CSE 105:
Data Structures and Algorithms-I
(Part 2)

Instructor

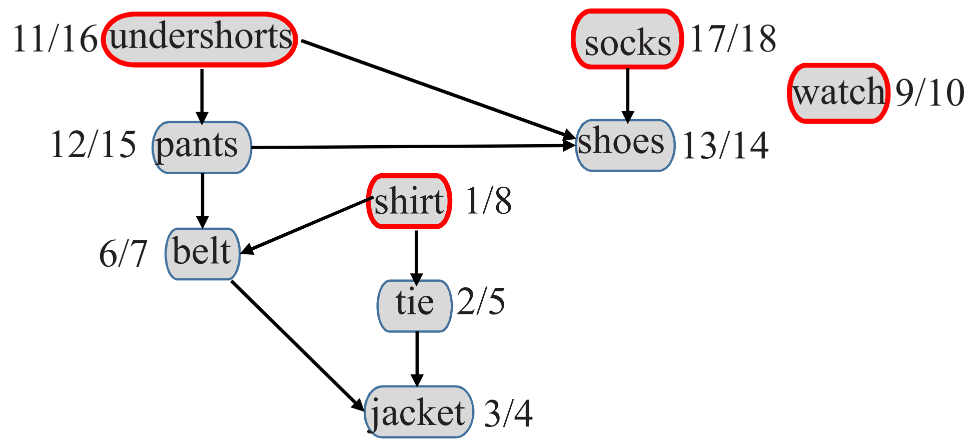
Dr Md Monirul Islam

Applications of DFS:

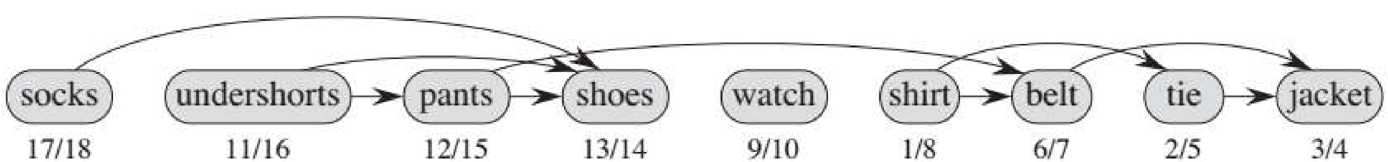
1. Topological sort of a DAG
2. Strongly Connected Components of a DiGraph

Topological sort Example: dressing of a person

Review



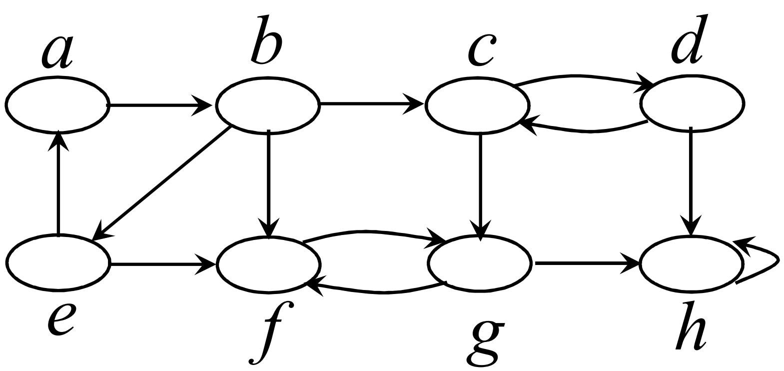
Find finishing times by DFS of the DAG



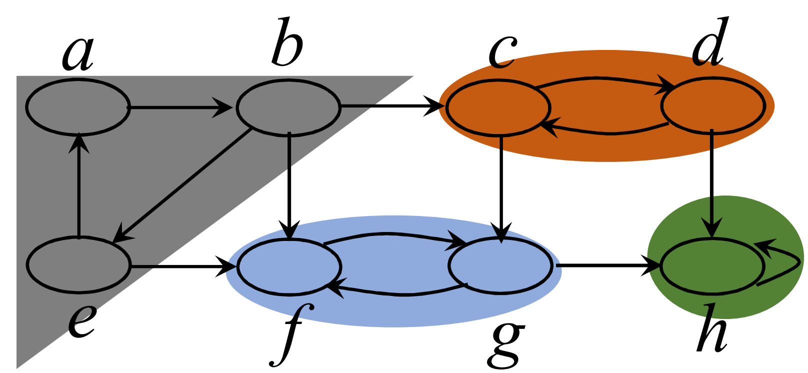
sorted by finishing times: use linked list

Strongly Connected Components

- DFS is run on a *directed graph*, $G = (V, E)$
  - decomposes G into several strongly connected components (sub-graphs)



Directed Graph

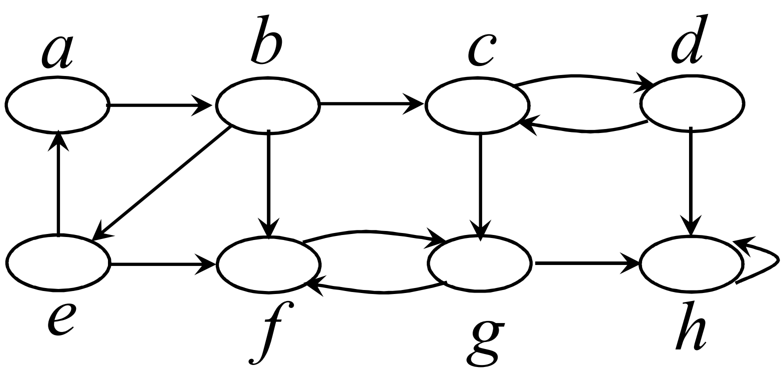


4 strongly connected components

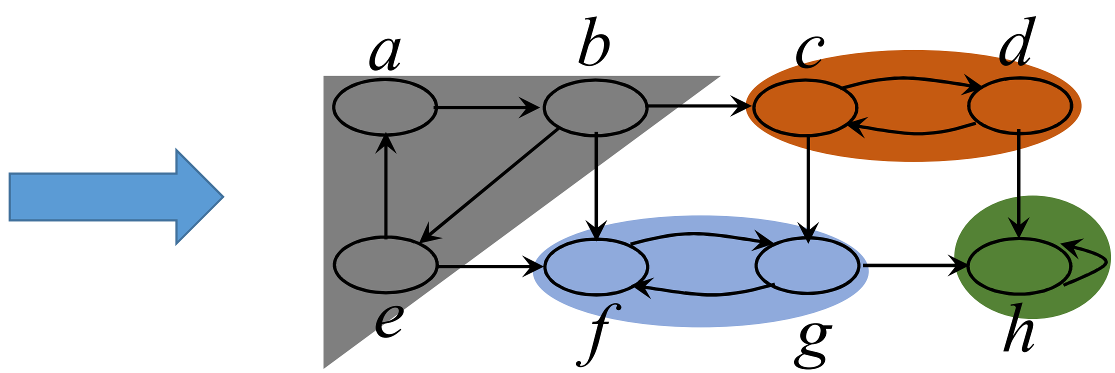
Strongly Connected Component: Definition

- An SCC is a **maximal** set of vertices $C \subseteq V$
- for any arbitrary vertex pair u and v, there are paths
  - from u to v, and
  - from v to u

There are paths: a -> b and b-> e-> b

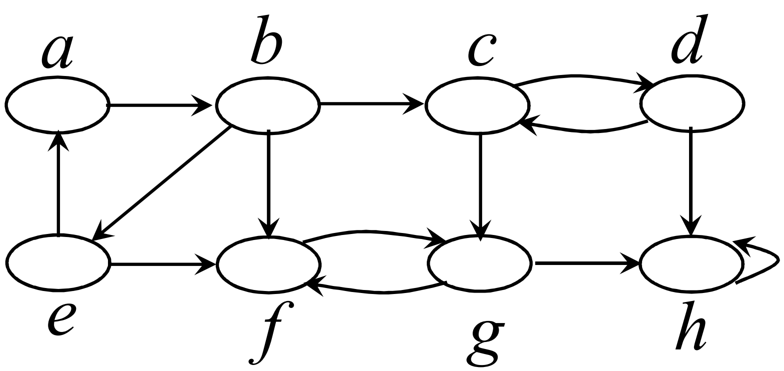


Directed Graph

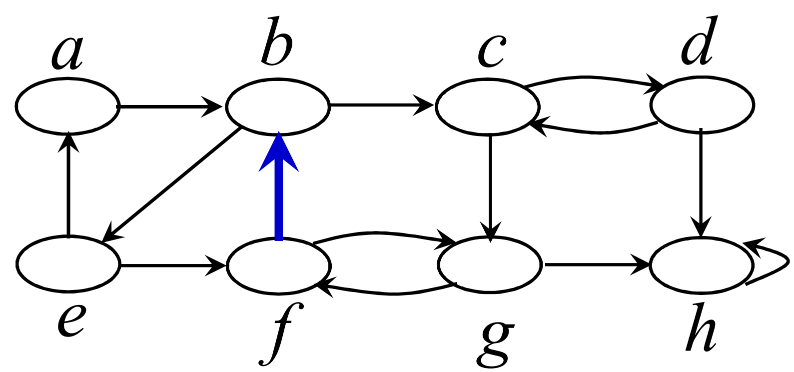


Strongly Connected Component: Definition

- An SCC is a **maximal** set of vertices $C \subseteq V$
- for any arbitrary vertex pair u and v, there are paths
  - from u to v, and
  - from v to u



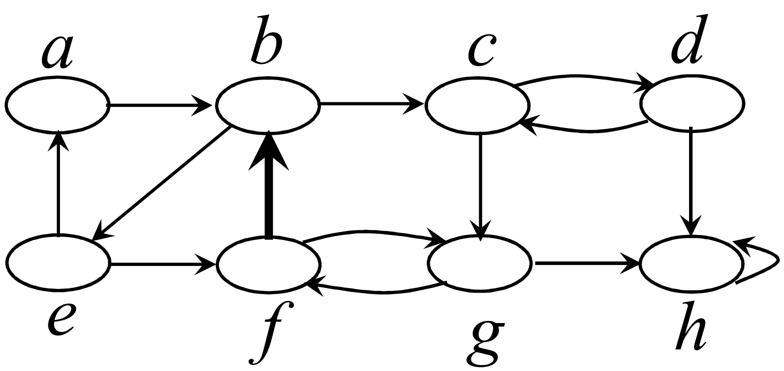
Directed Graph



Changed Graph

Strongly Connected Component: Definition

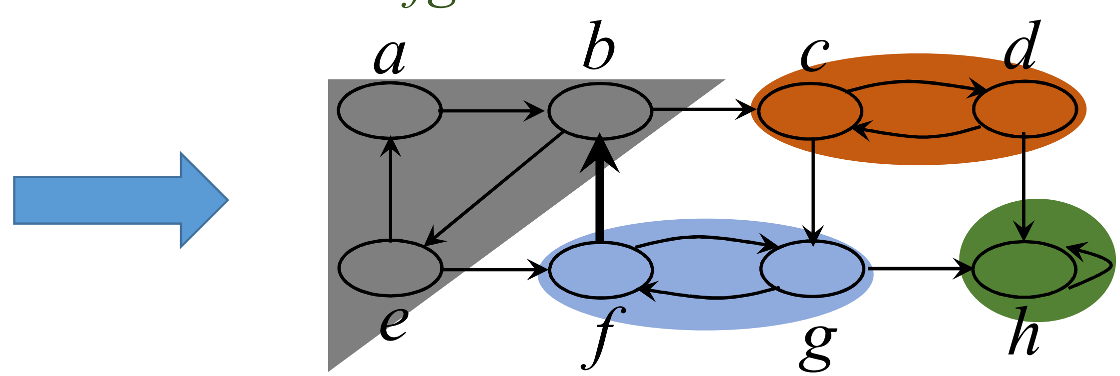
- An SCC is a **maximal** set of vertices $C \subseteq V$
- for any arbitrary vertex pair u and v, there are paths
  - from u to v, and
  - from v to u



