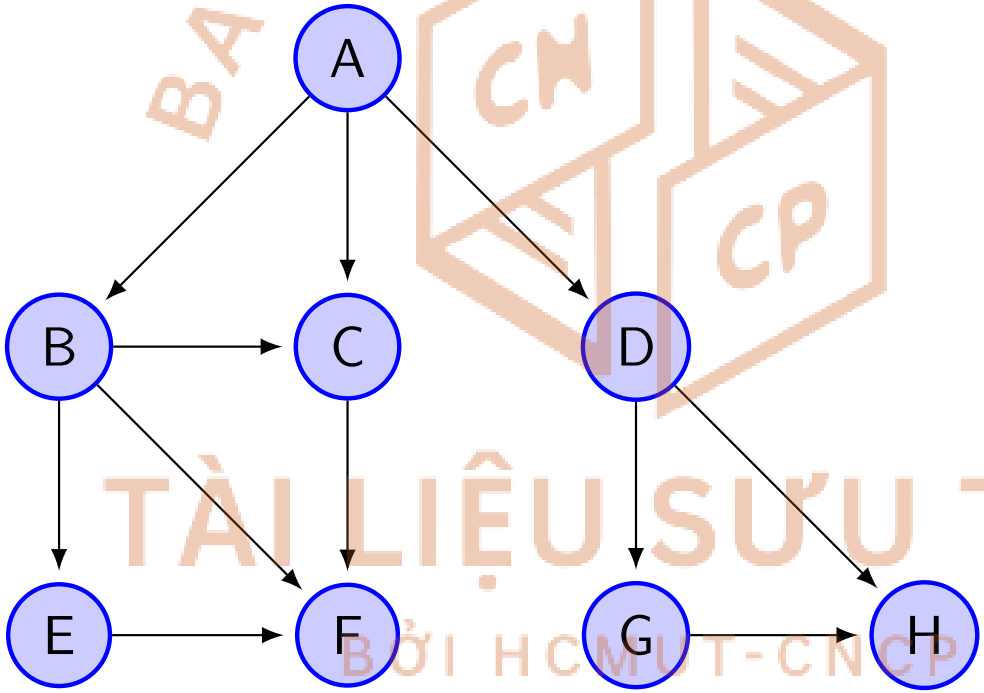
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.



Exercise

How to apply & modify DFS algorithm to ...

a. determine whether there exists a cycle in digraph,
b. determine whether a graph is bipartite,
c. determine topological order,
d. calculate number of connected components,
e. identify articulation points,
f. determine a longest path in a given digraph...

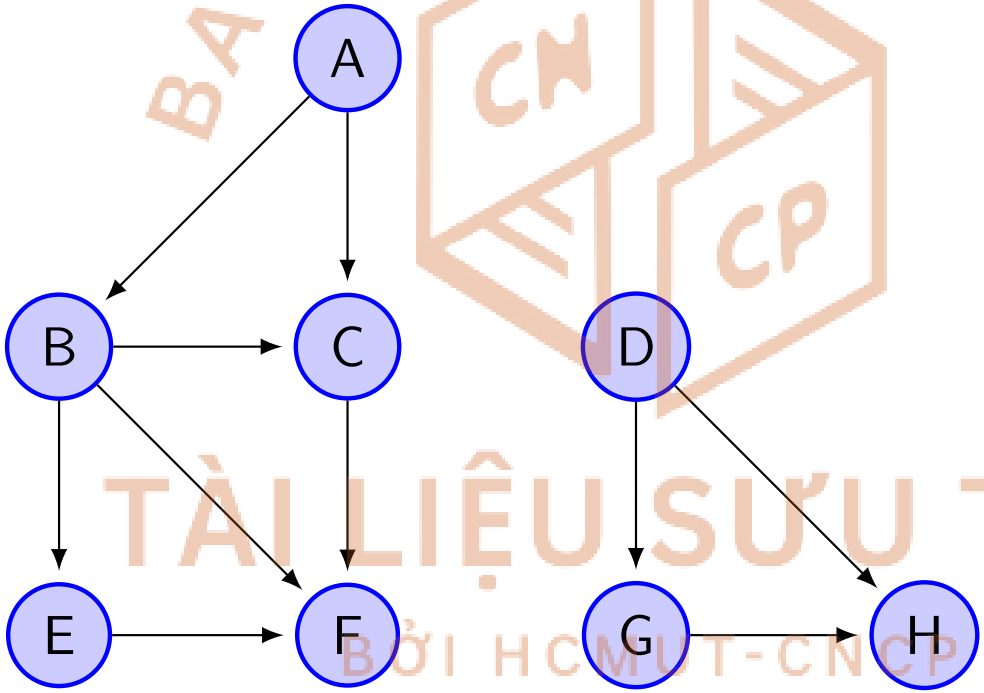
Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.

