



CS-2001 DATA STRUCTURE

Dr. Hashim Yasin

**National University of Computer
and Emerging Sciences,
Faisalabad, Pakistan.**

DEPTH FIRST SEARCH



Depth First Search

Go **as deep as** can visiting un-visited nodes

Choose any un-visited vertex when you have a choice

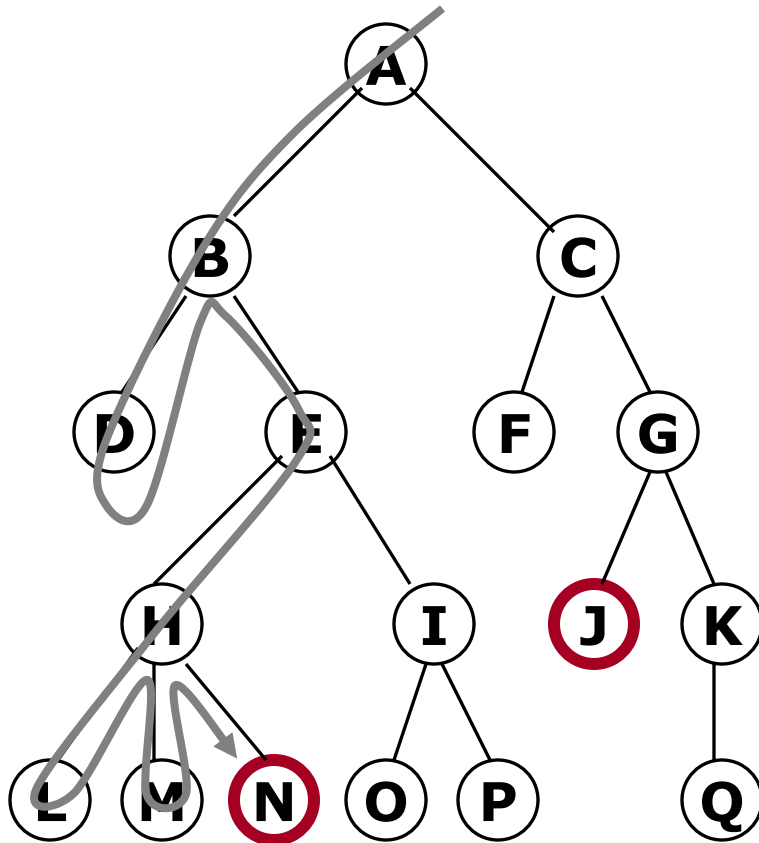
If stuck at a dead-end, backtrack as little as possible

Back up to where you could go to another unvisited vertex

Then continue to go on from that point

Eventually you'll return to where you started

Depth-first searching



A **depth-first search (DFS)** explores a path all the way to a leaf node before backtracking and exploring another path

For example, after searching A, then B, then D, the search backtracks and tries another path from B

Nodes are explored in the order A B D E H L M N I O P C F G J K Q

N will be found before J

DFS Algorithm

create a stack **S**

mark **v** as visited and push **v** onto **S**

while **S** is non-empty

 pop at the top **u** of **S**

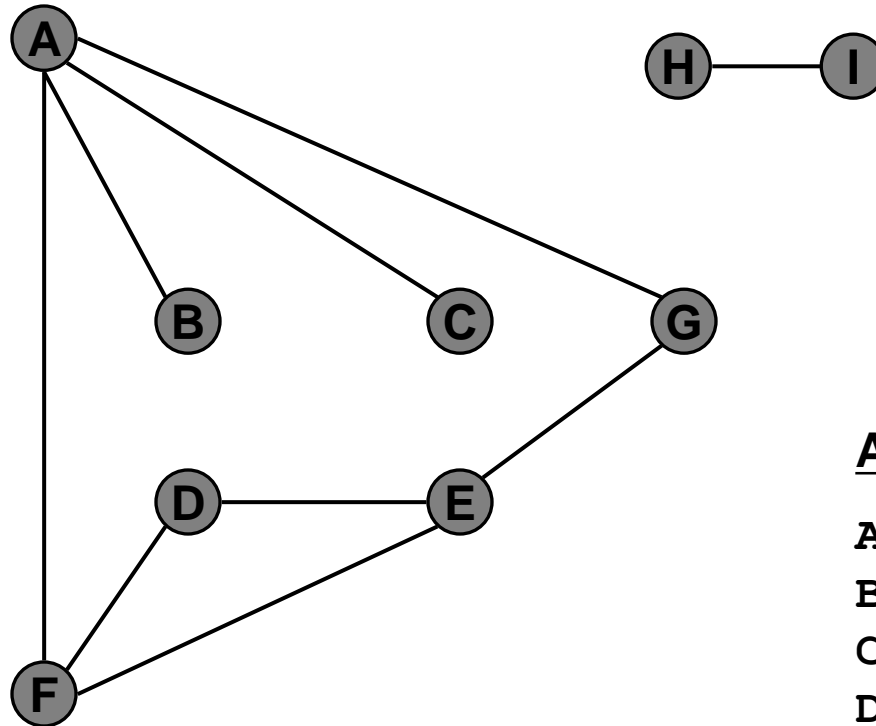
 if **u** has an (unvisited) neighbor **w**,

 mark **w** and push it onto **S**

 else

 pop **S**

Undirected Depth First Search



Adjacency Lists

A: F C B G

B: A

C: A

D: F E

E: G F D

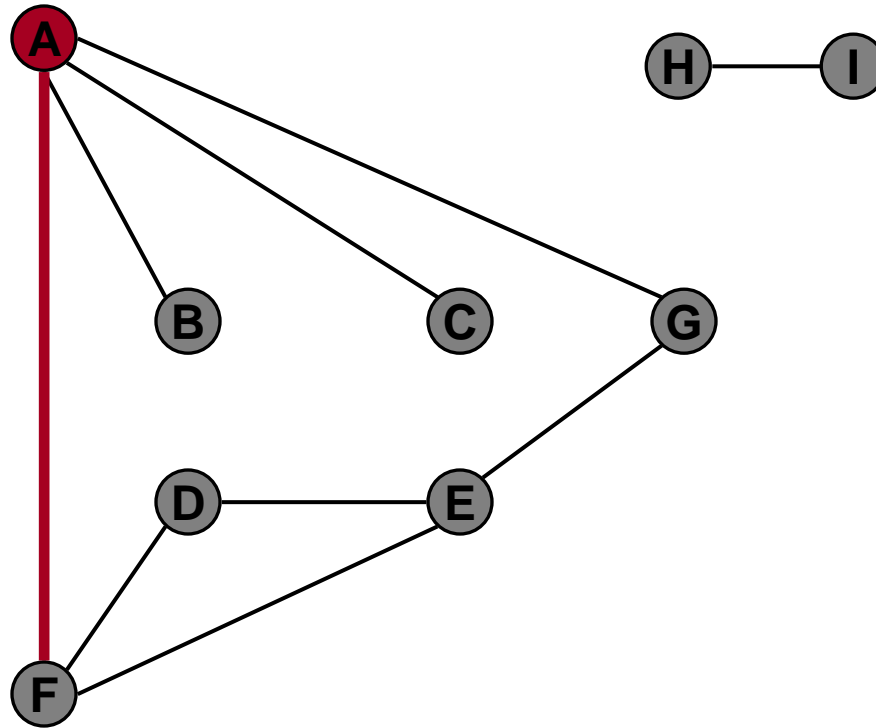
F: A E D:

G: E A:

H: I:

I: H:

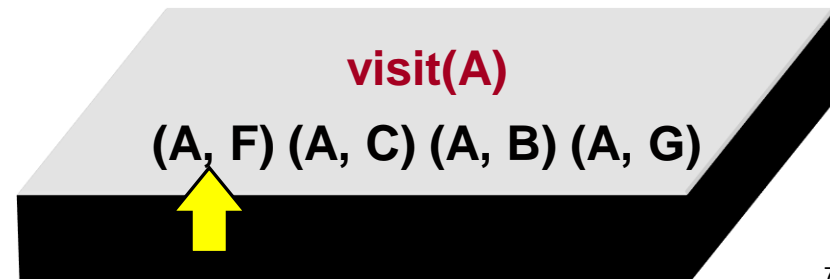
Undirected Depth First Search



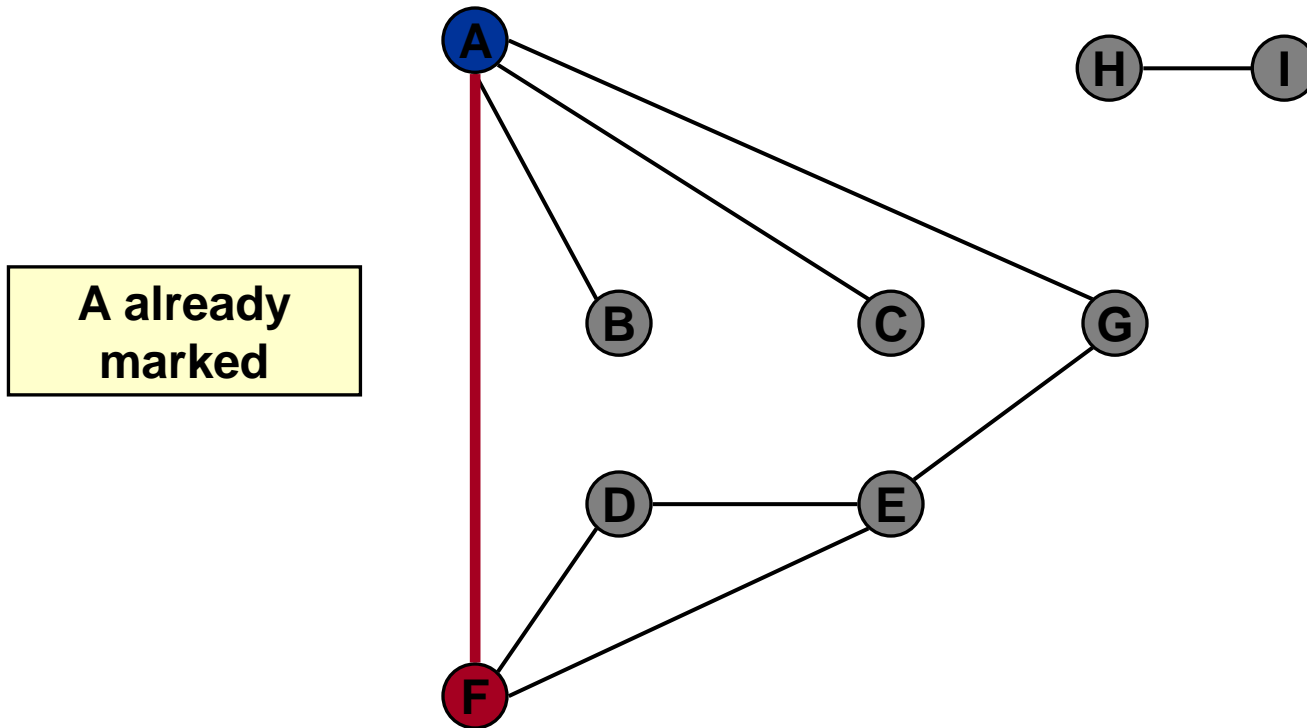
F newly
discovered



Stack

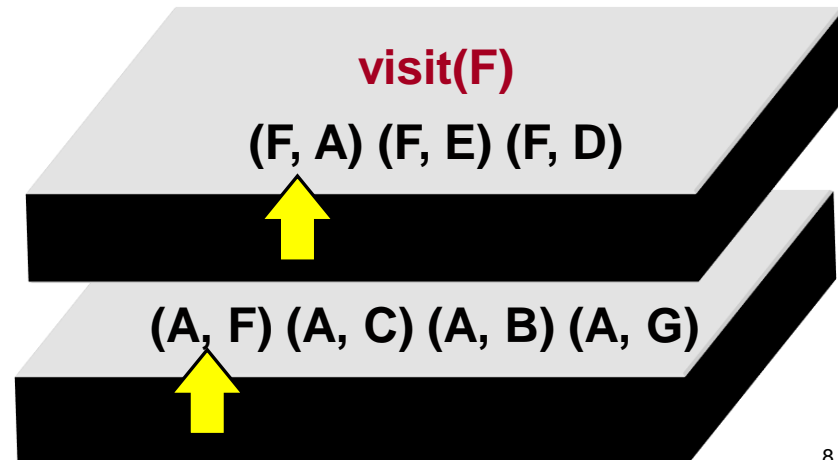


Undirected Depth First Search

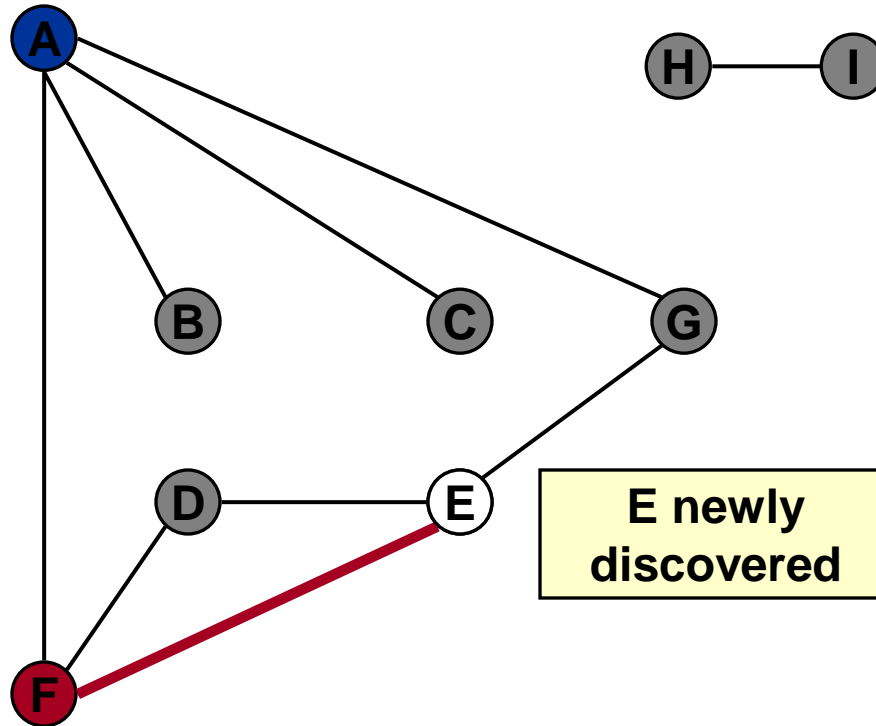


Undiscovered
Marked
Active
Finished

Stack

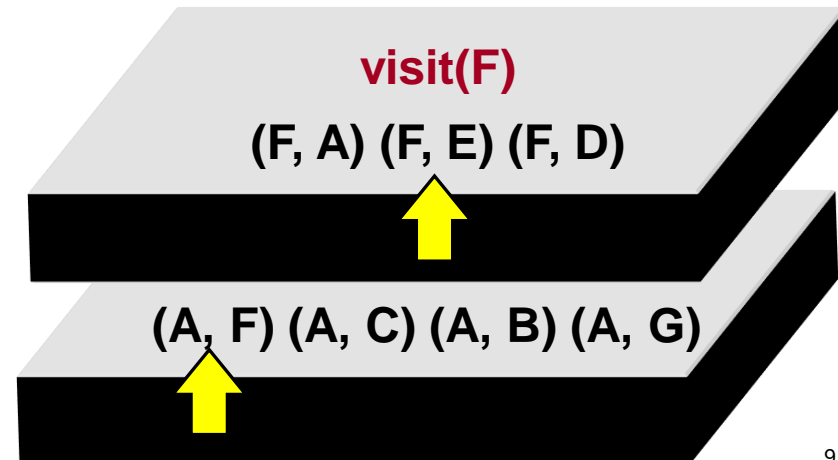


Undirected Depth First Search

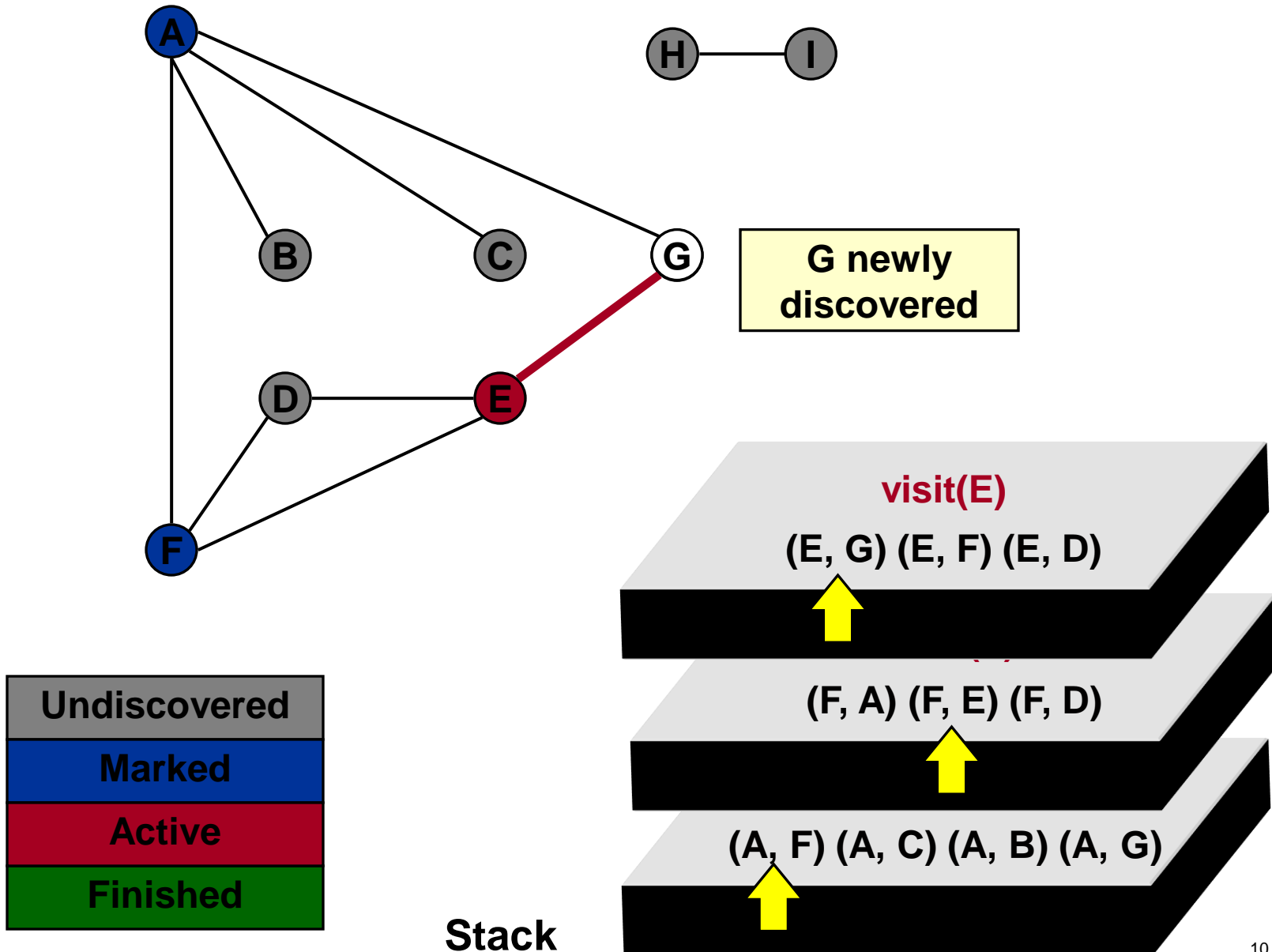


Undiscovered
Marked
Active
Finished

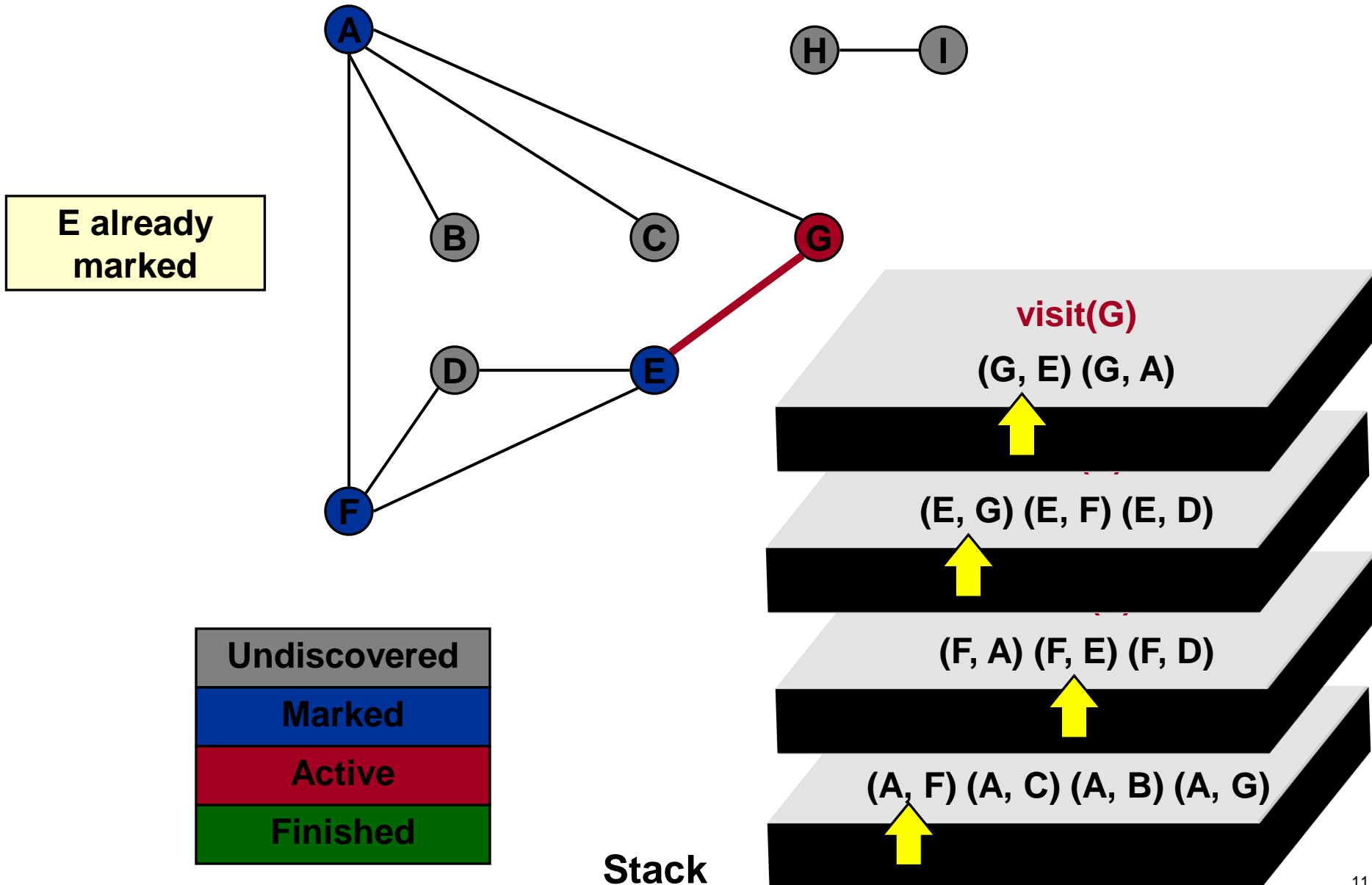
Stack



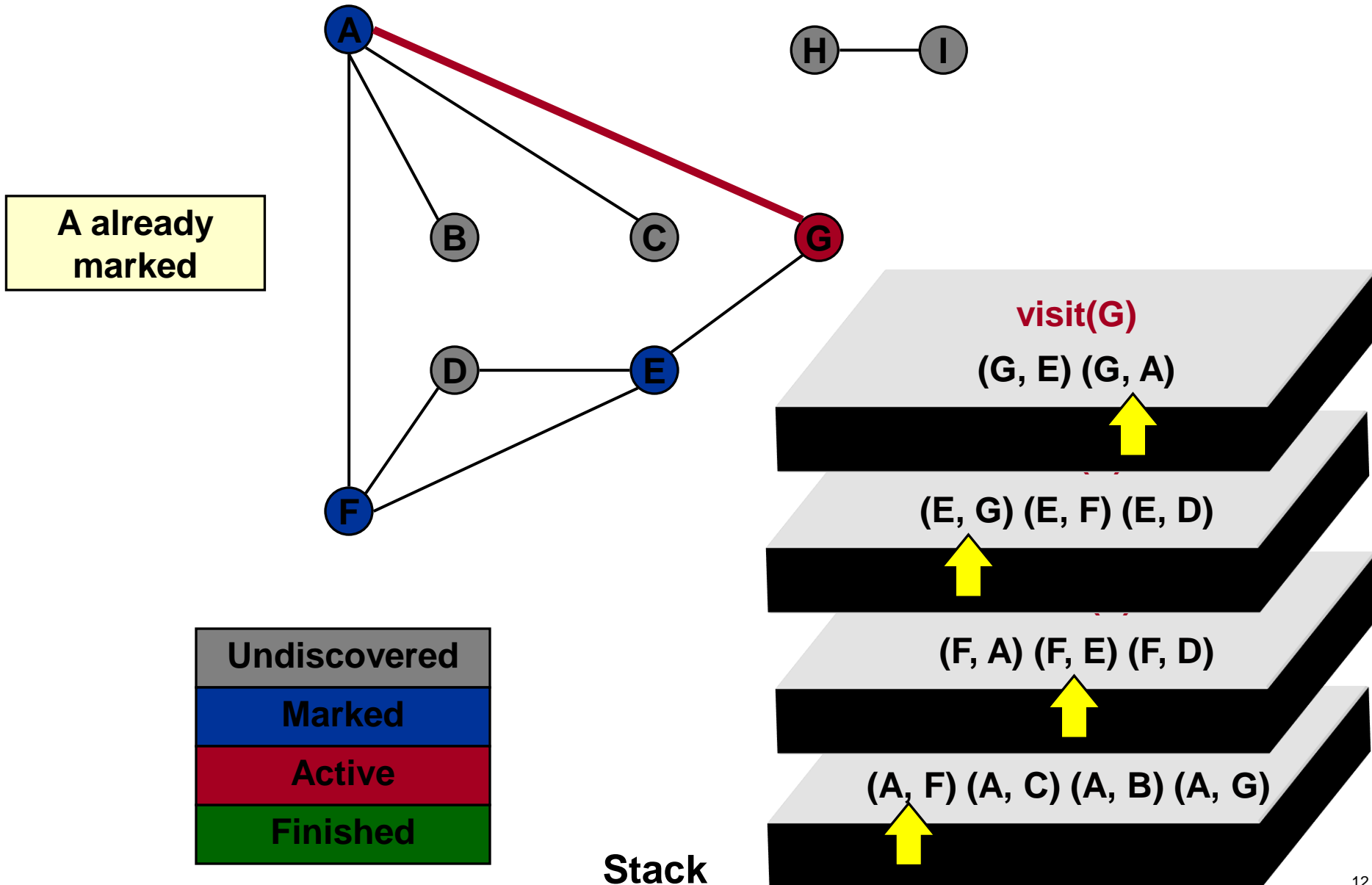
Undirected Depth First Search



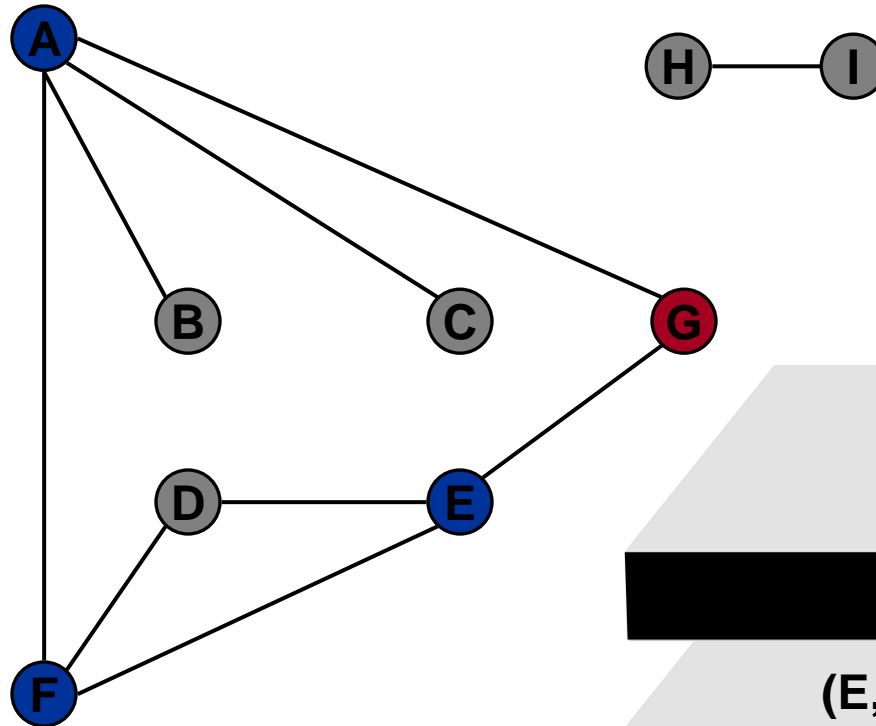
Undirected Depth First Search



Undirected Depth First Search

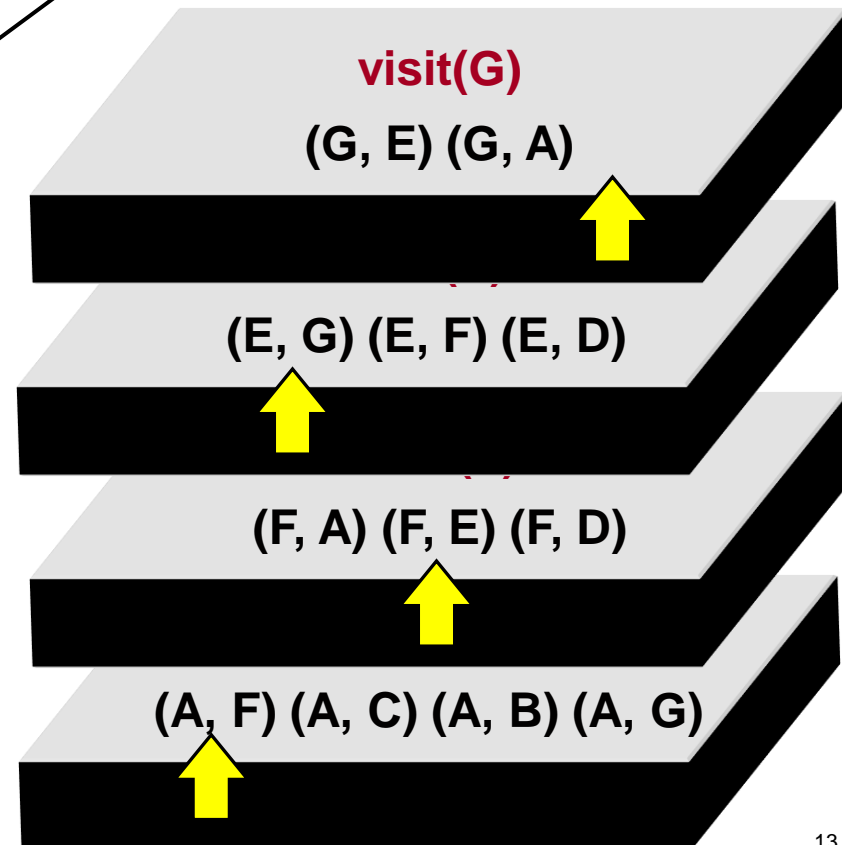


Undirected Depth First Search