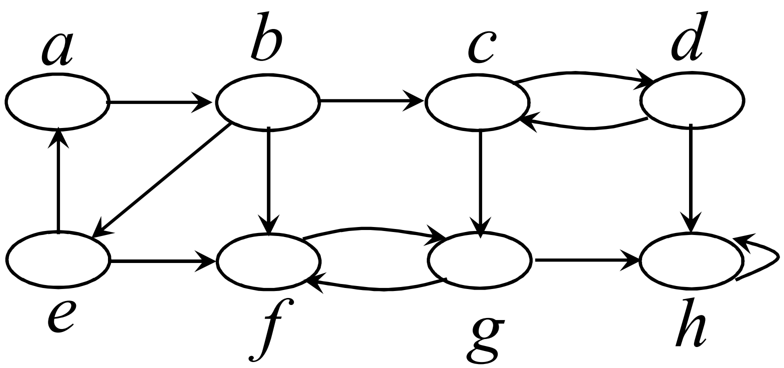
Directed Graph

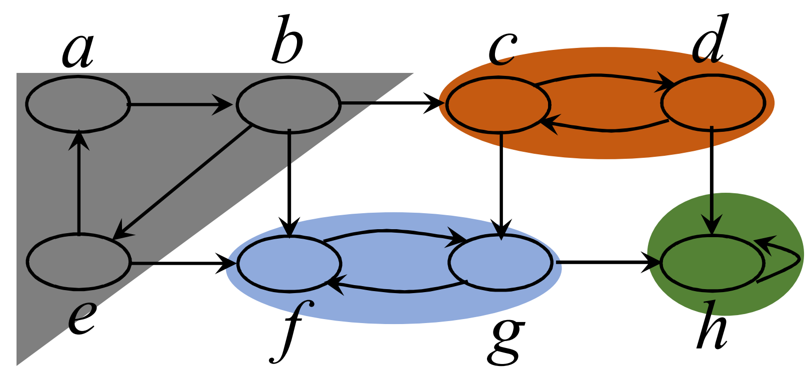
abe and *fg* are NOT maximal

abefg is maximal



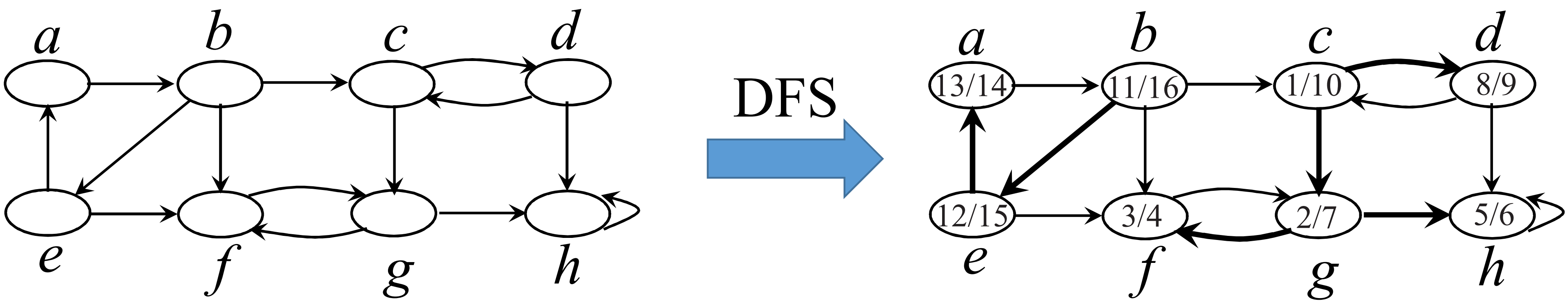
Finding Strongly Connected Components



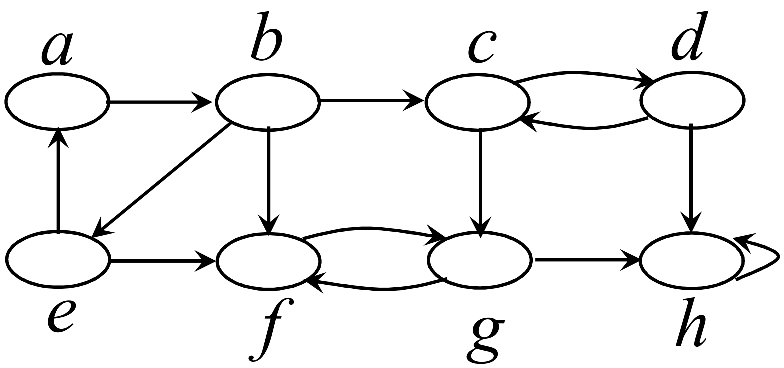


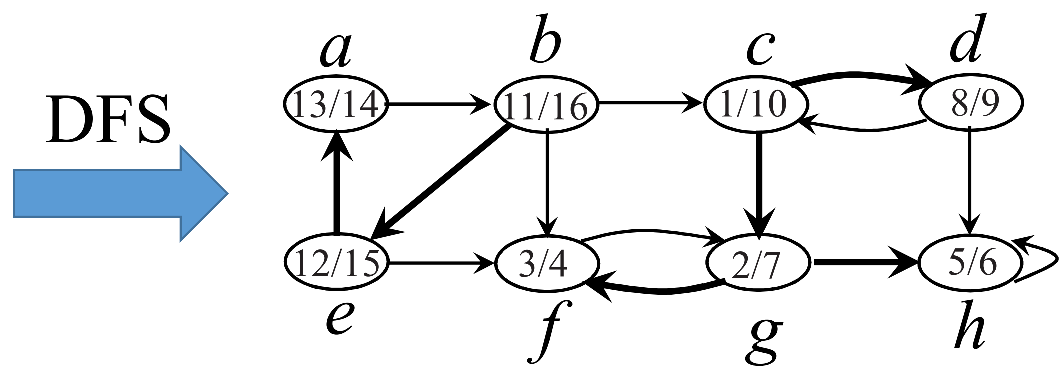
Identifies the cycles

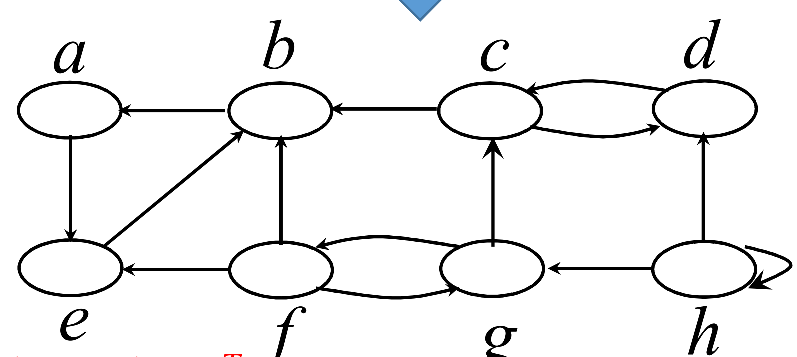
Finding Strongly Connected Components



Finding Strongly Connected Components





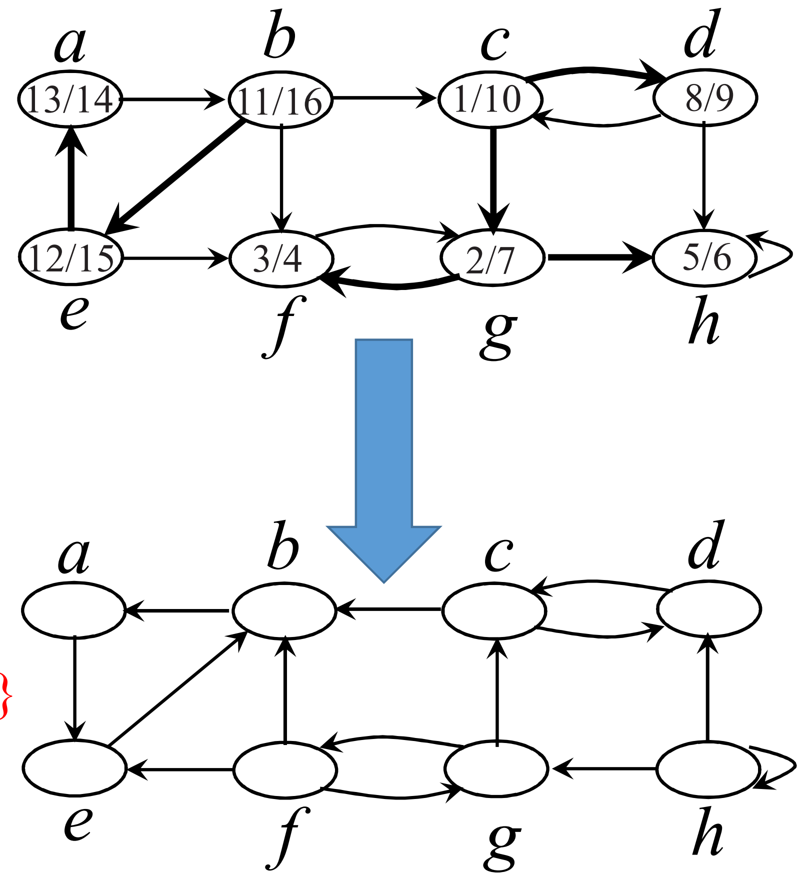


Transposed graph, G^T

Finding Strongly Connected Components

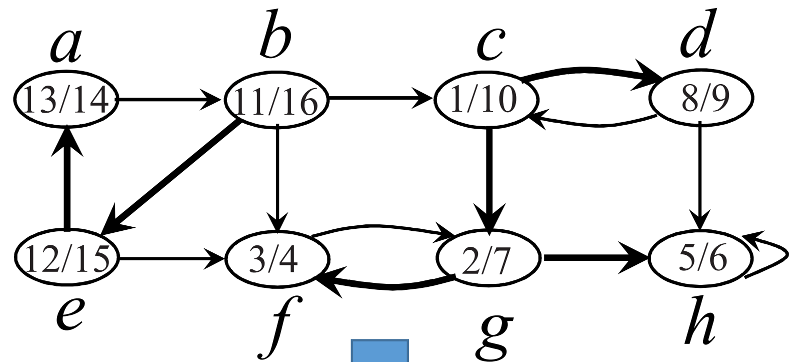
$G=(V, E)$