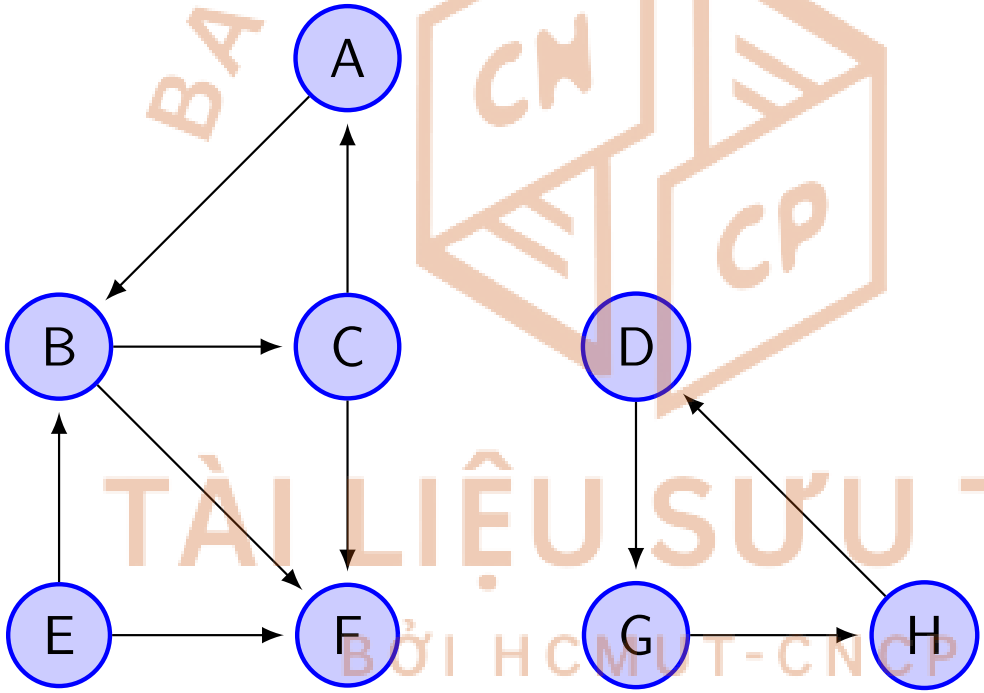


Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.