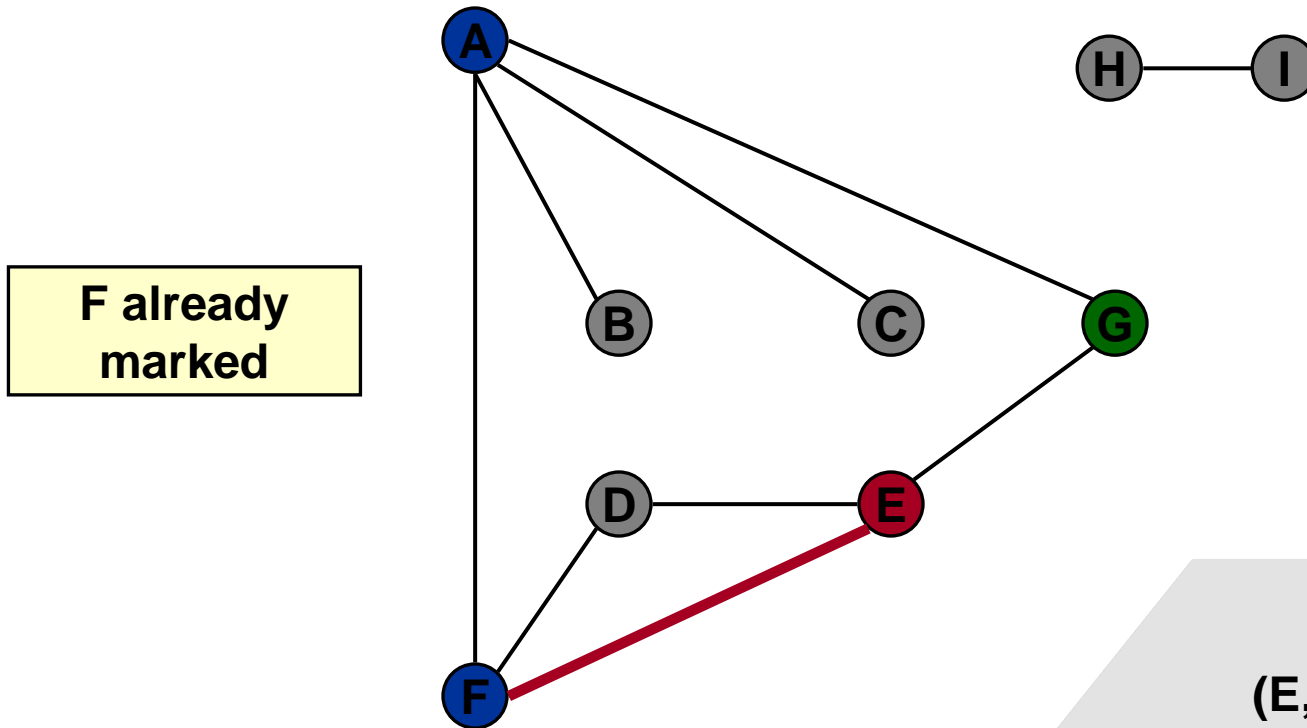


Finished G

Stack

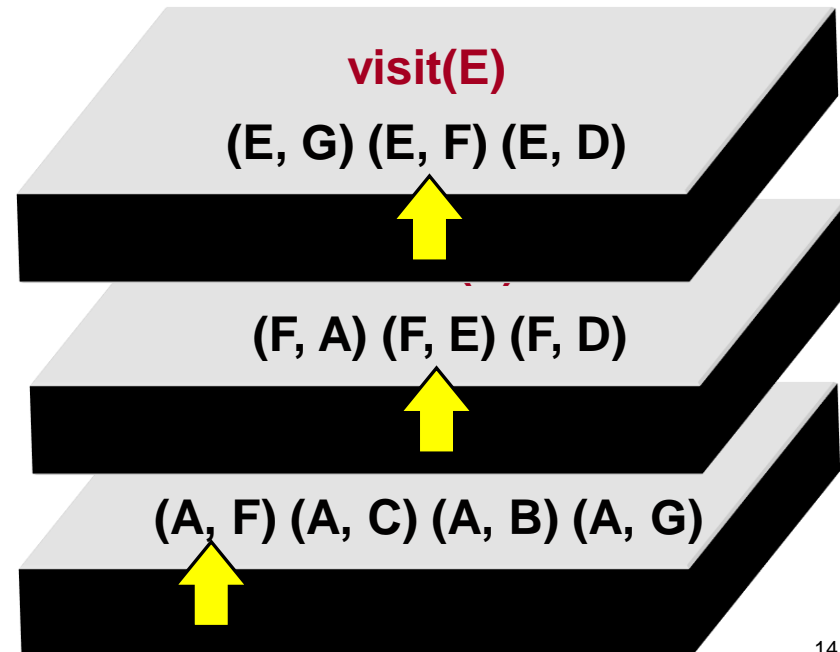


Undirected Depth First Search

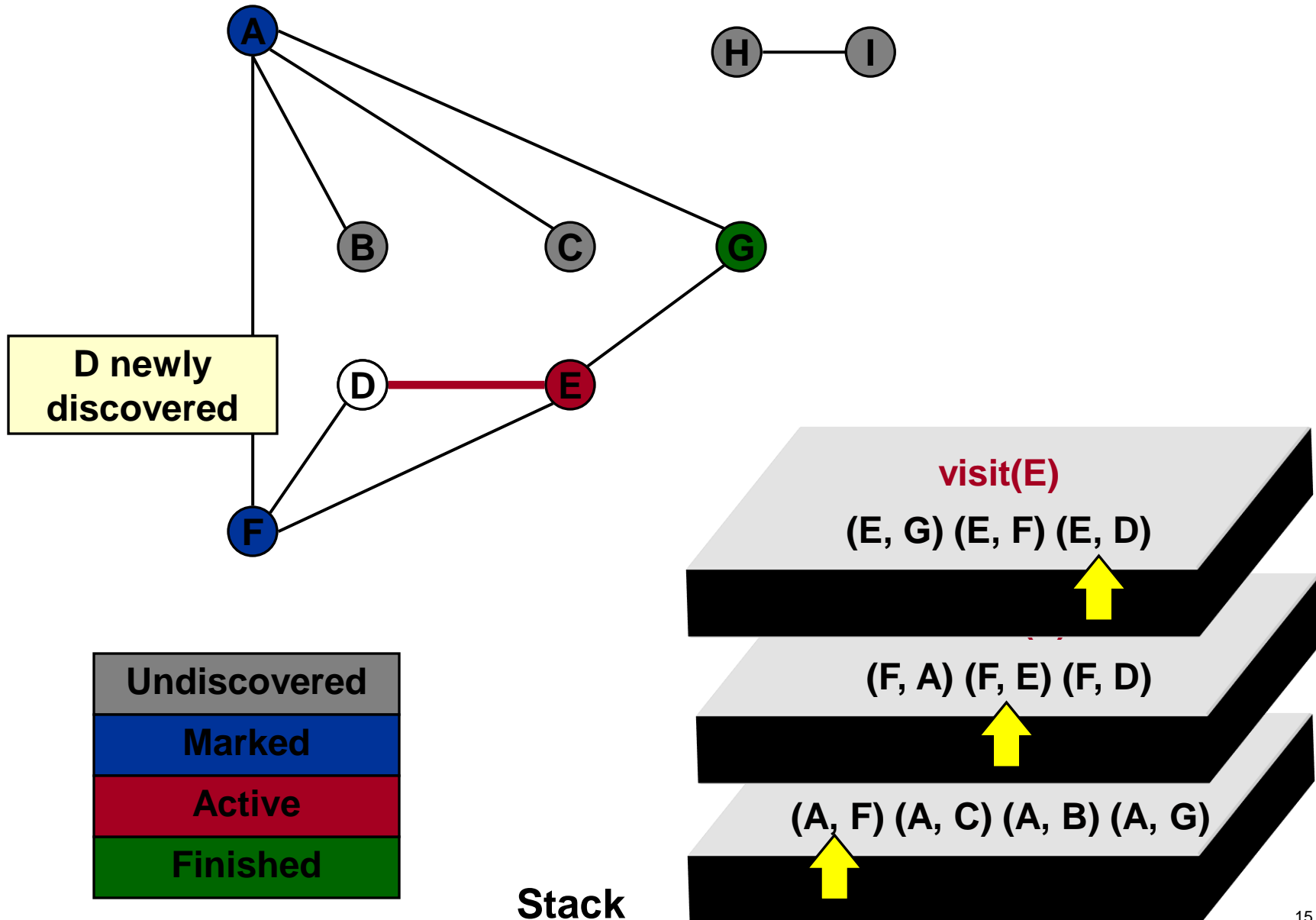


Undiscovered
Marked
Active
Finished

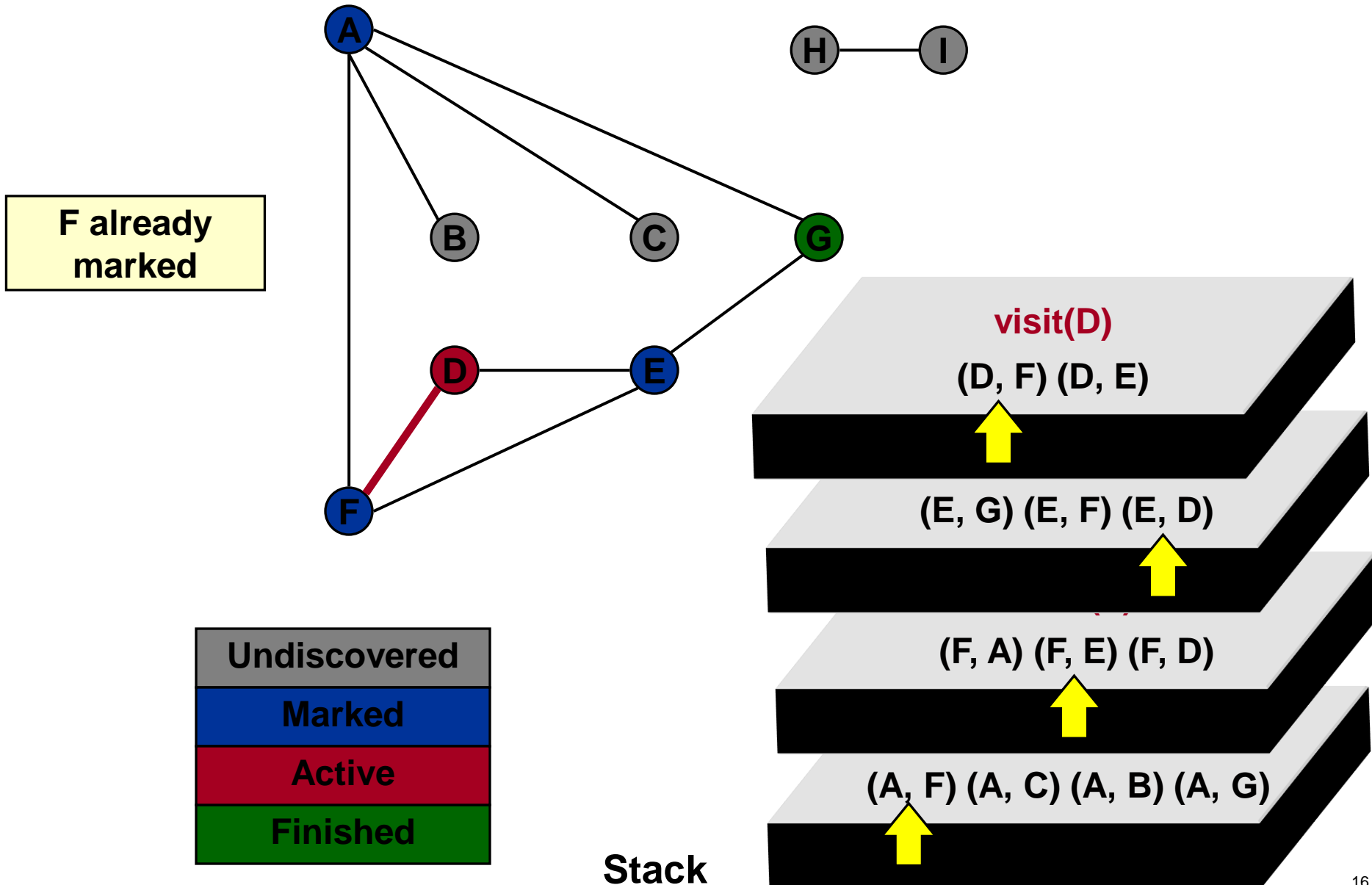
Stack



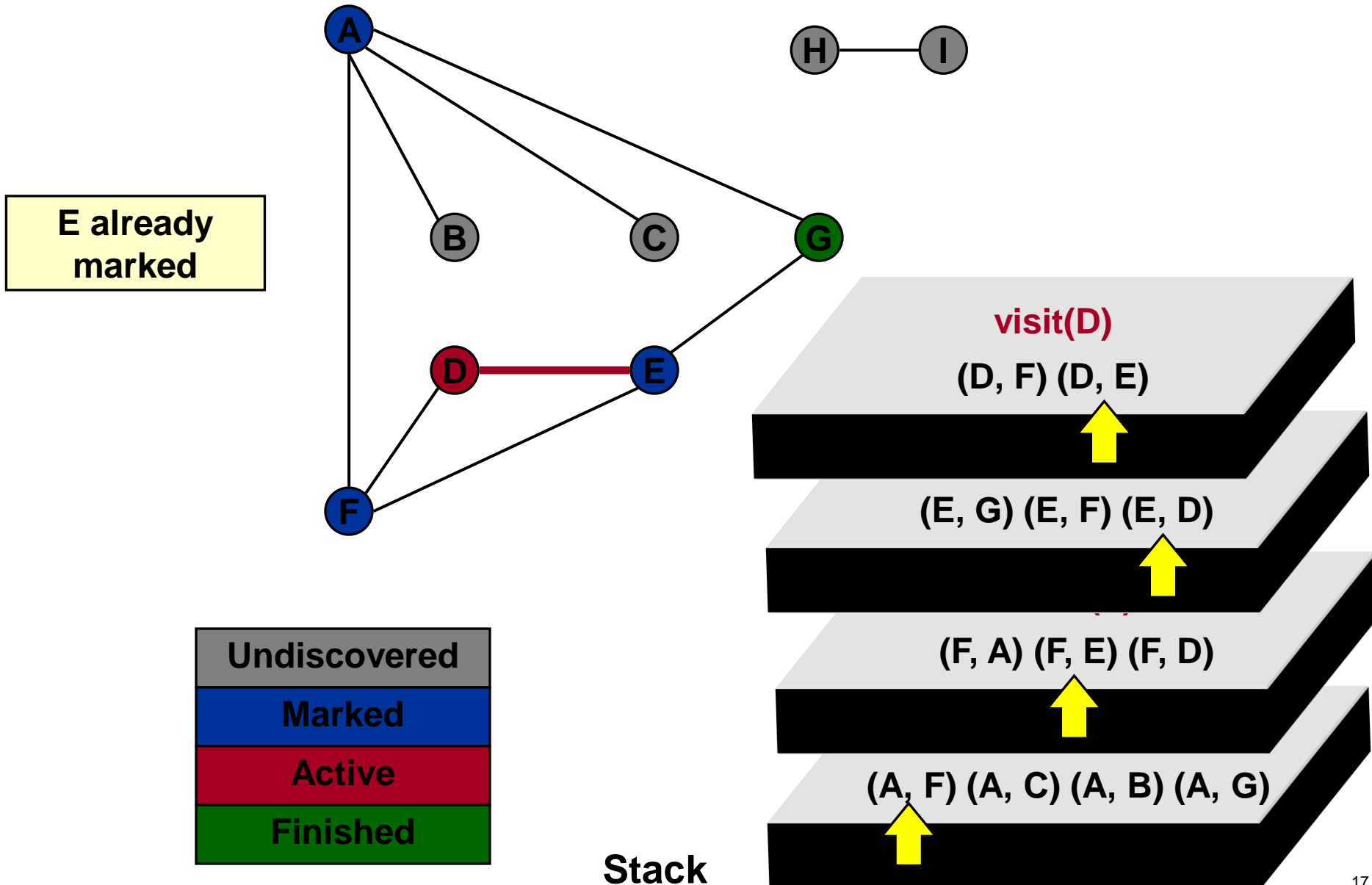
Undirected Depth First Search



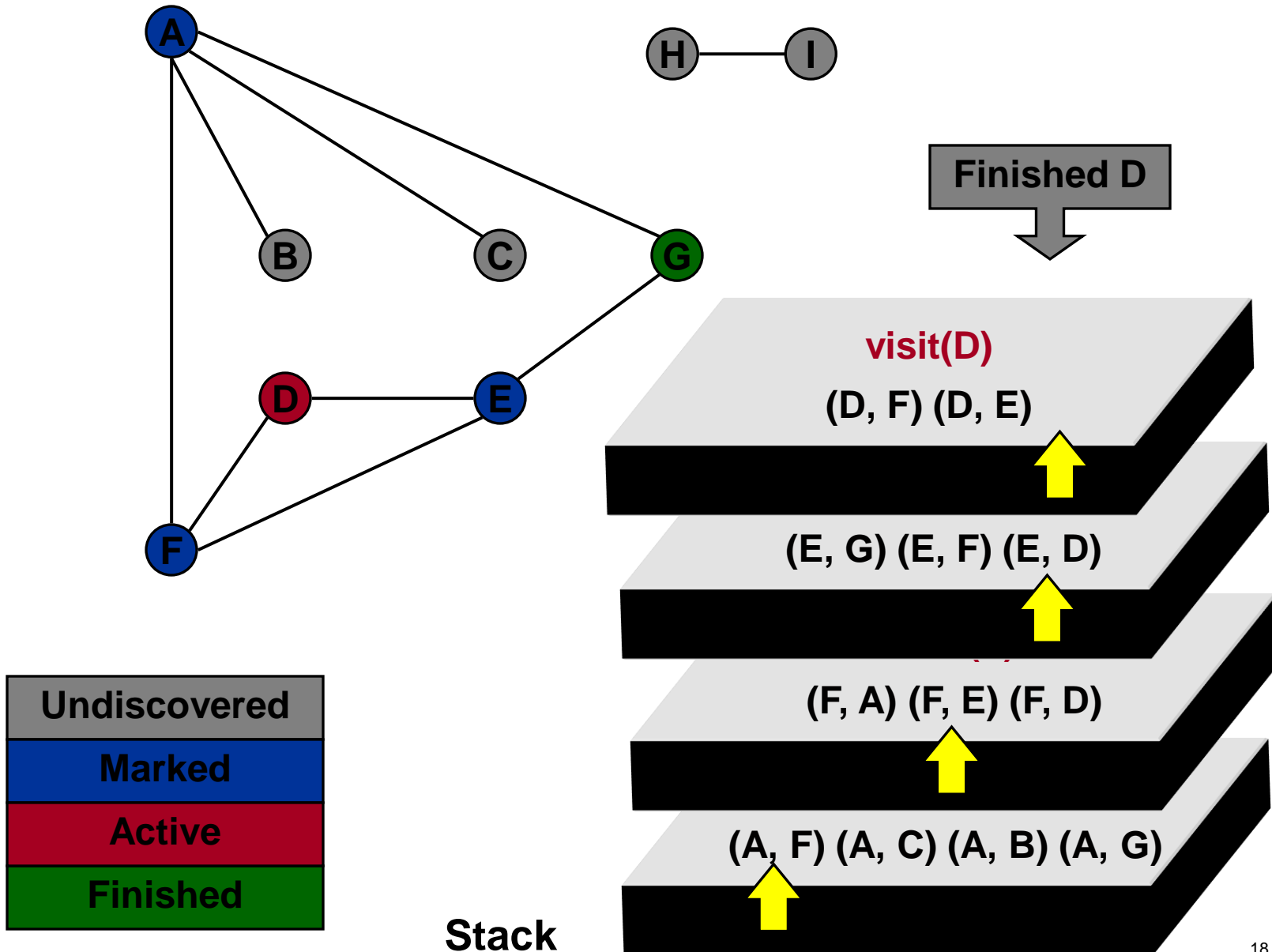
Undirected Depth First Search



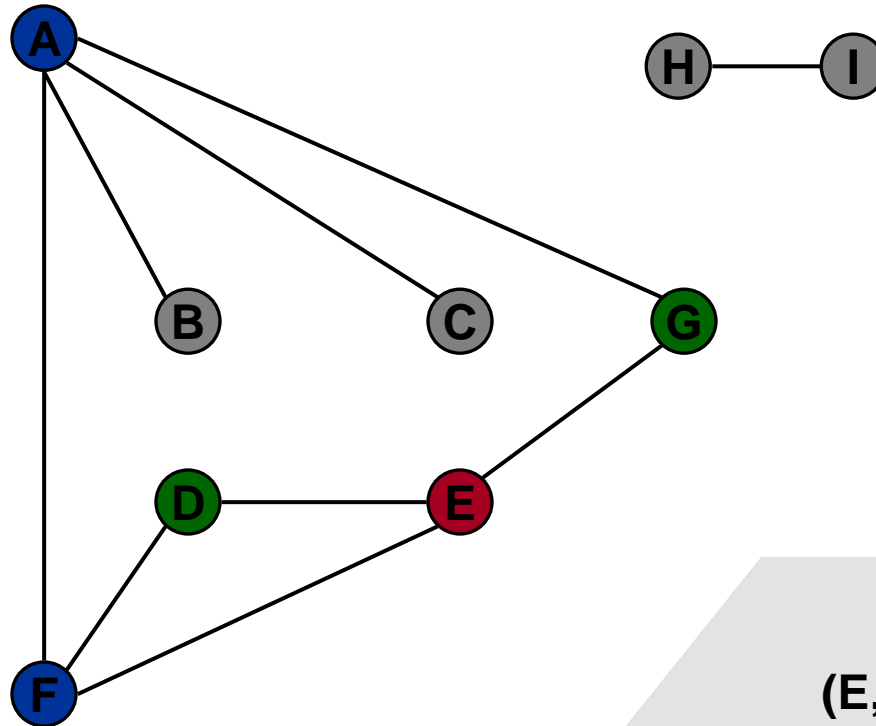
Undirected Depth First Search



Undirected Depth First Search

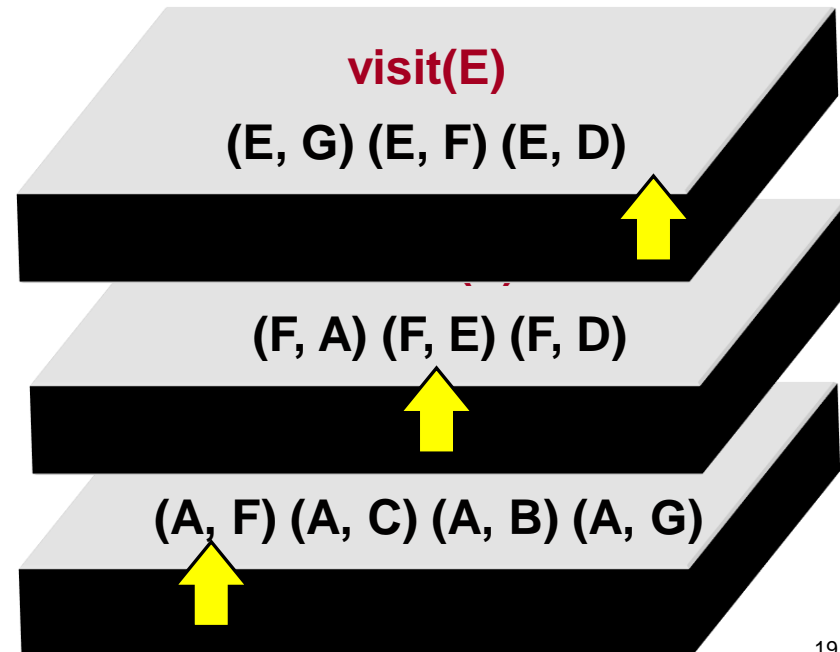


Undirected Depth First Search

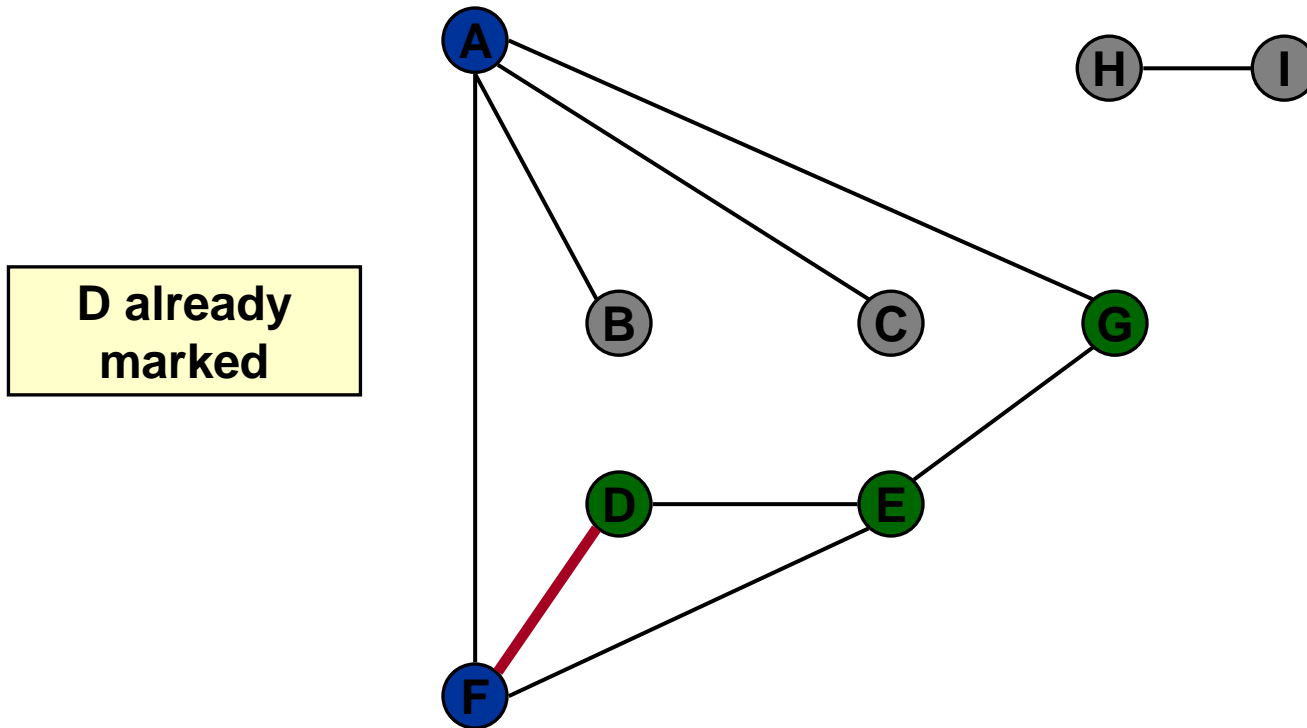


Finished E

Stack

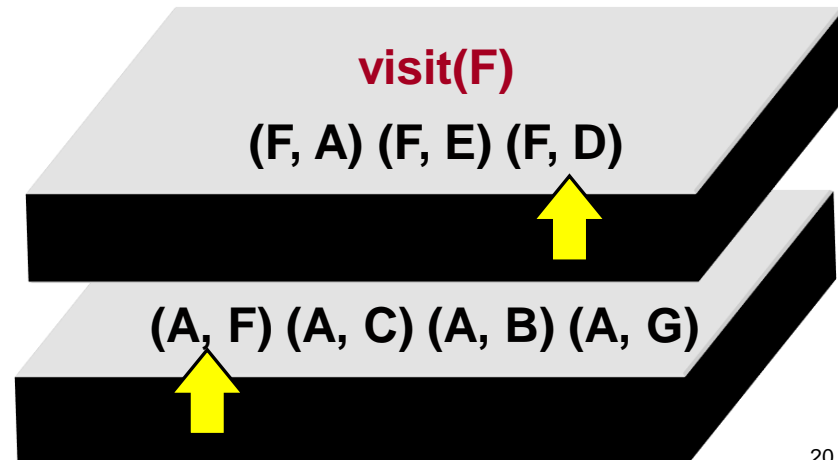


Undirected Depth First Search