$G^T=(V, E^T)$, where $E^T = \{(u, v) : (v, u) \in E\}$



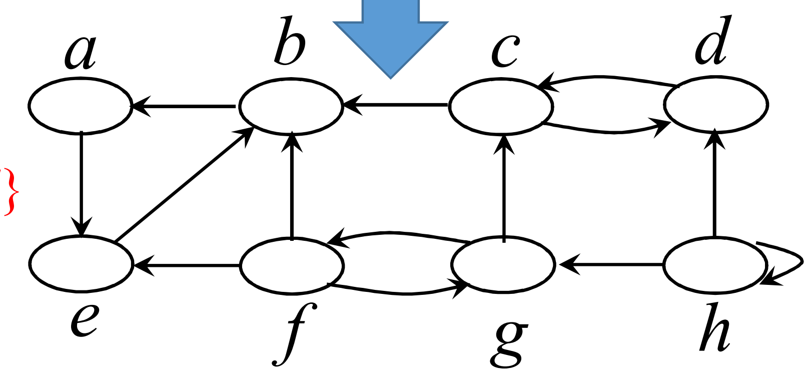
Finding Strongly Connected Components

$G=(V, E)$



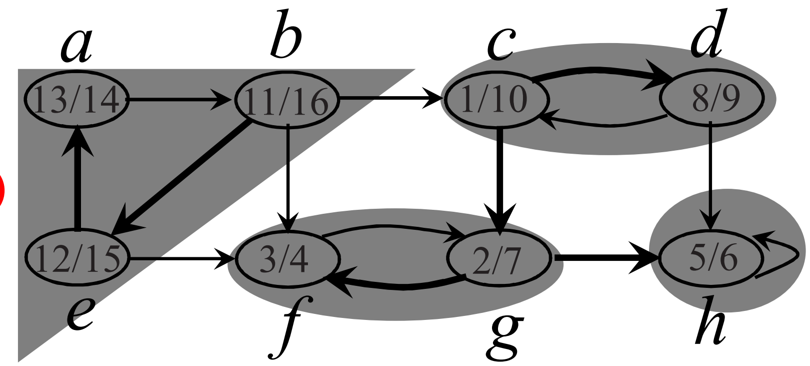
$O(V+E)$ complexity if G is in adjacency list

$G^T=(V, E^T)$, where $E^T = \{(u, v) : (v, u) \in E\}$



Finding Strongly Connected Components

$G=(V, E)$

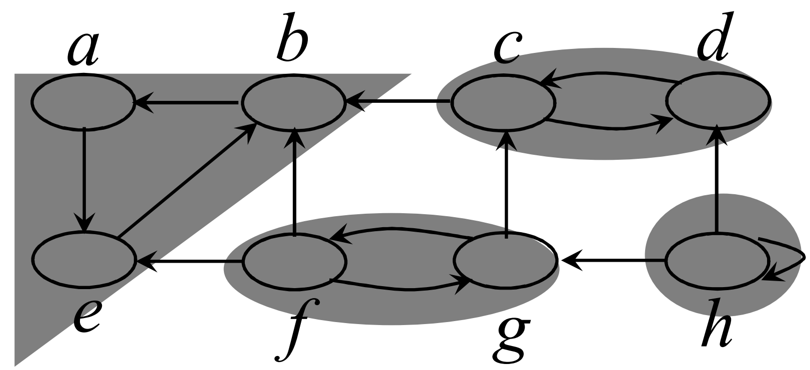


G and G^T both have exactly the same SCCs, because

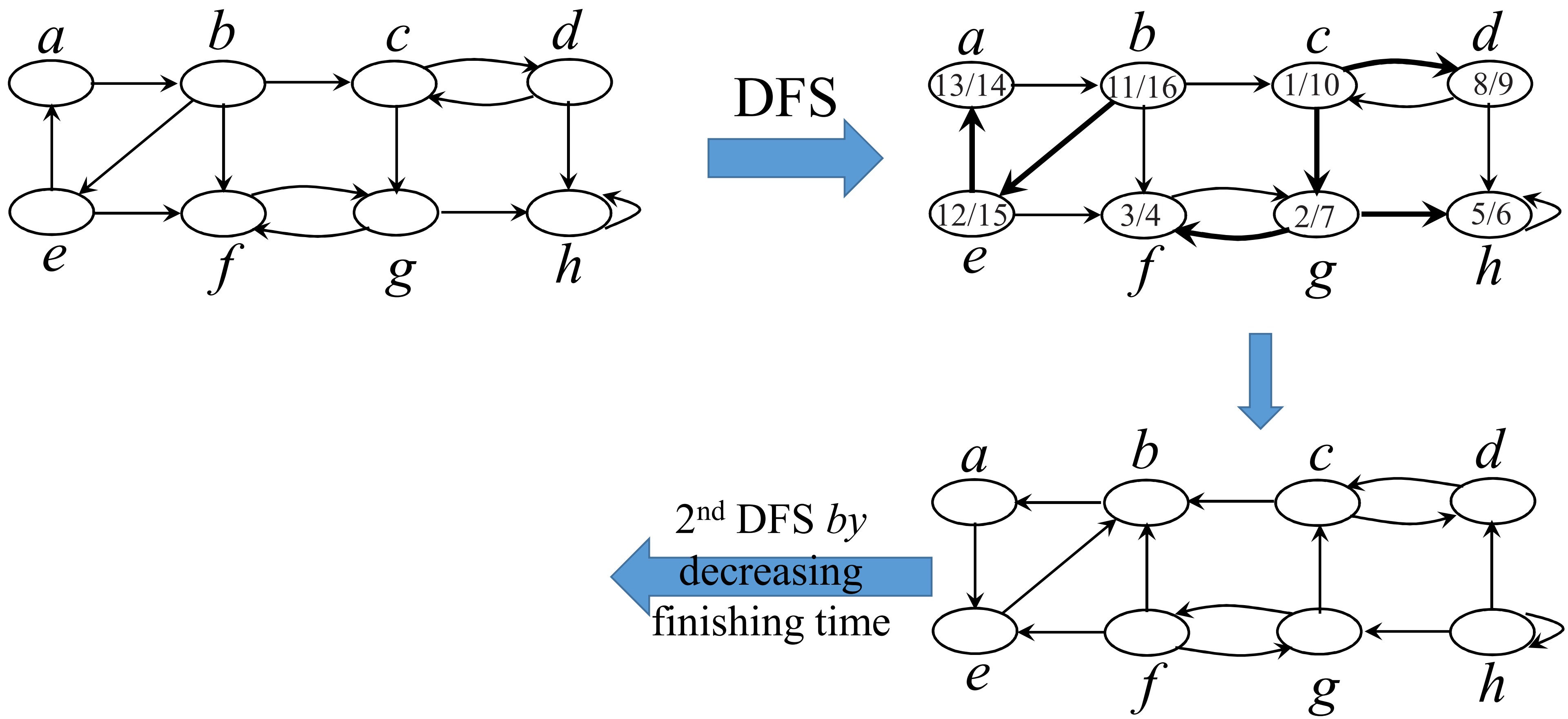
u and v are reachable from each other in G

if and only if

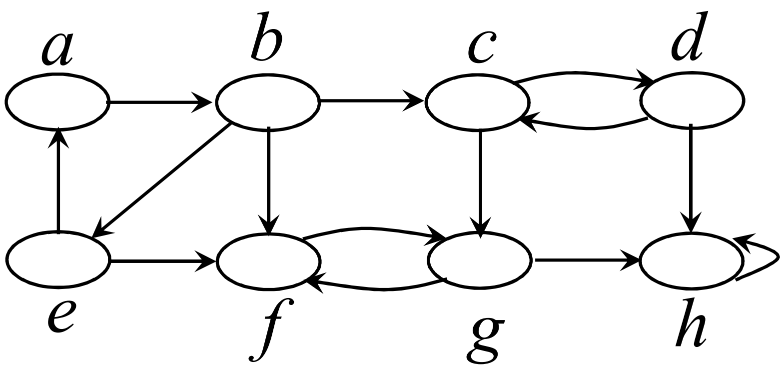
u and v are reachable from each other in G^T