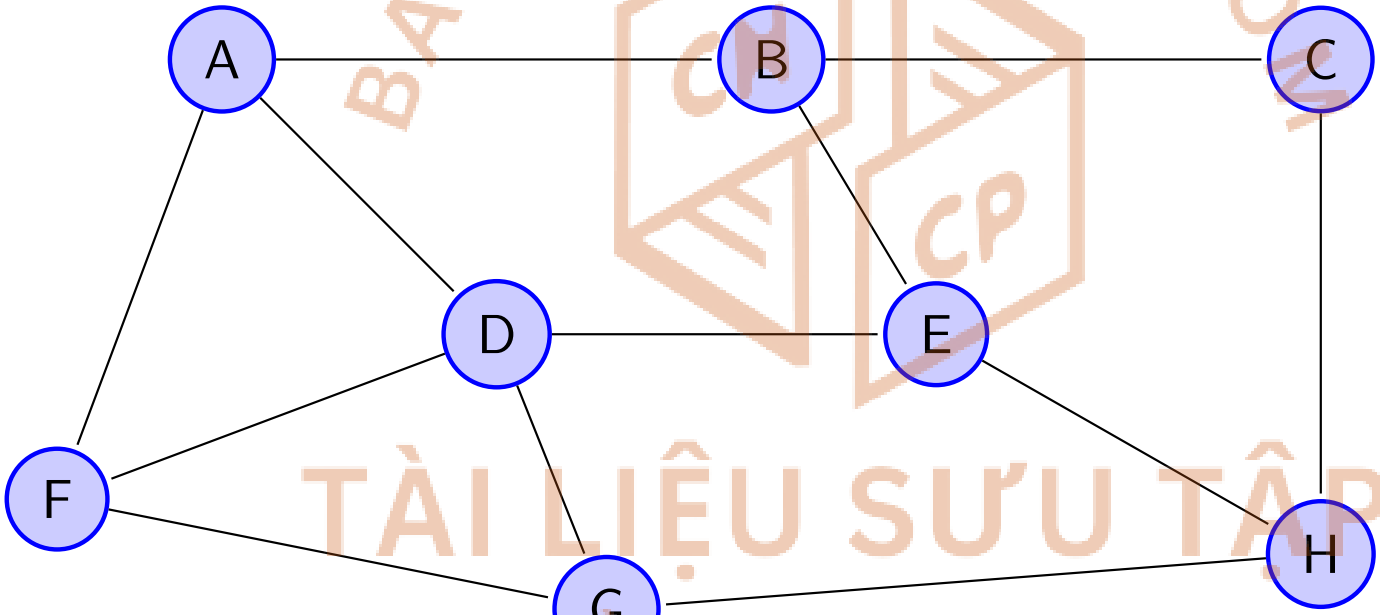


Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.