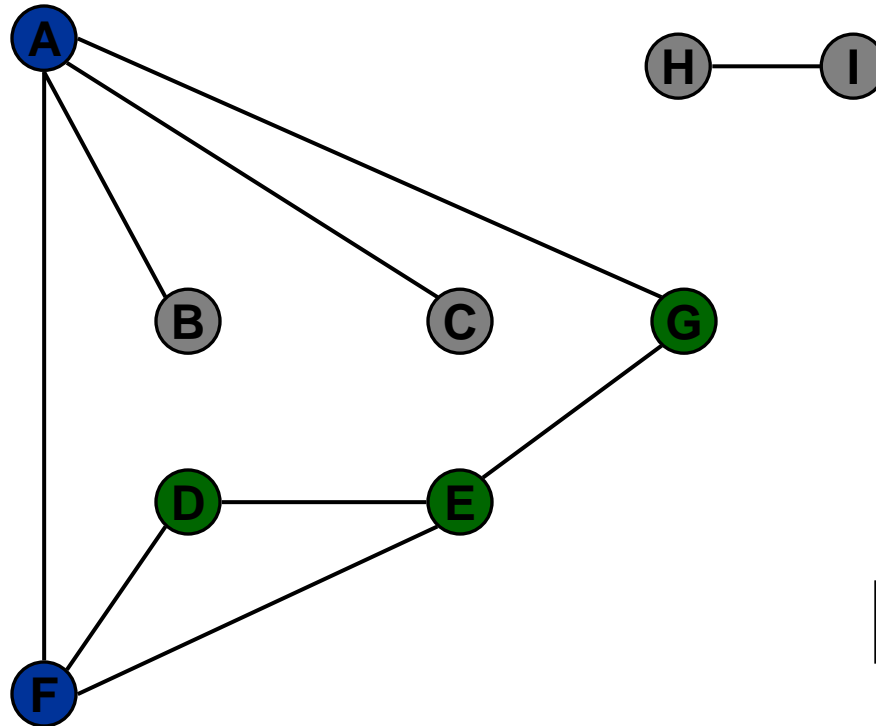


Undiscovered
Marked
Active
Finished

Stack



Undirected Depth First Search



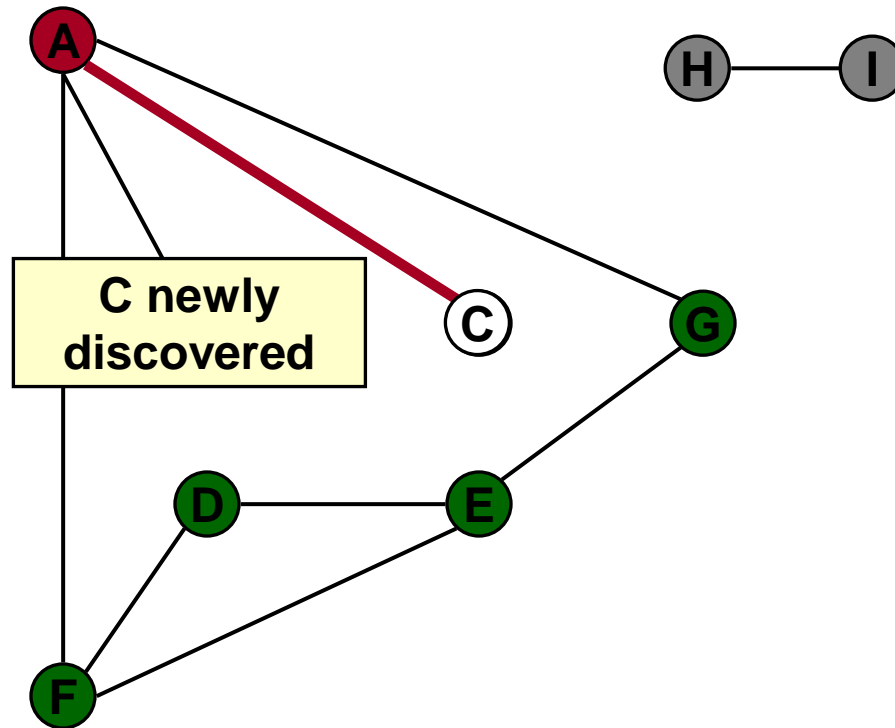
Undiscovered
Marked
Active
Finished

Finished F

Stack

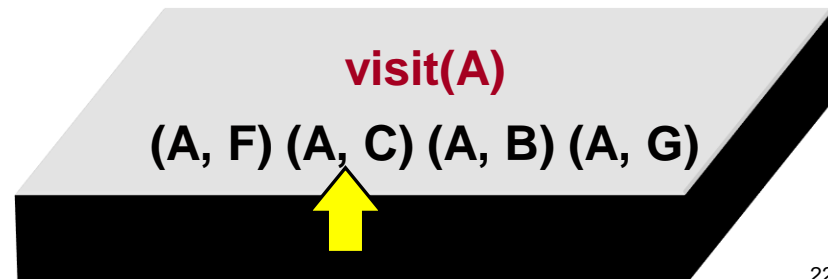


Undirected Depth First Search

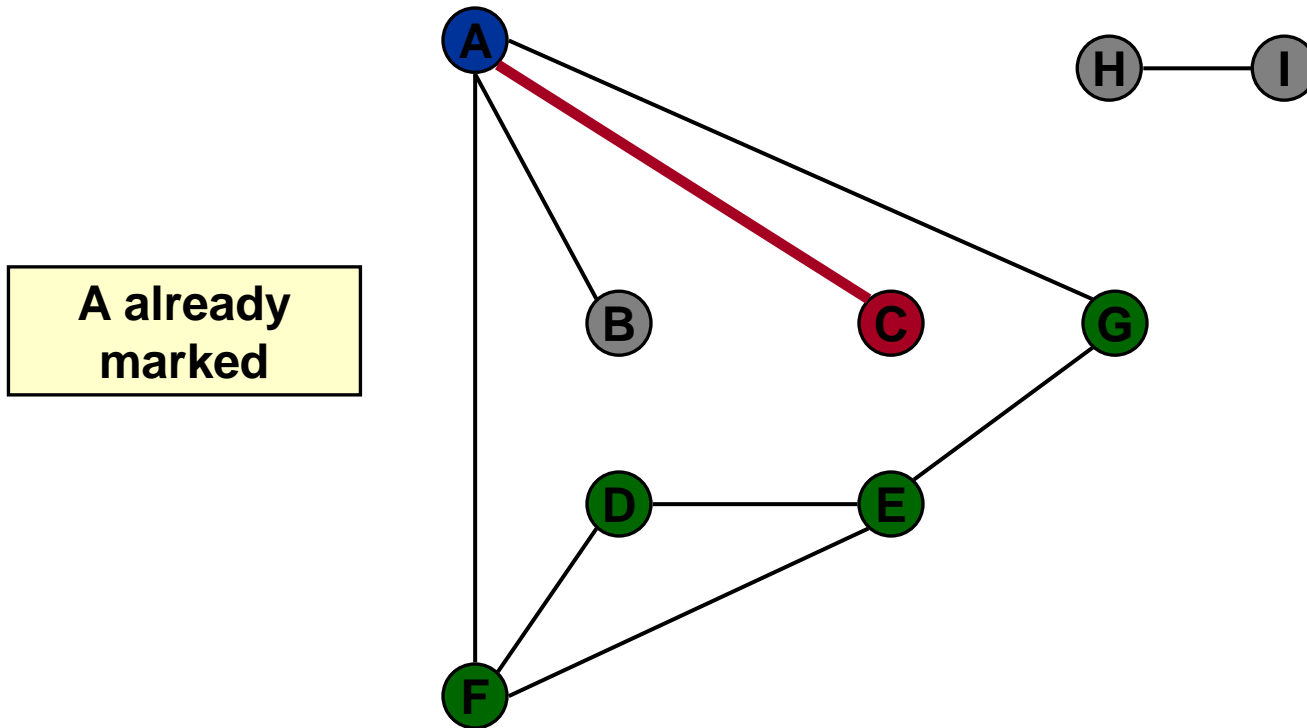


Undiscovered
Marked
Active
Finished

Stack



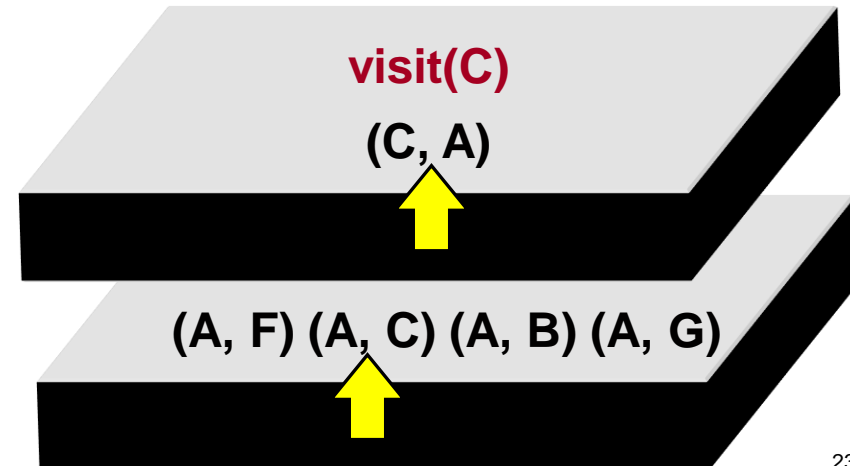
Undirected Depth First Search



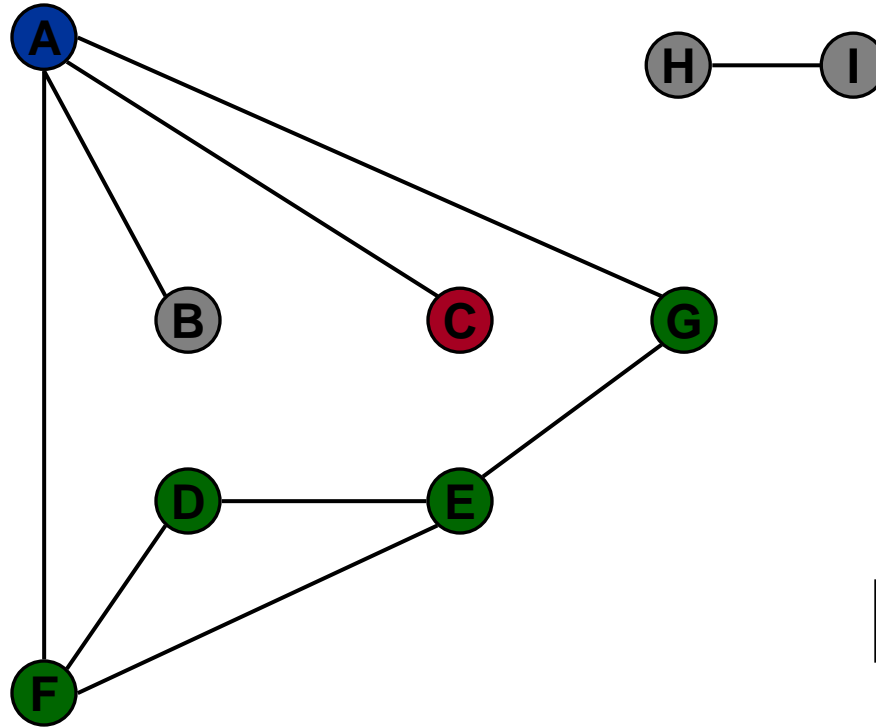
A already
marked

Undiscovered
Marked
Active
Finished

Stack

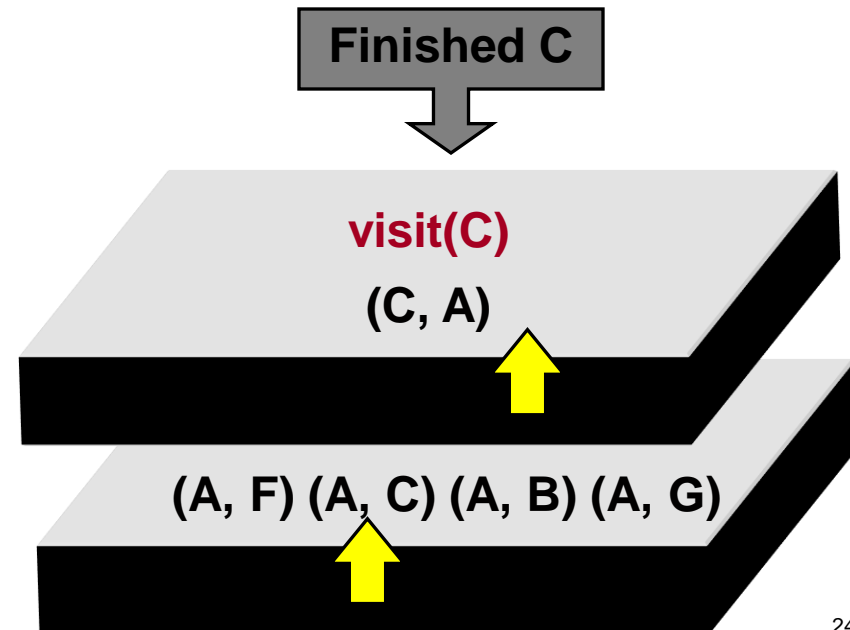


Undirected Depth First Search

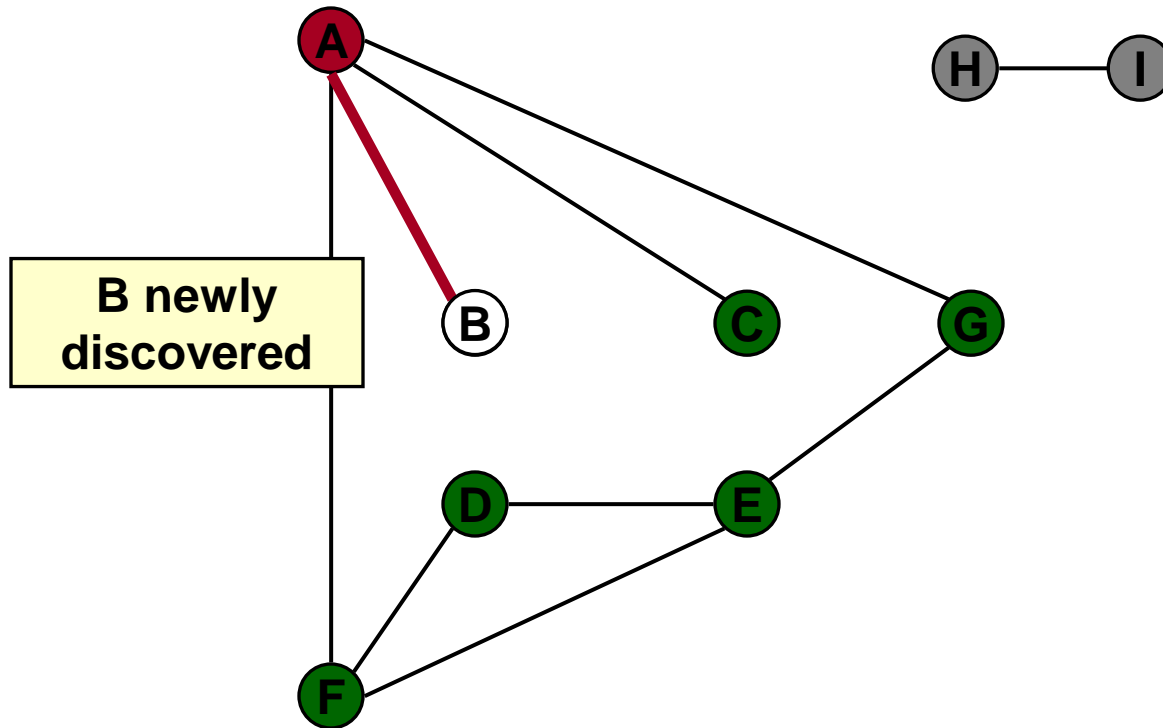


Undiscovered
Marked
Active
Finished

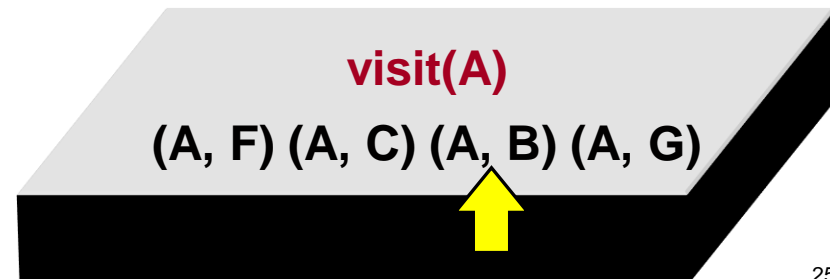