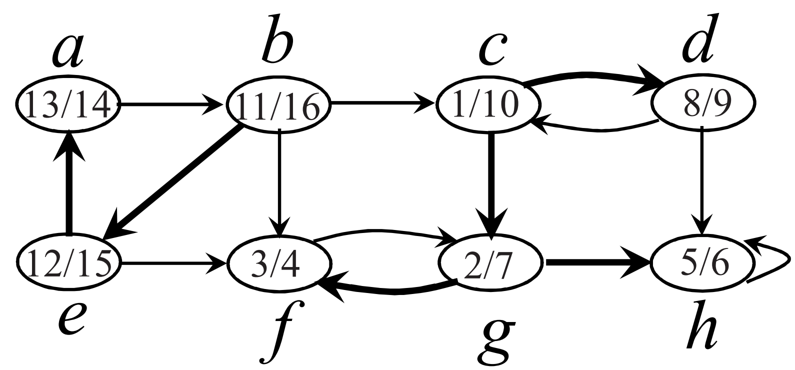
Finding Strongly Connected Components

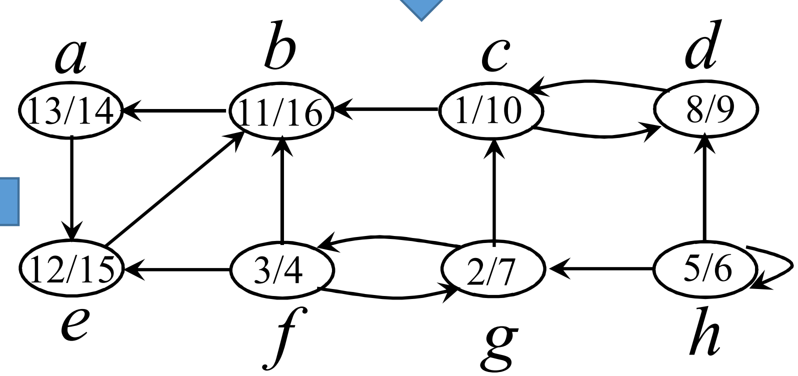


Finding Strongly Connected Components



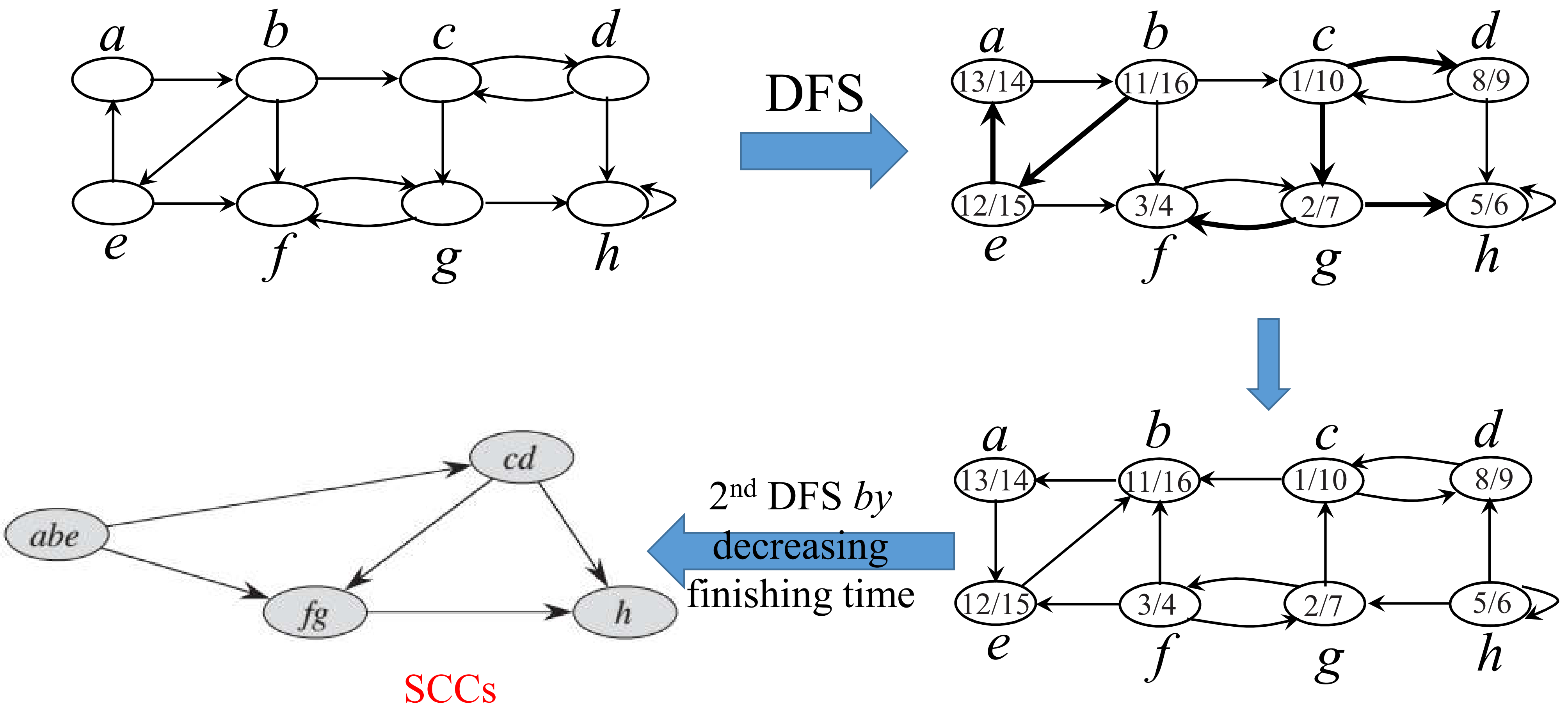
DFS





2nd DFS *by* decreasing finishing time

Finding Strongly Connected Components



Algorithm to find SCCs

STRONGLY-CONNECTED-COMPONENTS (G)

1. call DFS (G) to compute finishing times $u.f$ for each vertex u

2. compute G^T

3. call DFS (G^T), but in the main loop of DFS, consider the vertices in order of decreasing $u.f$ (as computed in line 1)

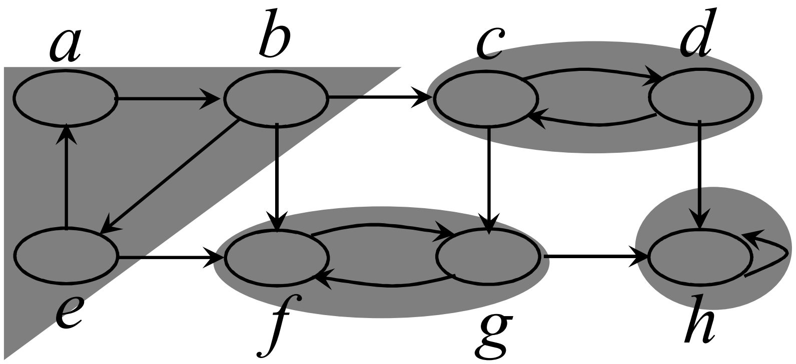
4. output the vertices of each tree in the depth-first forest formed in line 3 as a separate SCC

Algorithm to find SCCs: Complexity

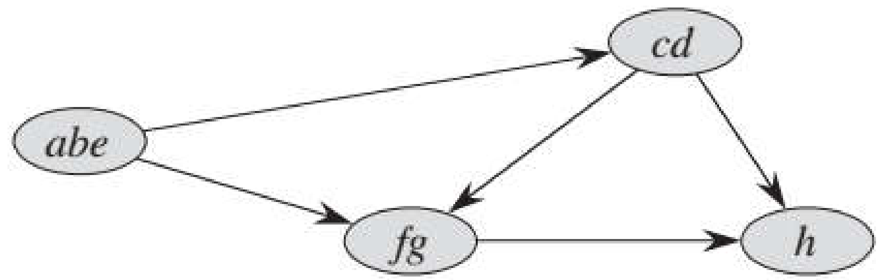
STRONGLY-CONNECTED-COMPONENTS (G)

1. call DFS (G) to compute finishing times $u.f$ for each vertex u ⟹ $O(E+V)$
2. compute G^T ⟹ $O(E+V)$
3. call DFS (G^T), but in the main loop of DFS, consider the vertices in order of decreasing $u.f$ (as computed in line 1) ⟹ $O(E+V)$
4. output the vertices of each tree in the depth-first forest formed in line 3 as a separate SCC ⟹ $O(V)$