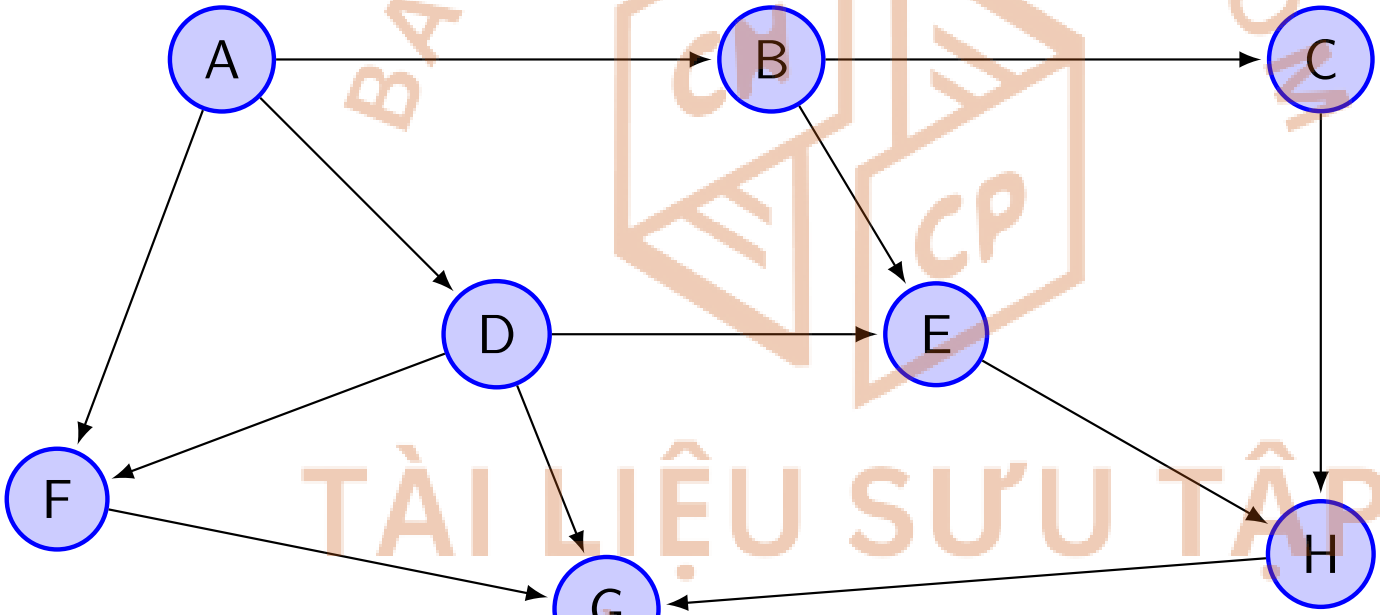


Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.

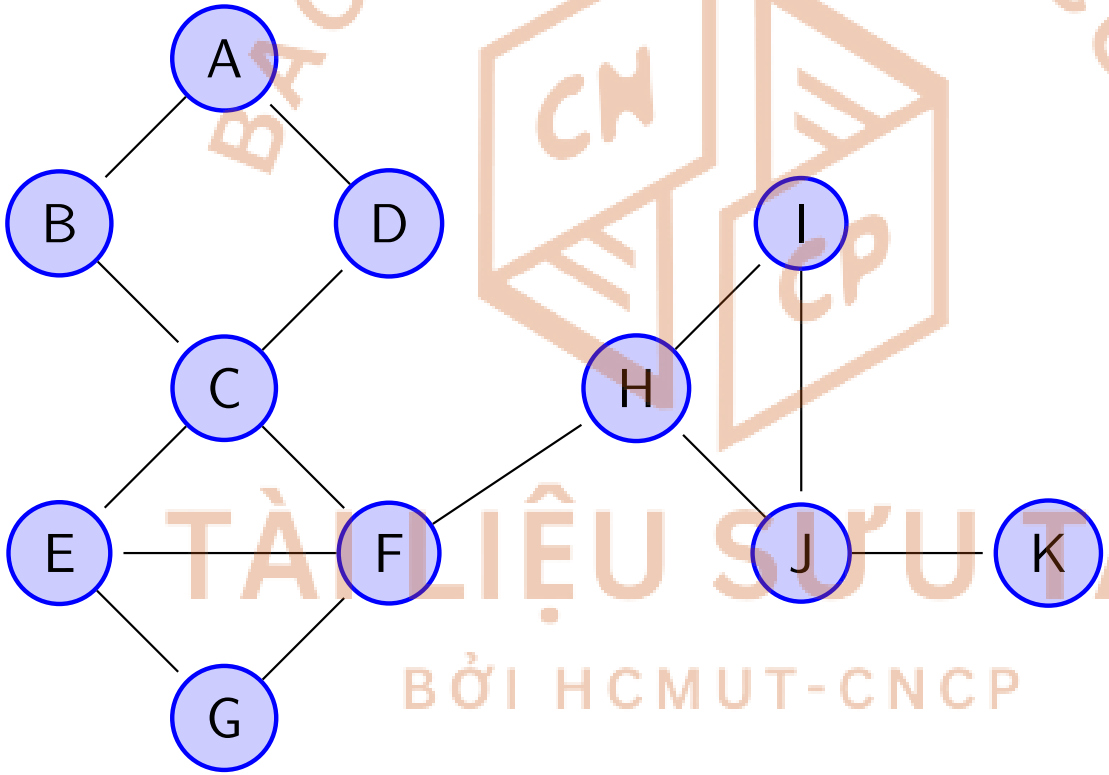


Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.