Stack



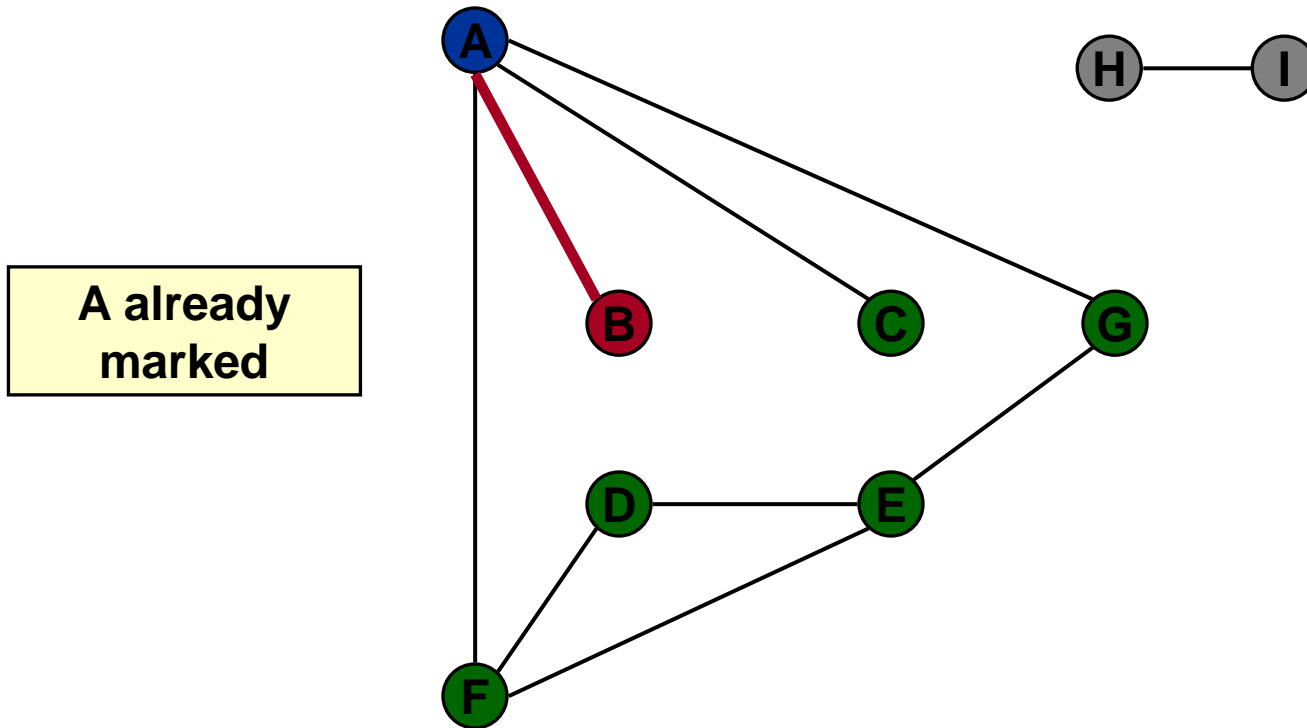
Undirected Depth First Search



Stack



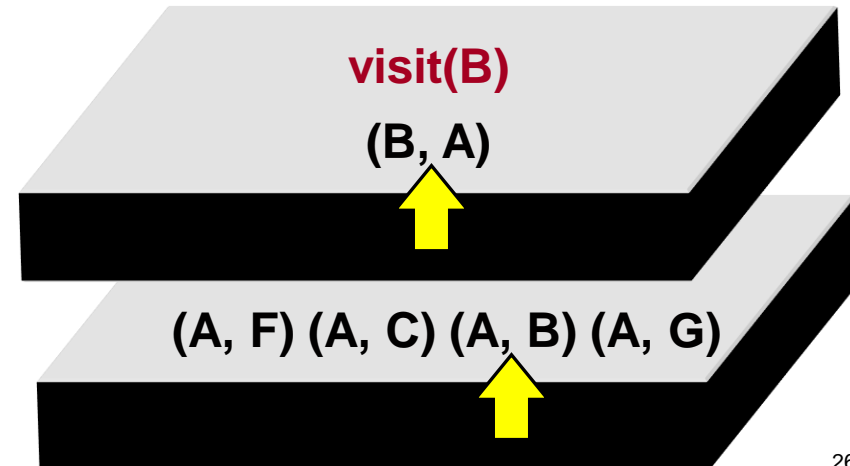
Undirected Depth First Search



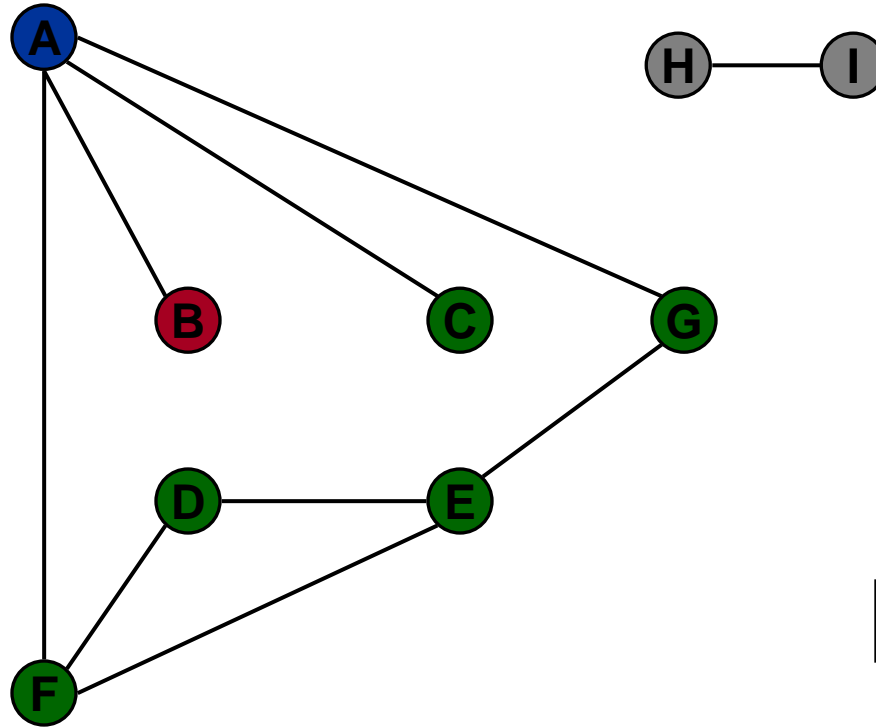
A already
marked

Undiscovered
Marked
Active
Finished

Stack

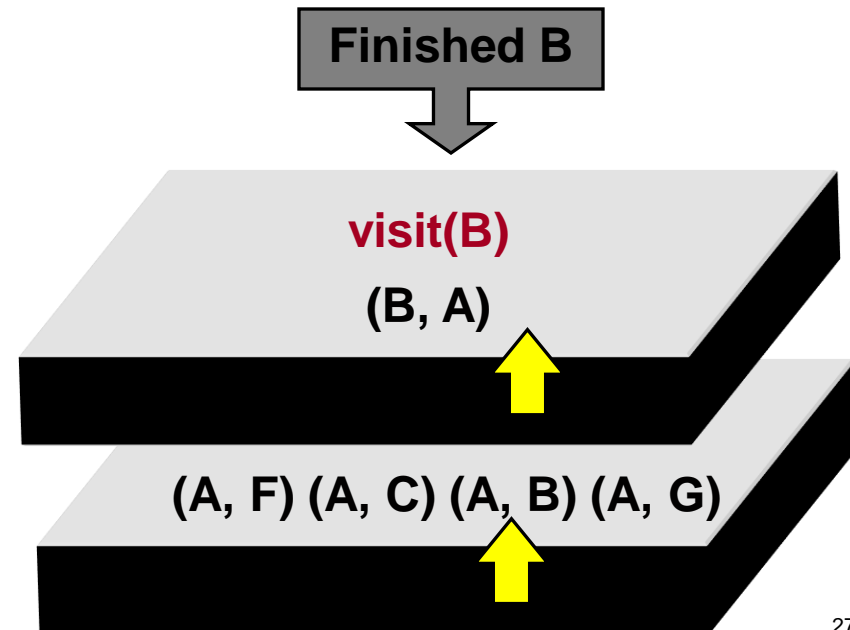


Undirected Depth First Search

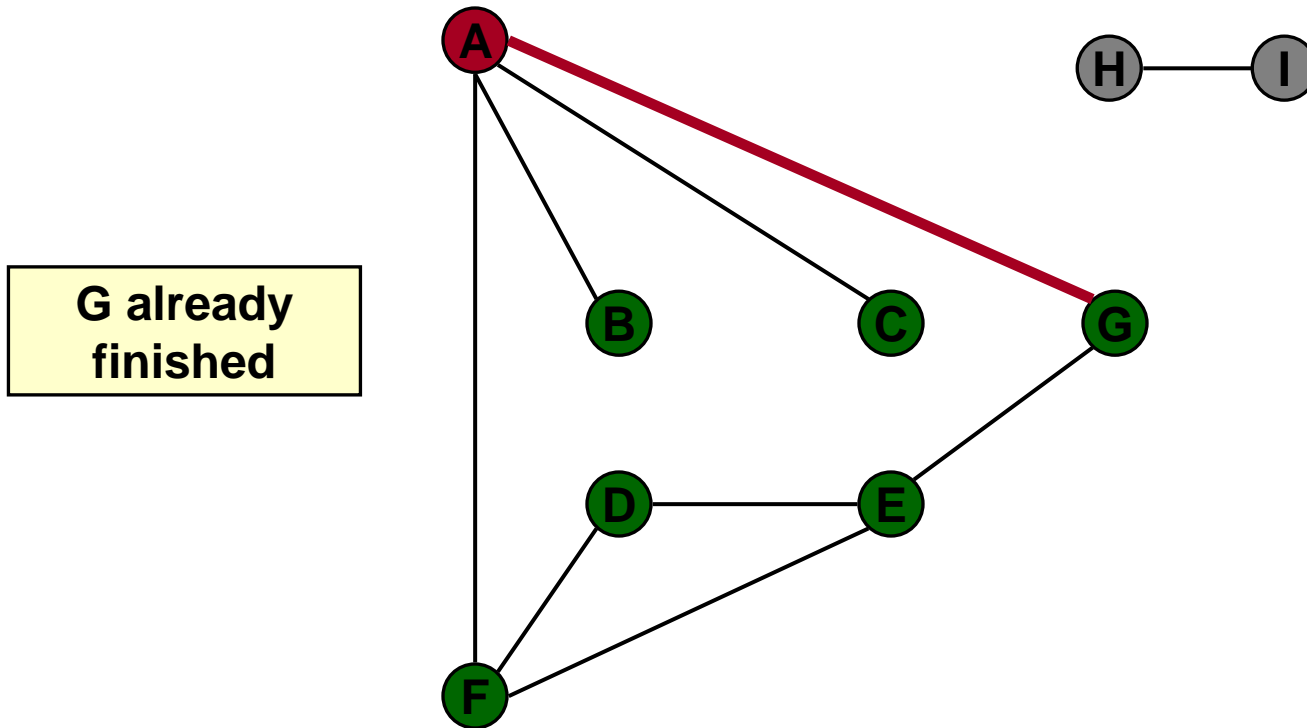


Undiscovered
Marked
Active
Finished

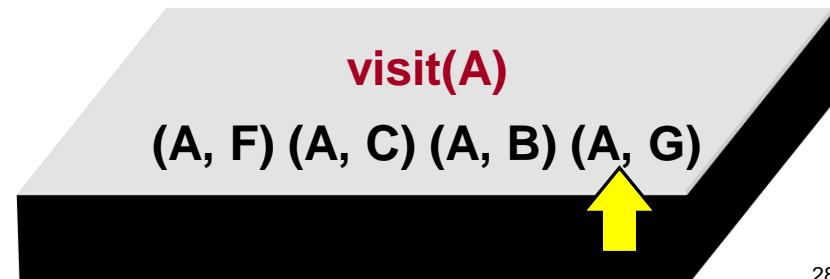
Stack



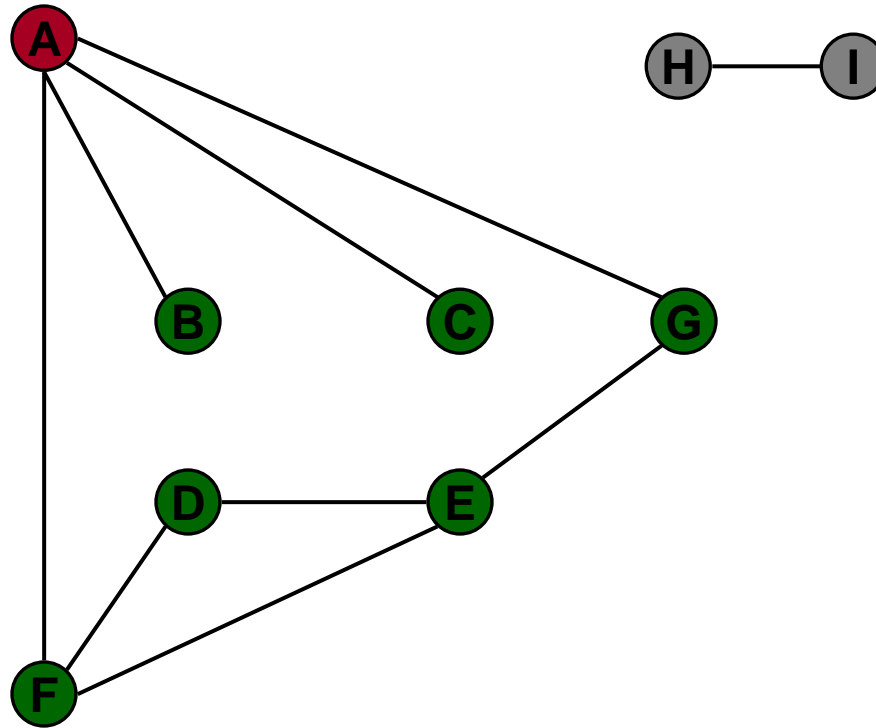
Undirected Depth First Search



Stack



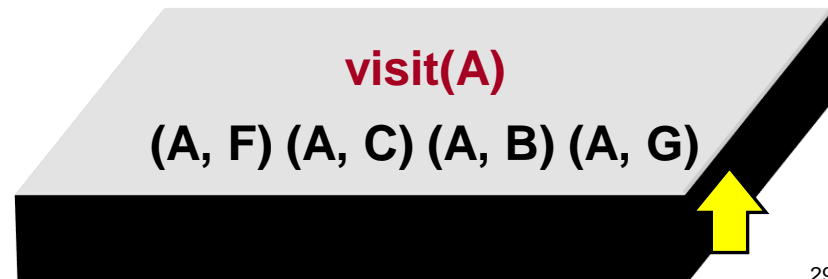
Undirected Depth First Search



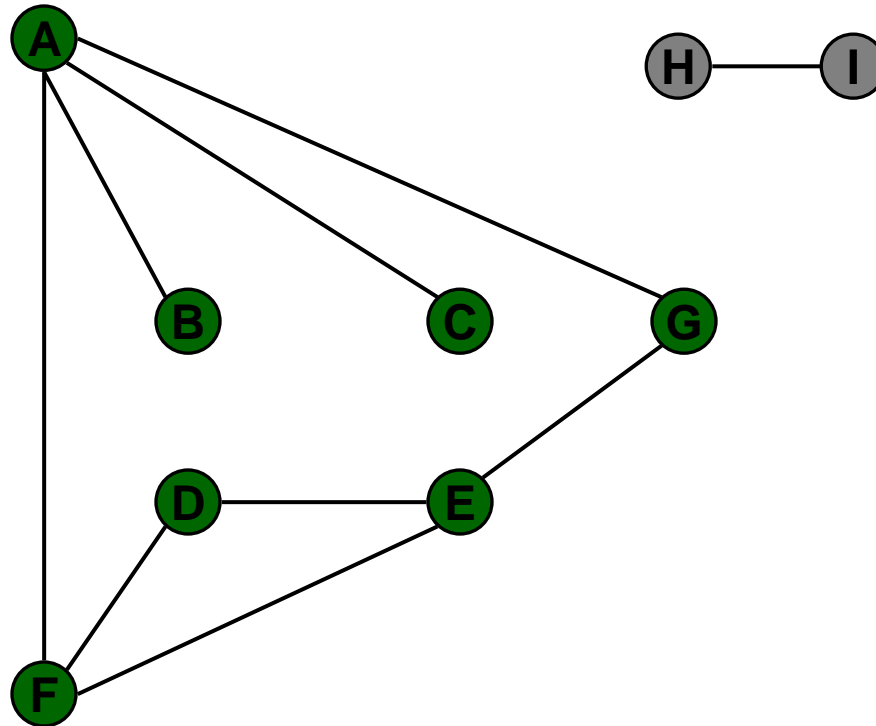
Undiscovered
Marked
Active
Finished

Stack

Finished A