Finding Strongly Connected Components

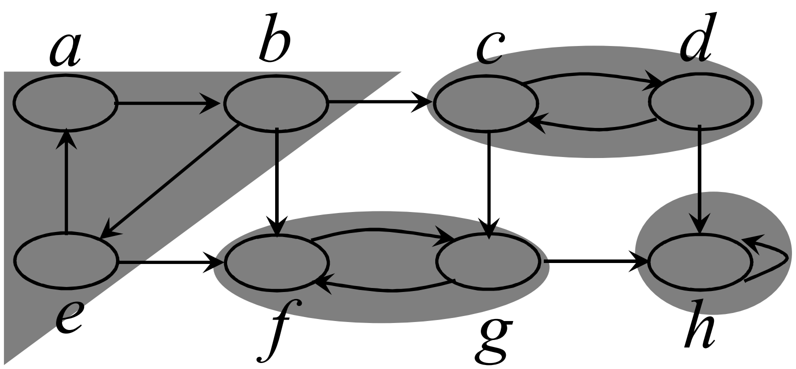


Graph, *G*

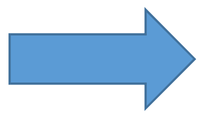


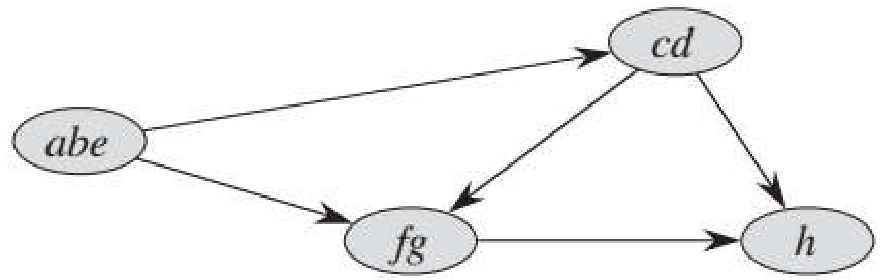
Strongly Connected Components

Finding Strongly Connected Components



Graph, G



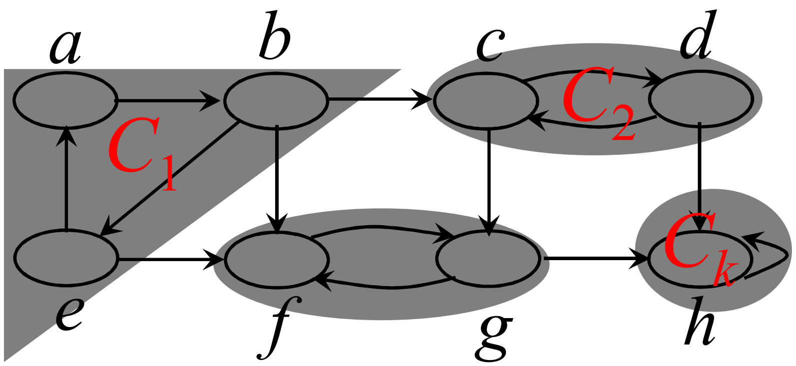


Strongly Connected Components

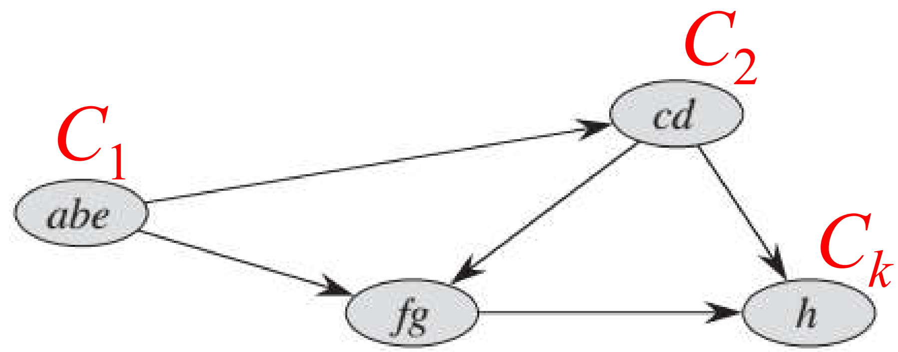
Also called Component Graph, G^{SCC}

$G^{SCC} = (\, V^{SCC}, E^{SCC})$

Finding Strongly Connected Components



Graph, G



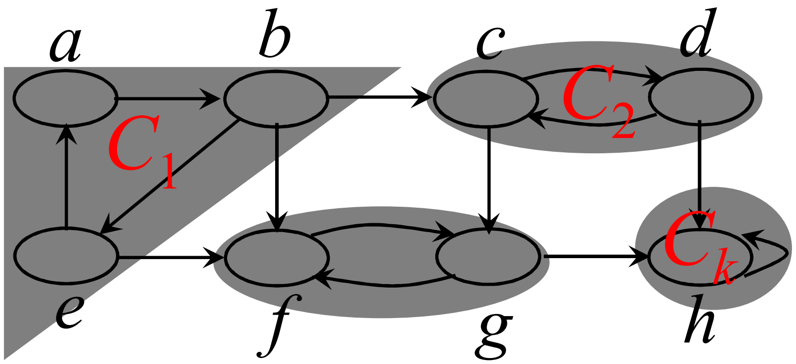
Strongly Connected Components

Also called Component Graph, G^{SCC}

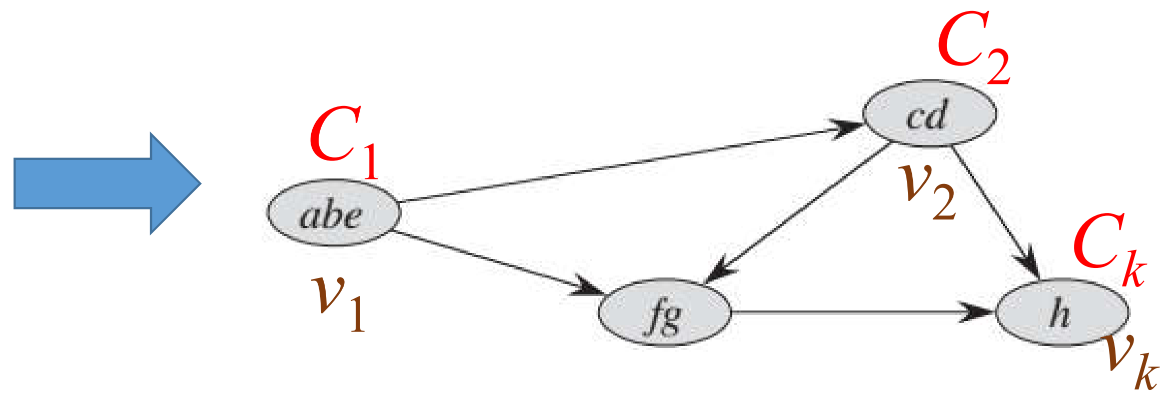
$$G^{SCC} = (\, V^{SCC}, E^{SCC})$$

Components: C_1, C_2, C_3, …,C_k

Finding Strongly Connected Components



Graph, G



Strongly Connected Components

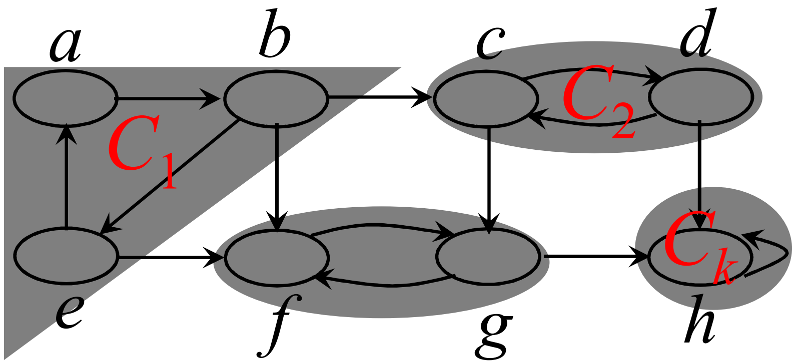
Also called Component Graph, G^{SCC}

$$G^{SCC} = (V^{SCC}, E^{SCC})$$

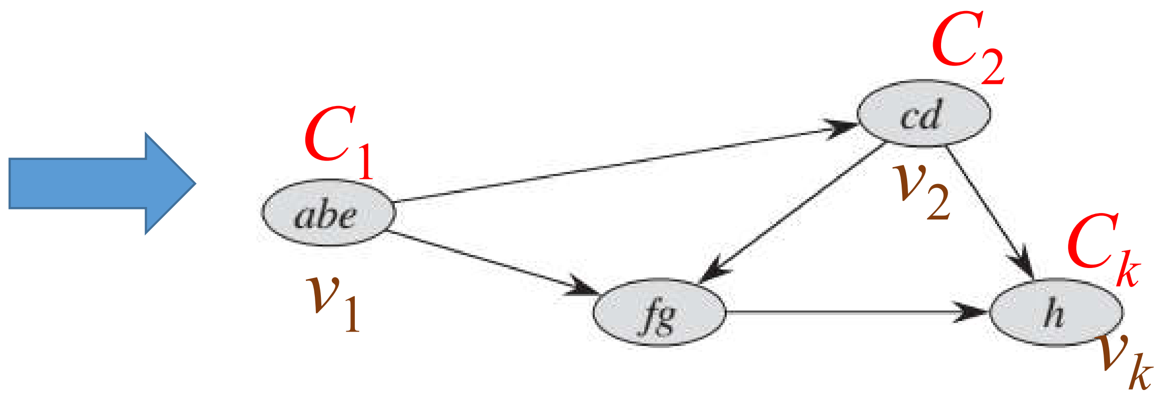
Components: $C_1, C_2, C_3, \ldots, C_k$

Vertices, $V^{SCC} = \{v_1, v_2, v_3, \ldots, v_k\}$

Finding Strongly Connected Components



Graph, G



Strongly Connected Components

Also called Component Graph, G^{SCC}

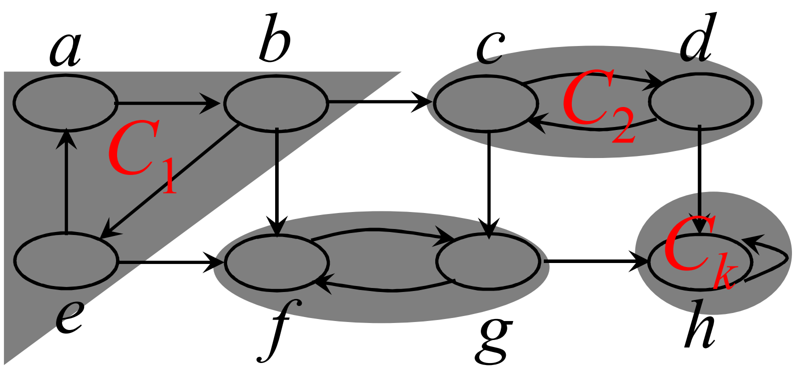
$(v_i, v_j) \in E^{SCC}$ if
$(x, y) \in G$ such that $x \in C_i$ and $y \in C_j$

$G^{SCC} = (V^{SCC}, E^{SCC})$

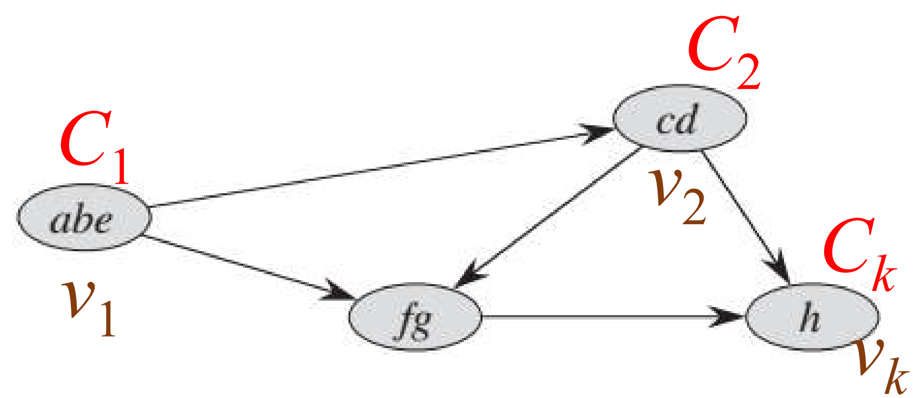
Components: $C_1, C_2, C_3, \ldots, C_k$

Vertices, $V^{SCC} = \{v_1, v_2, v_3, \ldots, v_k\}$

Finding Strongly Connected Components



Graph, G



Strongly Connected Components

Also called Component Graph, G^{SCC}

G^{SCC} is a DAG

$$G^{SCC} = (\, V^{SCC}, E^{SCC})$$

Components: $C_1, C_2, C_3, \ldots, C_k$

Vertices, $V^{SCC} = \{v_1, v_2, v_3, \ldots, v_k\}$

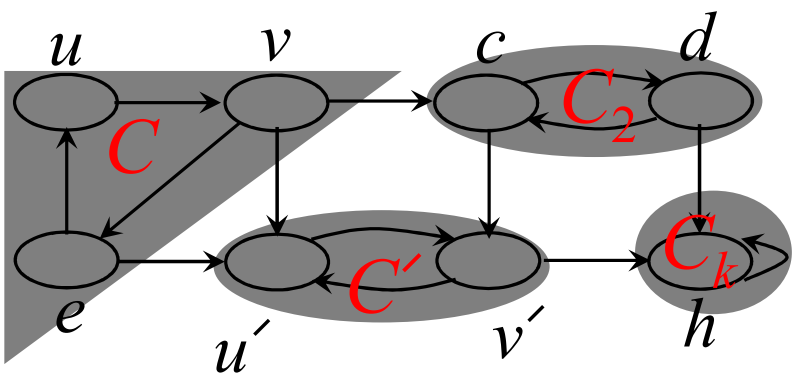
Lemma 22.13

Let C and C′ be **distinct** SCCs in directed graph $G = (V, E)$,
let $u, v \in C$, and $u', v' \in C'$ and suppose that G contains a path u to u'.

Then G cannot also contain a path v' to v

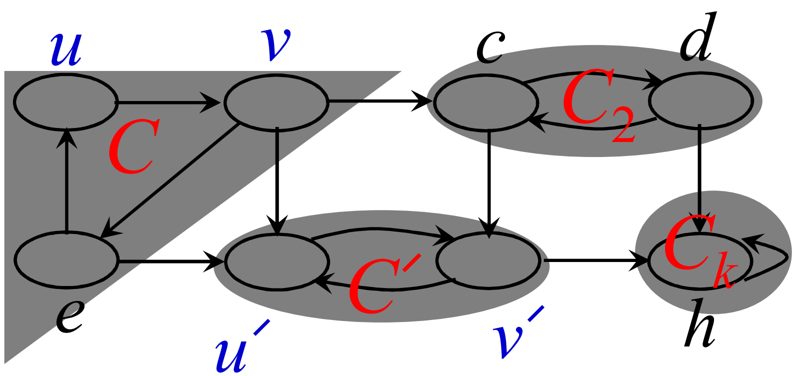
Lemma 22.13

Let C and C′ be **distinct** SCCs in directed graph $G = (V, E)$,
let $u, v \in C$, and $u', v' \in C'$ and suppose that G contains a path u to u'.
Then G cannot also contain a path v' to v



Lemma 22.13

Let C and C′ be **distinct** SCCs in directed graph $G = (V, E)$,
let u, $v \in$ C , and u′, v′ $\in$ C′ and suppose that G contains a path u to u'.
Then G cannot also contain a path v′ to v



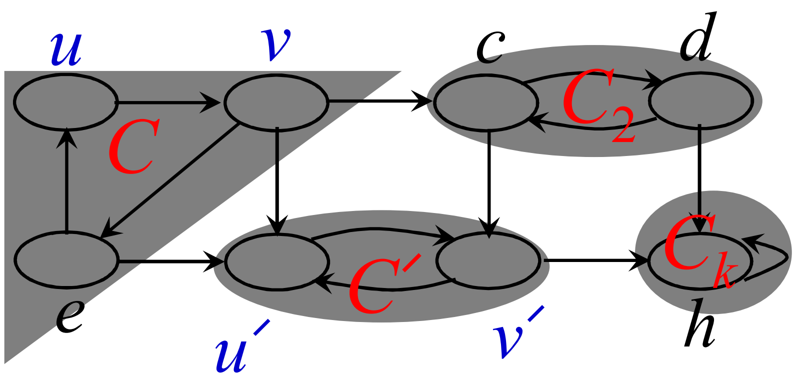
We have $u \leadsto u'$ given

$u' \leadsto v'$ Members of C'