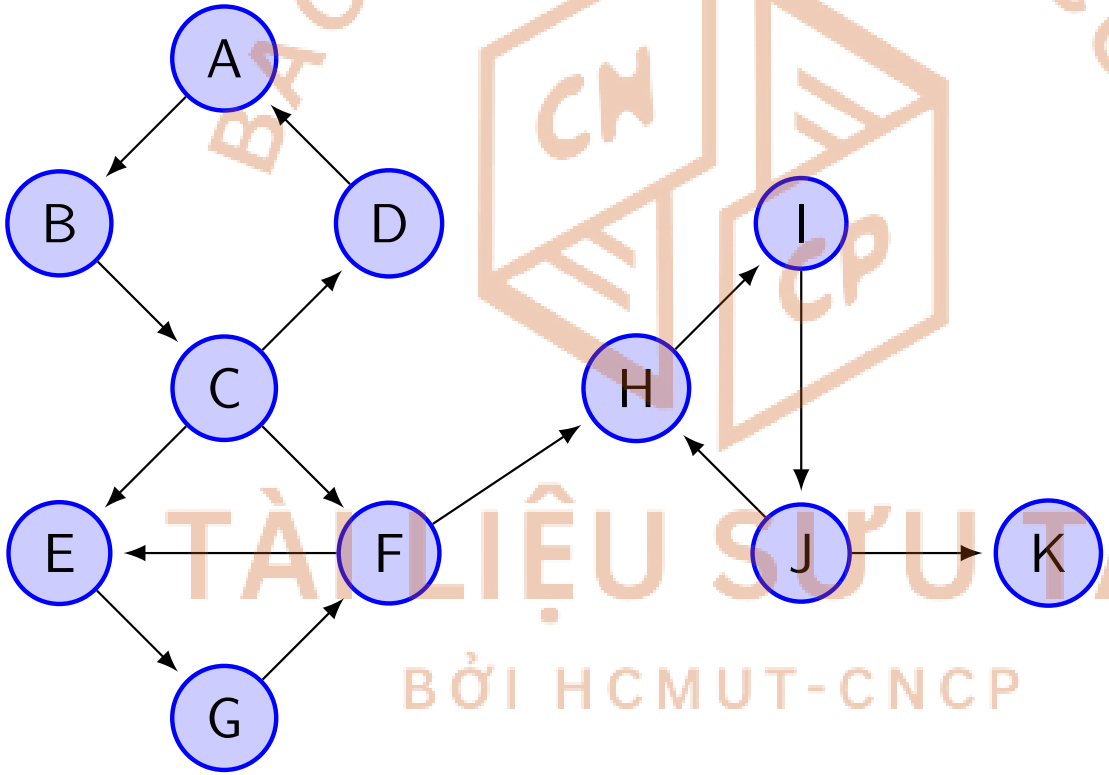


Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.

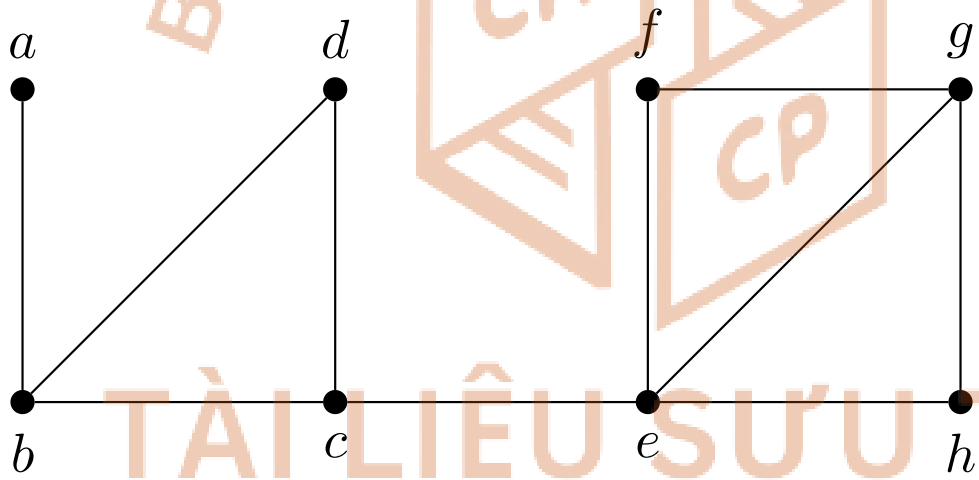


Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.



Exercise

Huynh Tuong Nguyen,
Nguyen Ngoc Le

Exercise

Apply DFS into the following graph.