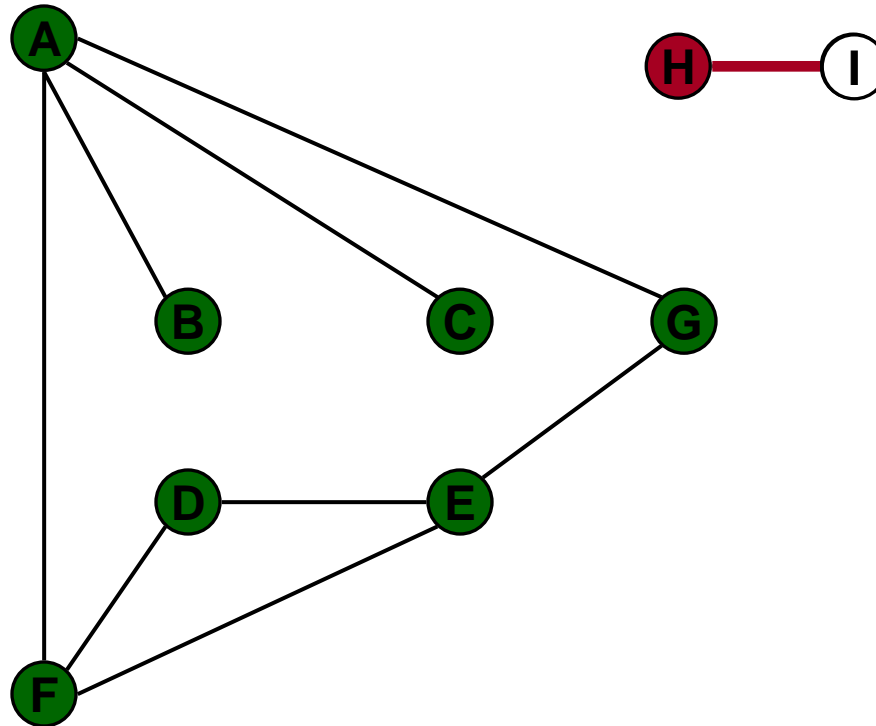


Undirected Depth First Search



Stack

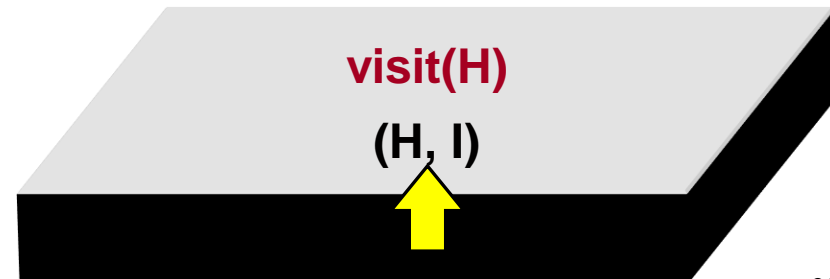
Undirected Depth First Search



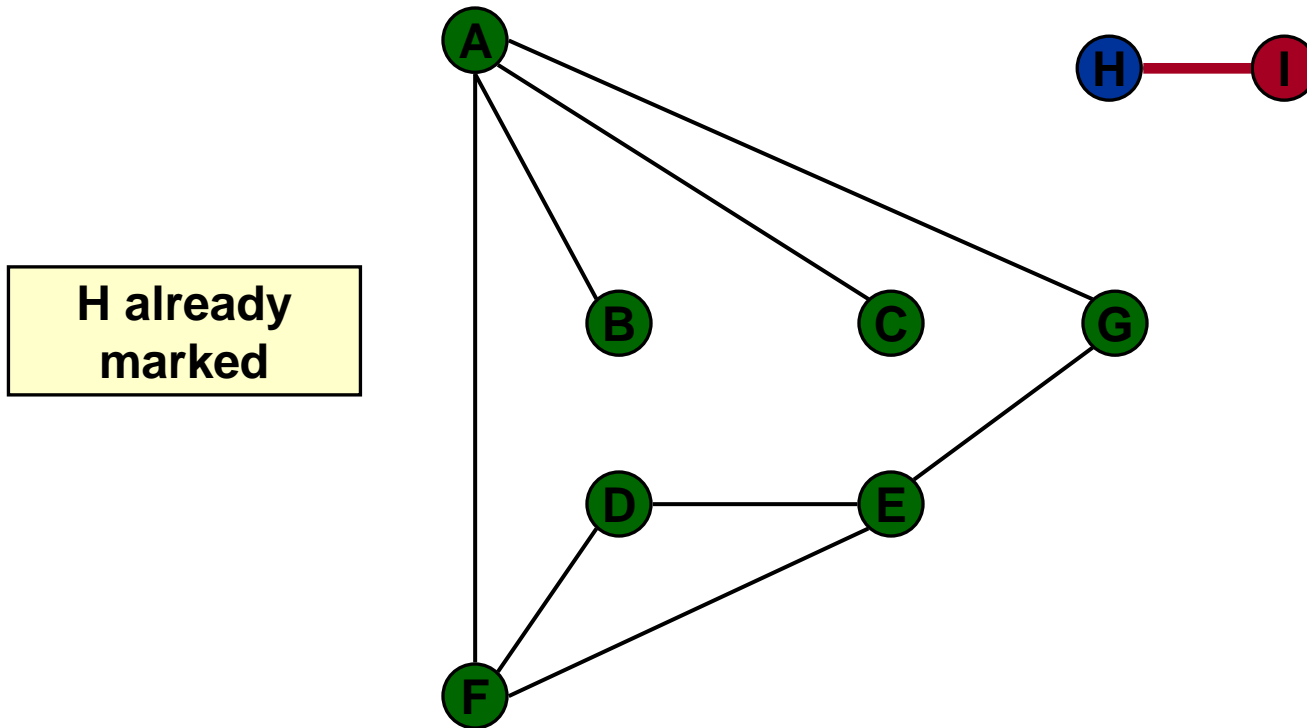
I newly
discovered



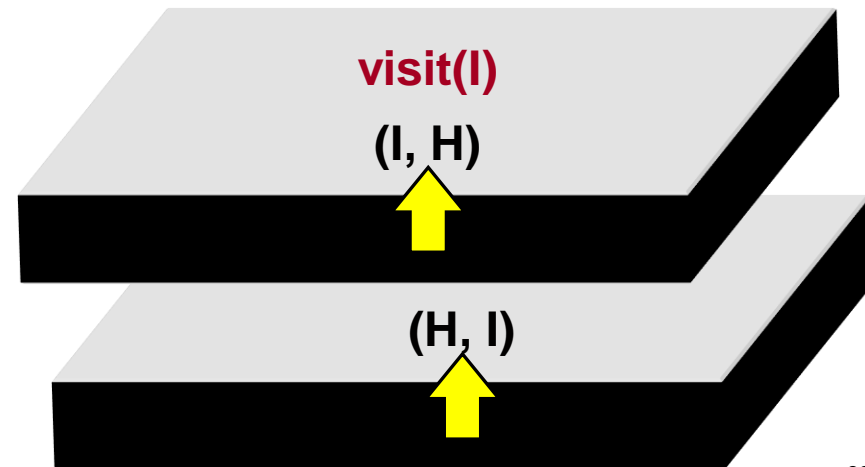
Stack



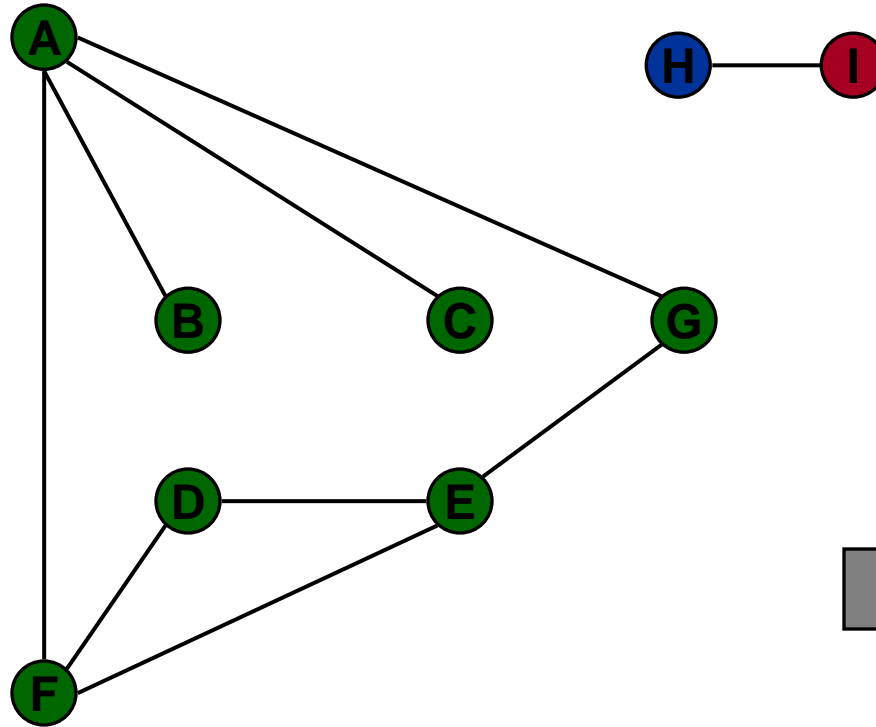
Undirected Depth First Search



Stack



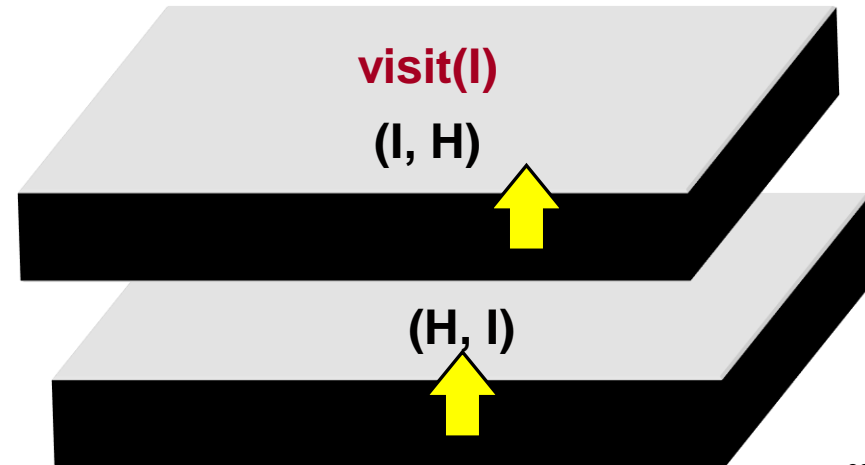
Undirected Depth First Search



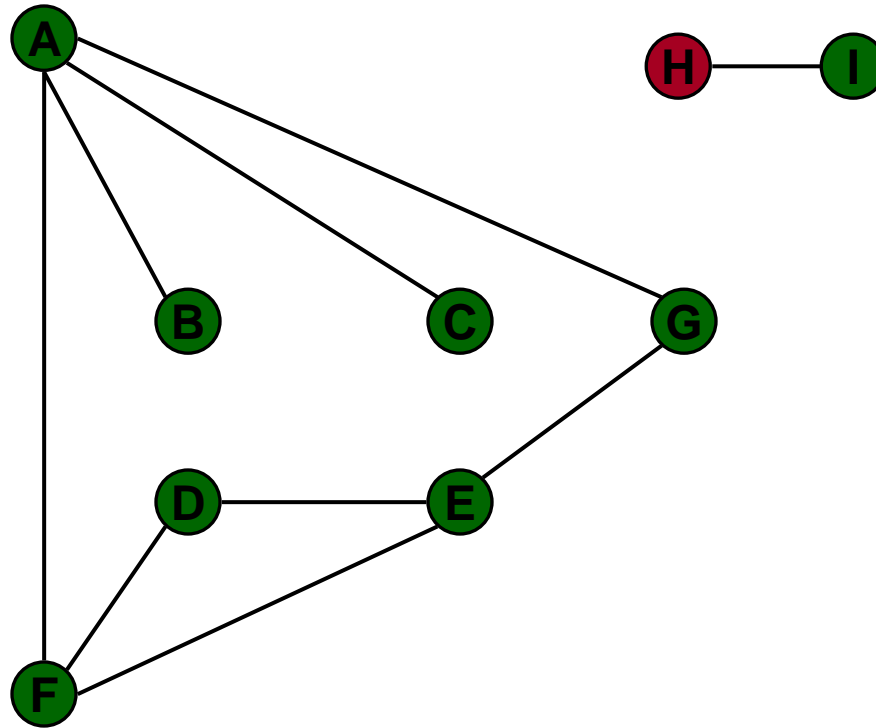
Undiscovered
Marked
Active
Finished

Finished I

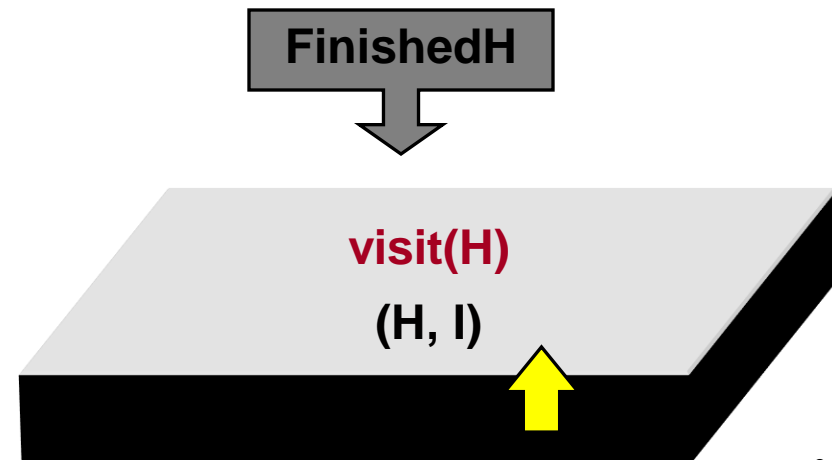
Stack



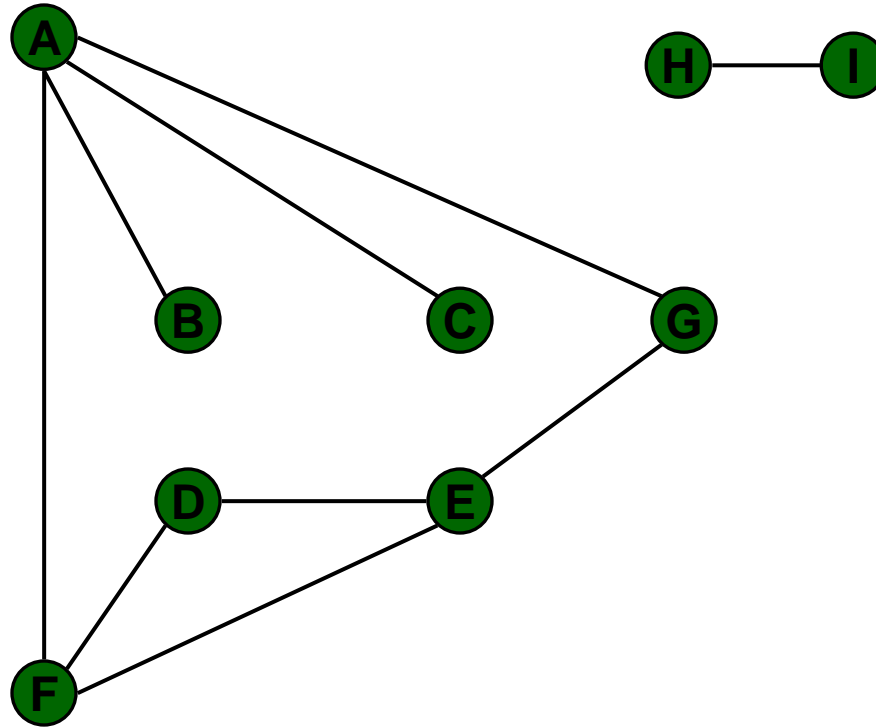
Undirected Depth First Search



Stack



Undirected Depth First Search



Stack

DFS IMPLEMENTATION

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