$v \leadsto u$ Members of C

Lemma 22.13

Let C and C′ be **distinct** SCCs in directed graph $G = (V, E)$,
let $u, v \in C$, and $u', v' \in C'$ and suppose that G contains a path u to u'.
Then G cannot also contain a path v' to v



We have $u \leadsto u'$

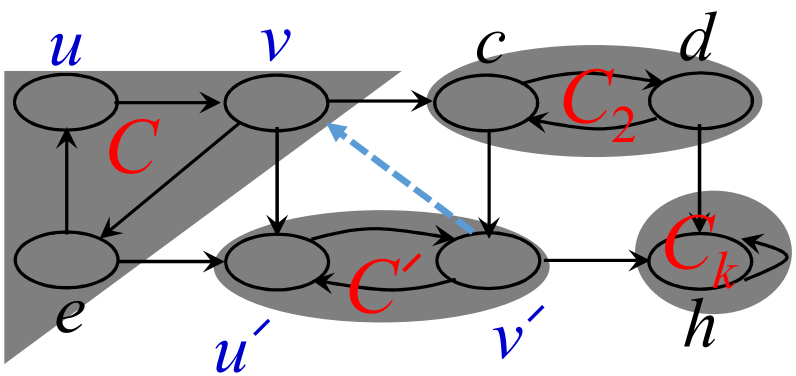
$u' \leadsto v'$

$v \leadsto u$

Therefore, we have $u \leadsto u' \leadsto v'$

Lemma 22.13

Let C and C′ be **distinct** SCCs in directed graph $G = (V, E)$,
let $u, v \in C$, and $u', v' \in C'$ and suppose that G contains a path u to u'.
Then G cannot also contain a path v′ to v



We have $u \rightsquigarrow u'$

$u' \rightsquigarrow v'$

$v \rightsquigarrow u$

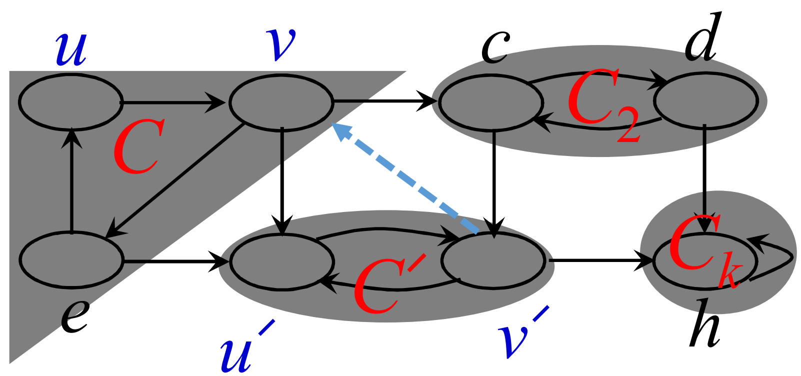
Therefore, we have $u \rightsquigarrow u' \rightsquigarrow v'$

If G contains a path $v' \rightsquigarrow v$

We will have $v' \rightsquigarrow v \rightsquigarrow u$

Lemma 22.13

Let C and C′ be **distinct** SCCs in directed graph $G = (V, E)$,
let $u, v \in C$, and $u', v' \in C'$ and suppose that G contains a path u to u'.
Then G cannot also contain a path v' to v



We have $u \leadsto u'$

$u' \leadsto v'$

$v \leadsto u$

Therefore, we have $u \leadsto u' \leadsto v'$

If G contains a path $v' \leadsto v$

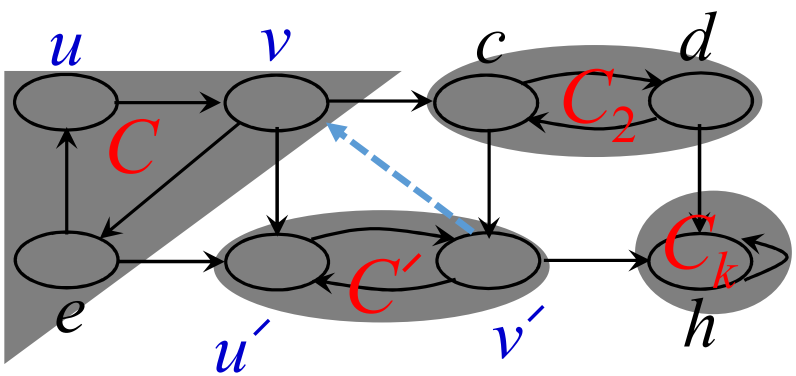
We will have $v' \leadsto v \leadsto u$

We will have a cycle

$u \leadsto u' \leadsto v' \leadsto v \leadsto u$

Lemma 22.13

Let C and C′ be **distinct** SCCs in directed graph $G = (V, E)$,
let $u, v \in C$, and $u', v' \in C'$ and suppose that G contains a path u to u'.
Then G cannot also contain a path v' to v

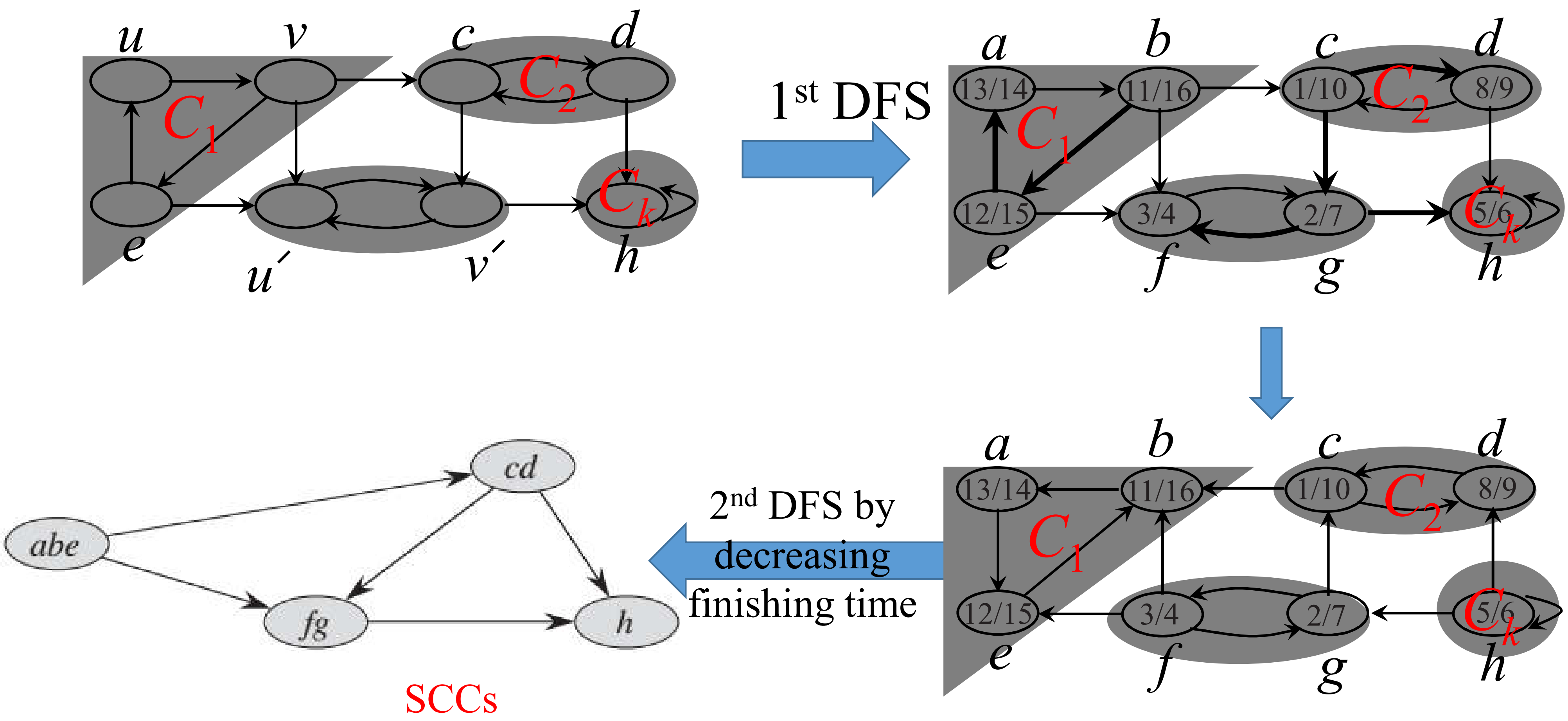


We will have a cycle

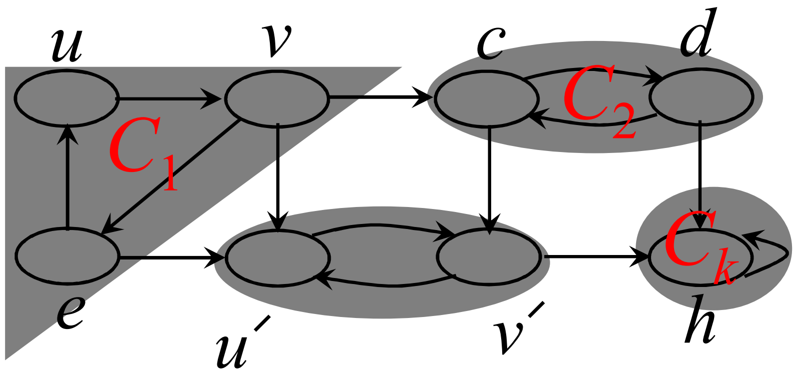
$u \rightsquigarrow u' \rightsquigarrow v' \rightsquigarrow v \rightsquigarrow u$

If C and C′ will NOT be distinct!!
Contradiction!!