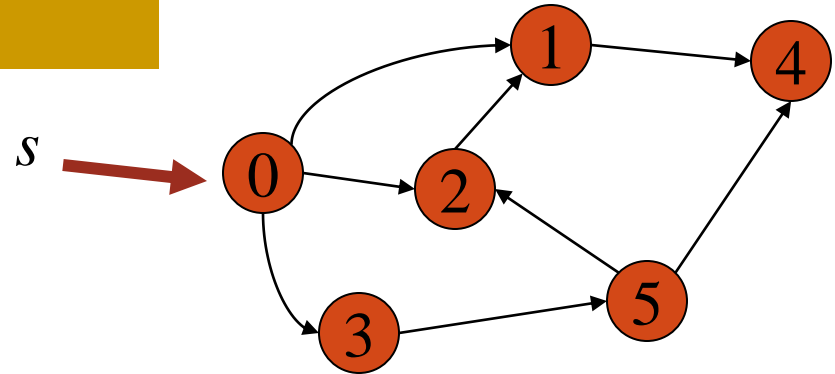
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S = \emptyset$

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

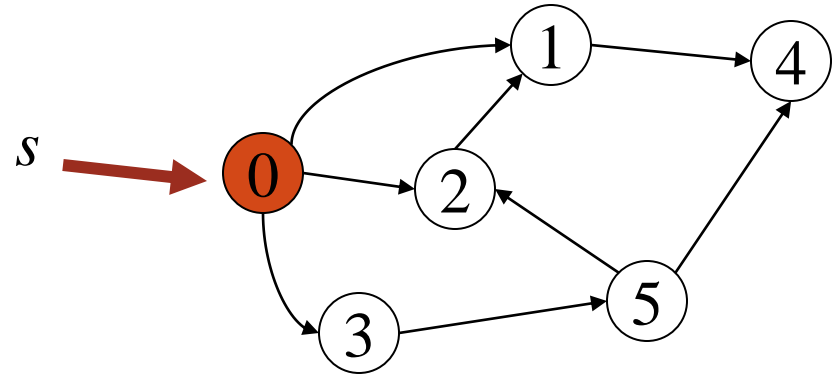
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S = \emptyset$

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

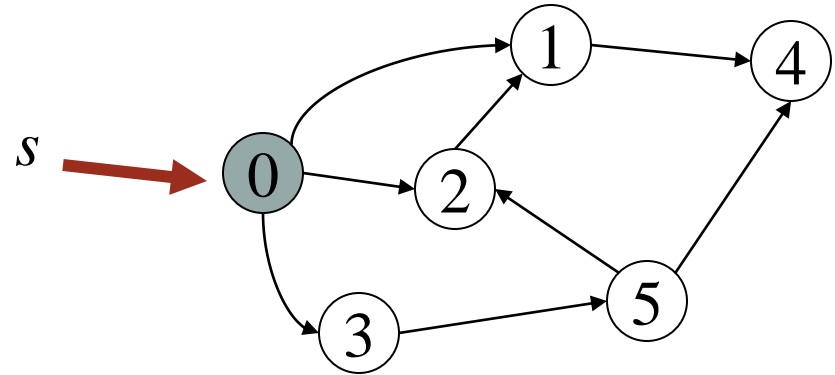
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$ 0

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

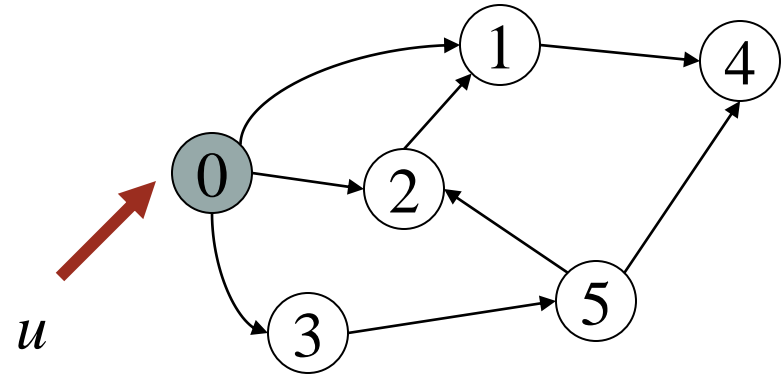
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S = \emptyset$

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

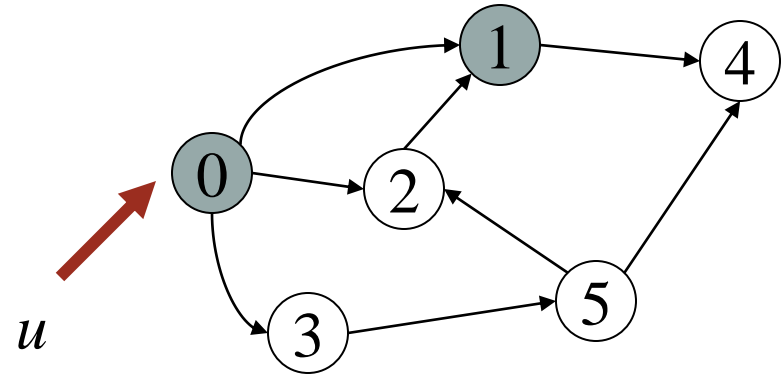
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$ 1

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

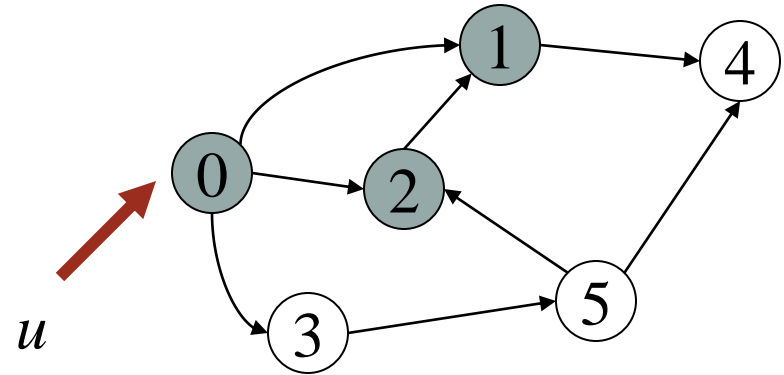
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$

2	1
---	---

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

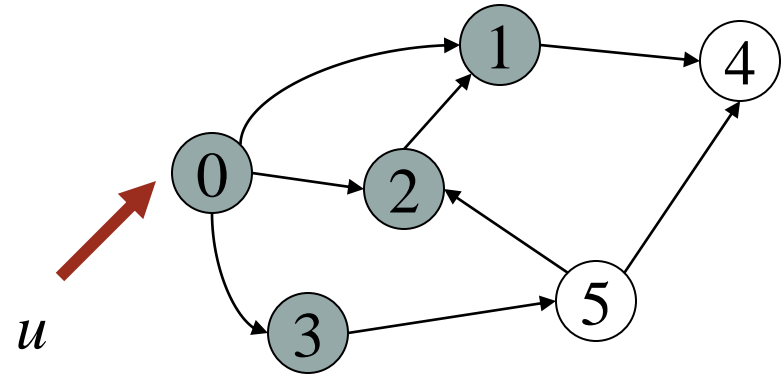
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$

3	2	1
---	---	---

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

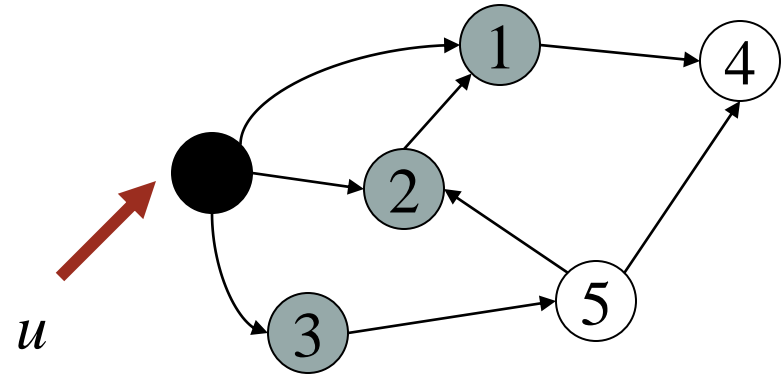
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$

3	2	1
---	---	---

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

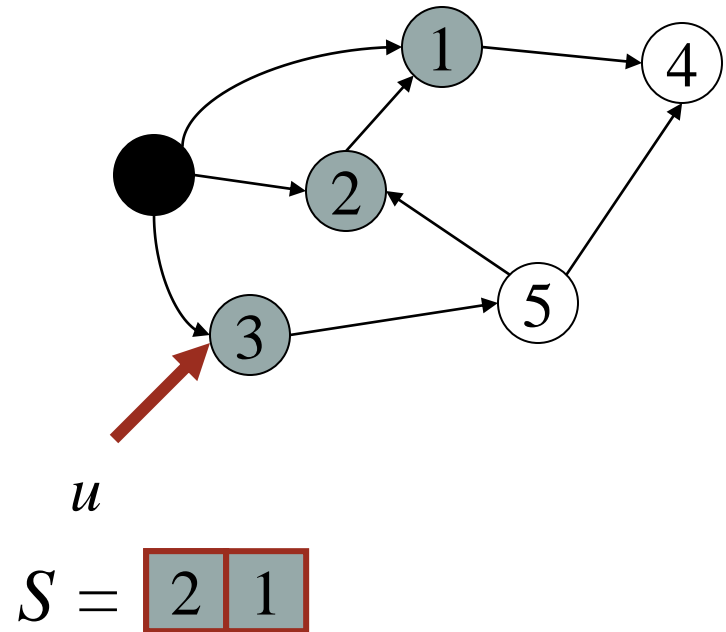
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

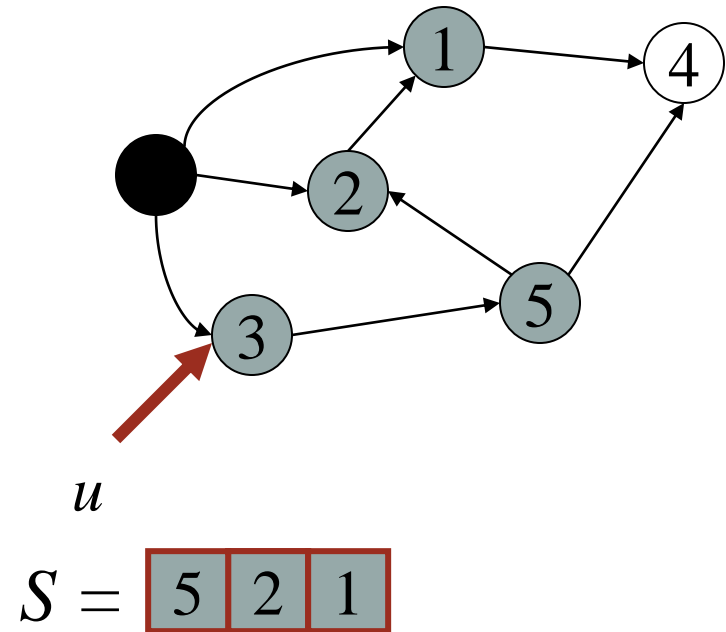
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

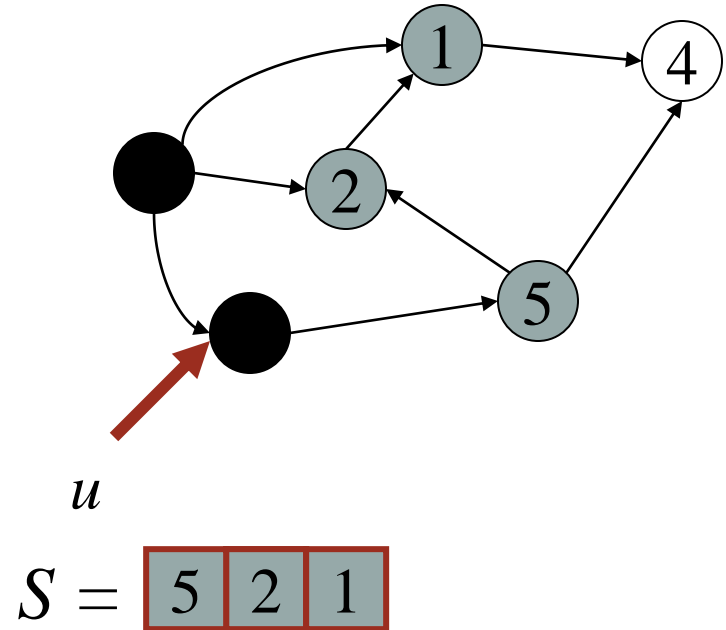
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

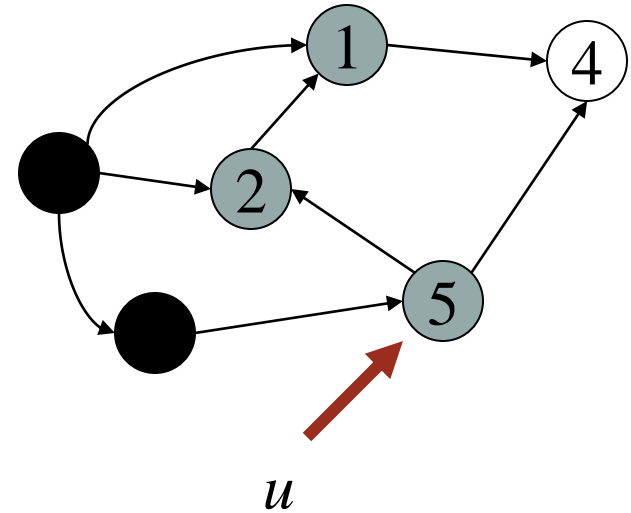
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$

2	1
---	---

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

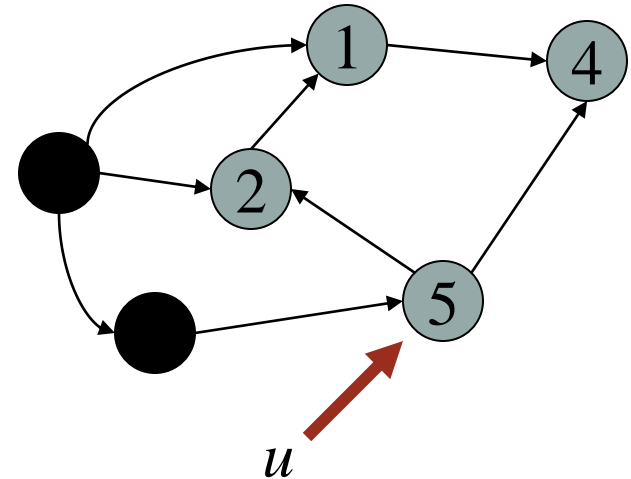
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$

4	2	1
---	---	---

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

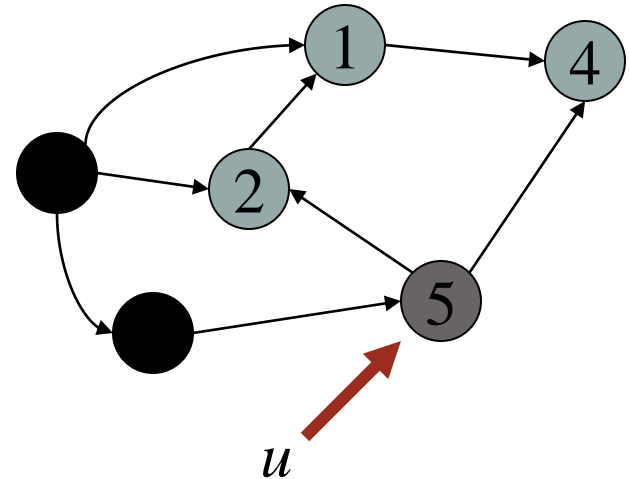
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$

4	2	1
---	---	---

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

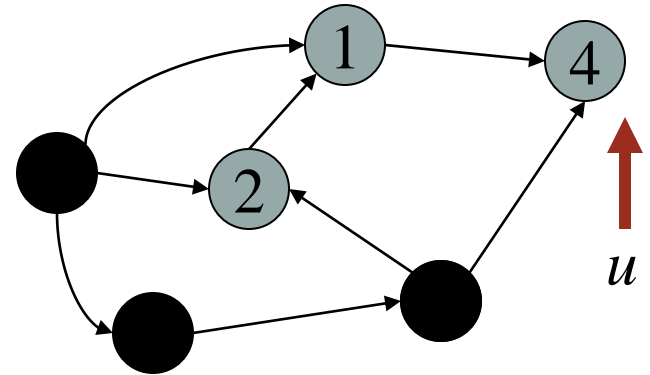
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$

2	1
---	---

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

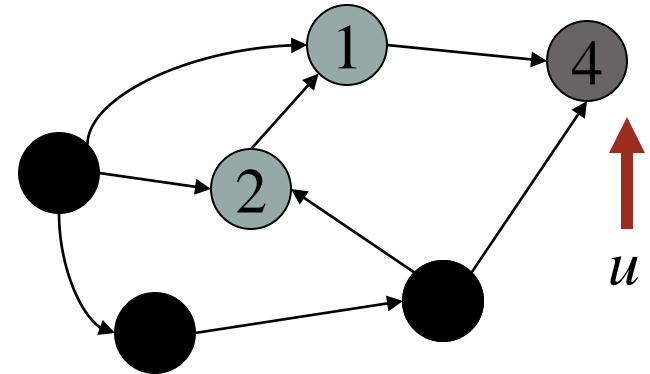
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$

2	1
---	---

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

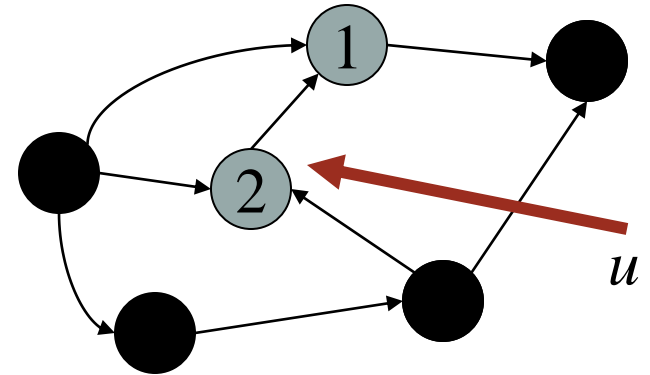
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$ 1

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

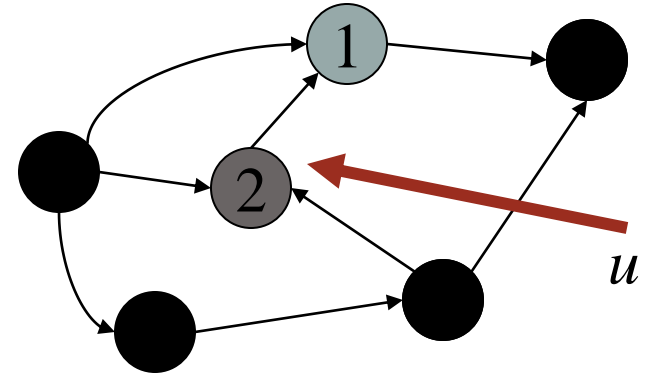
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S =$ 1

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

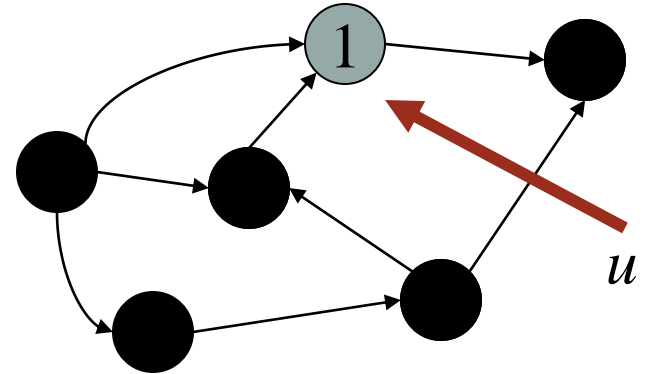
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S = \emptyset$

Depth First Search Algorithm

Given graph $G=(V,E)$ and source vertex $s \in V$

Create a stack S

For each vertex $u \in V - \{s\}$

$color[u] \leftarrow white$

$color[s] \leftarrow gray$

$S \leftarrow \{s\}$

While $S \neq \emptyset$

{

$u = \text{Pop}(S)$

for each $v \in \text{Adjacent}[u]$

if $color[v] = white$

{

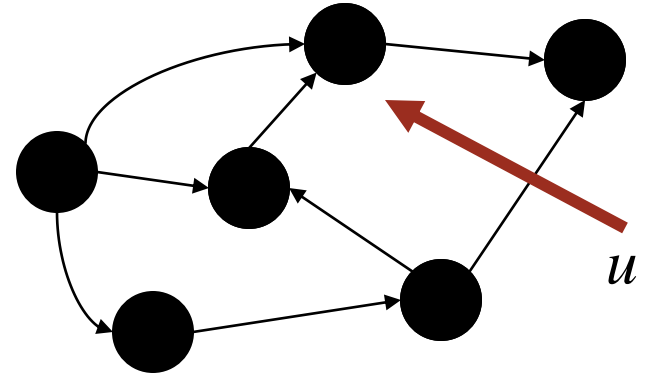
$color[v] \leftarrow gray$

Push(S, v)

}

$color[u] \leftarrow black;$

}



$S = \emptyset$