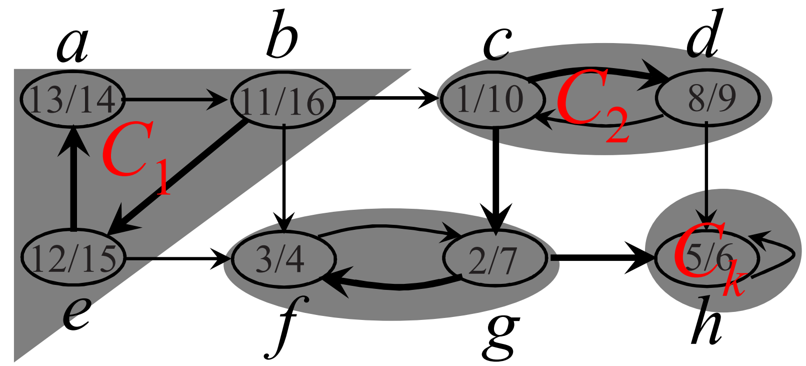
Some insights before next Lemma

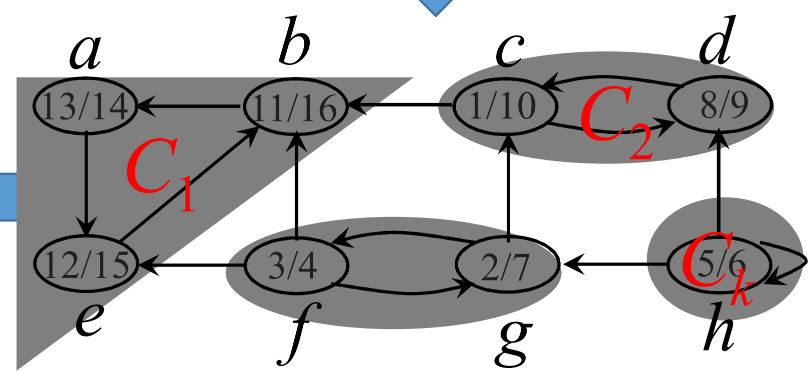


Some insights before next Lemma



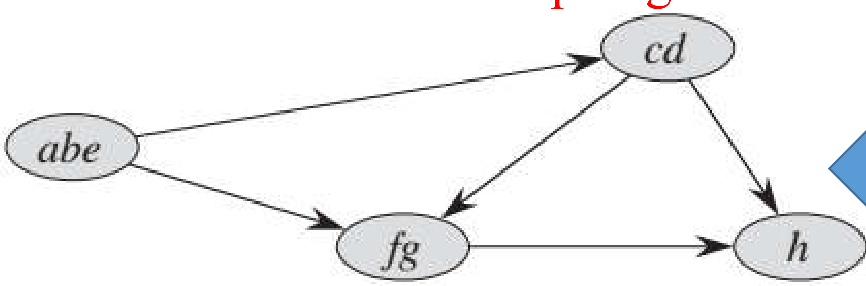
1st DFS





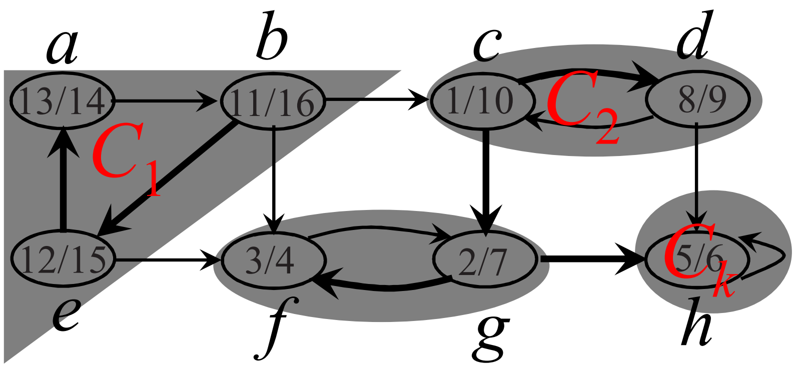
2nd DFS by decreasing finishing time

2nd DFS visits vertices of the Component graph in a topological order



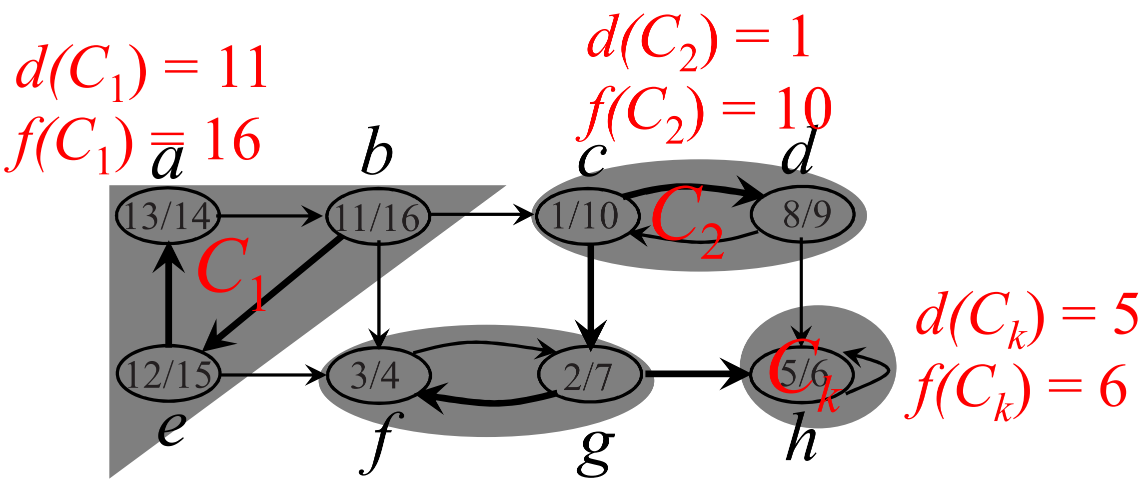
Component graph, G^{SCC}

Some insights before next Lemma



Every vertex has *d* and *f* time

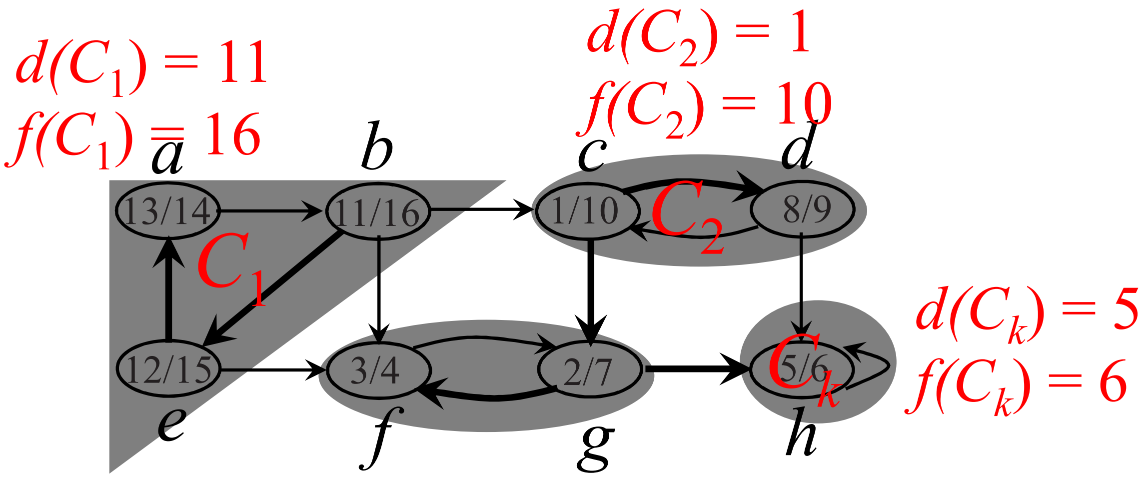
Some insights before next Lemma



Every vertex has *d* and *f* time

Every SCC also has *d* and *f* time

Some insights before next Lemma



Every vertex has d and f time

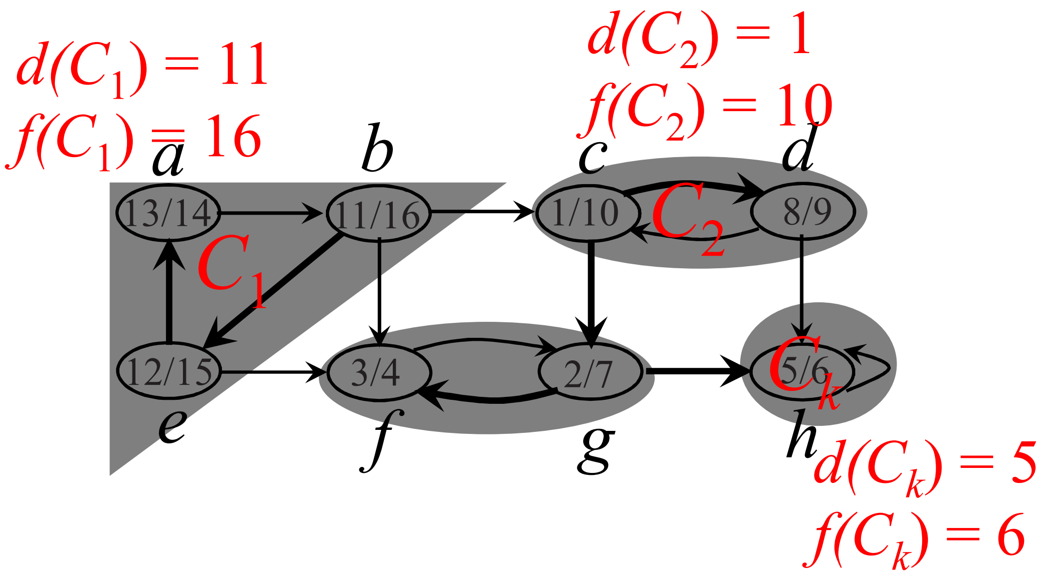
Every SCC also has d and f time

Formally we define,

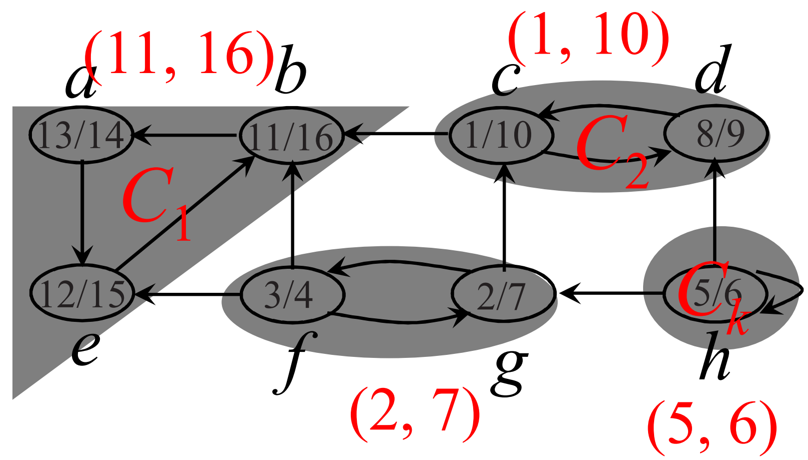
$d(U)=\min_{u \in U} \{u.d\}$

$f(U)=\max_{u \in U} \{u.f\}$

Some insights before next Lemma



Graph, G



Graph, G^T

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

Lemma 22.14

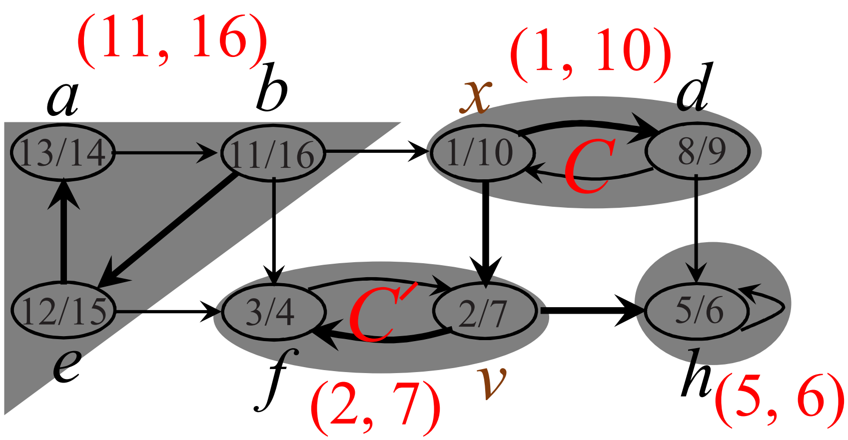
Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

Assume two sub cases
Case 1: $d(C) < d(C')$
Case 2: $d(C) > d(C')$

Lemma 22.14

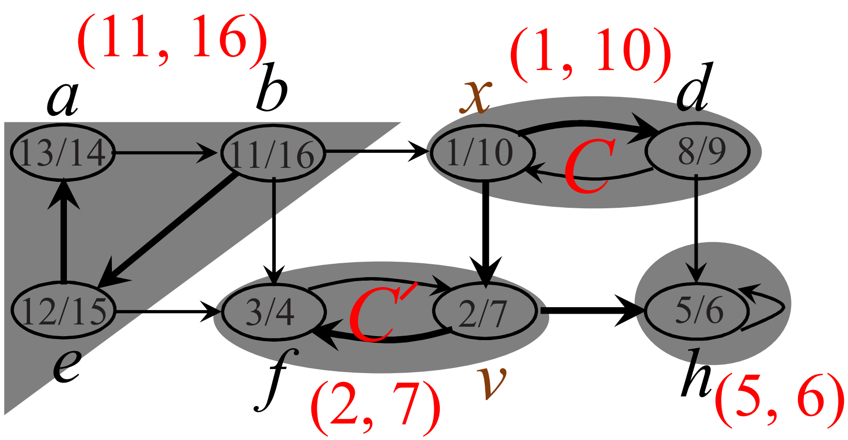
Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

Case1: $d(C) < d(C')$



Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$



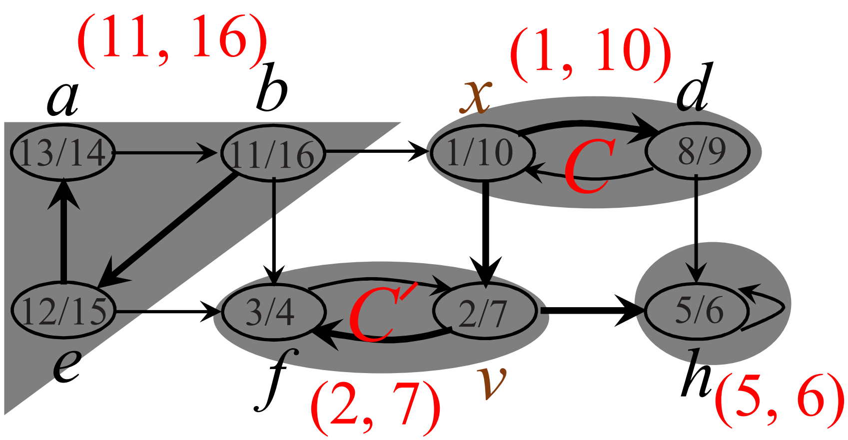
Case1: $d(C) < d(C')$

Let x be the first vertex in C

At $x.d$ all vertices in C and C′ are WHITE

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

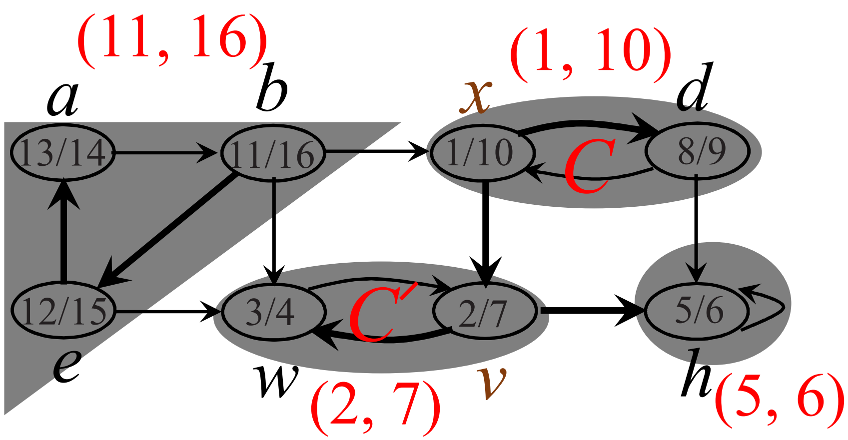


Case1: $d(C) < d(C')$

Let x be the first vertex in C
At $x.d$ all vertices in C and C' are WHITE
All vertices in C and C' are reachable from x by white vertices

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G=(V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

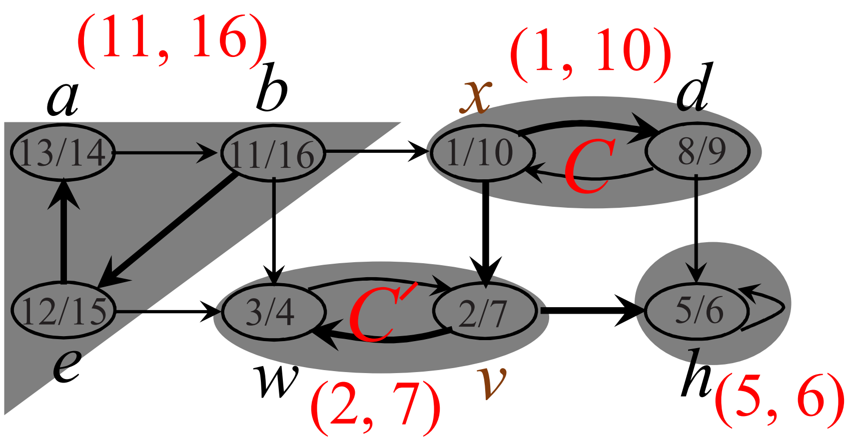


Case1: $d(C) < d(C')$

Let x be the first vertex in C
At $x.d$ all vertices in C and C' are WHITE
All vertices in C and C' are reachable from x by white vertices
Let $w \in C'$,
then $x \rightsquigarrow u \rightarrow v \rightsquigarrow w$

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

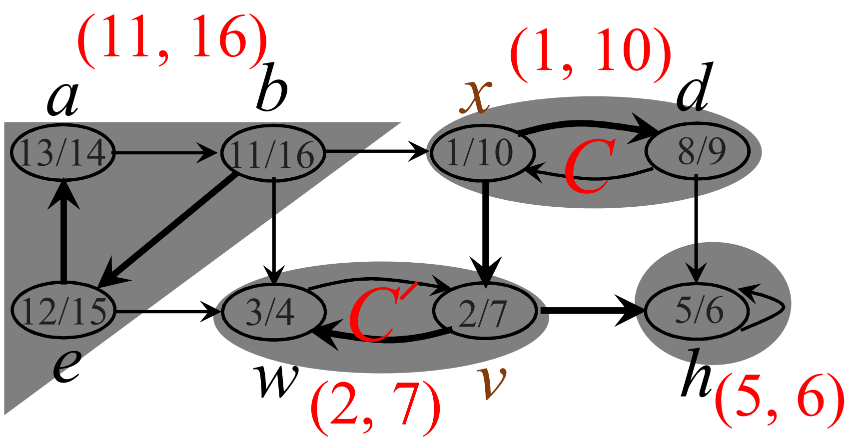


Case1: $d(C) < d(C')$

Let x be the first vertex in C
At $x.d$ all vertices in C and C' are WHITE
All vertices in C and C' are reachable from x by white vertices
Because let $w \in C'$,
Then $x \rightsquigarrow u \rightarrow v \rightsquigarrow w$
x is ancestor of all of C and C' *and will have the largest finishing time.*

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

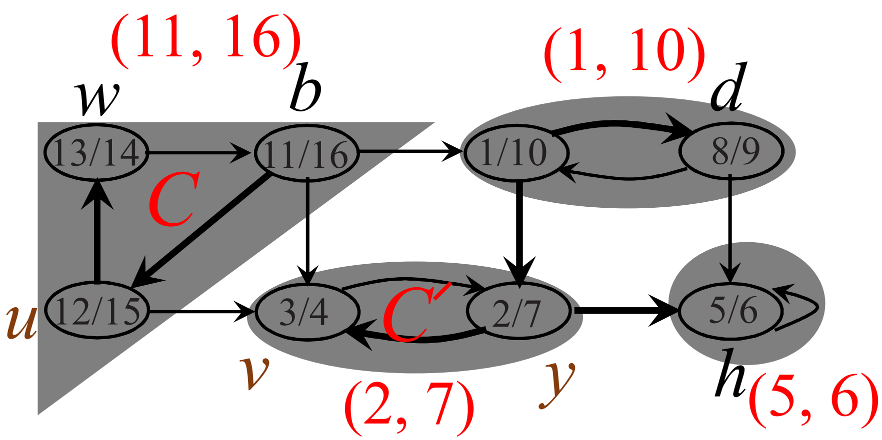


Case1: $d(C) < d(C')$

Let x be the first vertex in C
At $x.d$ all vertices in C and C' are WHITE
All vertices in C and C' are reachable from x by white vertices
Because let $w \in C'$,
Then $x \rightsquigarrow u \rightarrow v \rightsquigarrow w$
x is ancestor of all of C and C' *and will have the largest finishing time.*
Therefor, $x.f = f(C) > f(C')$

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

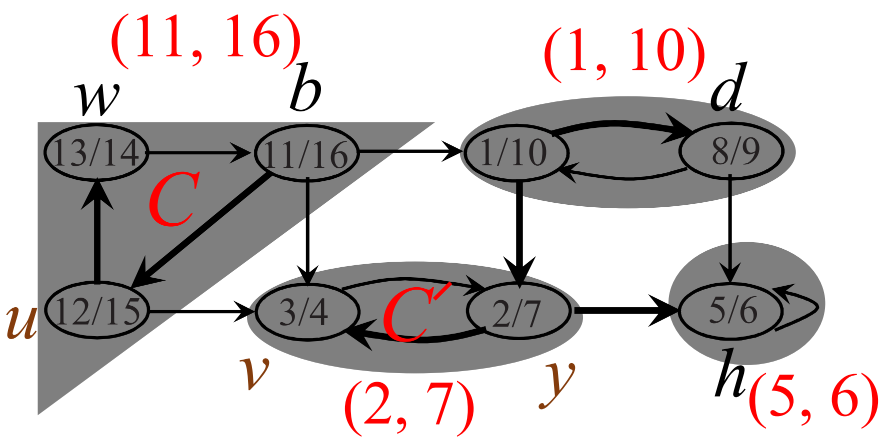


Case 2: $d(C) > d(C')$

Let y be the first vertex in C'

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

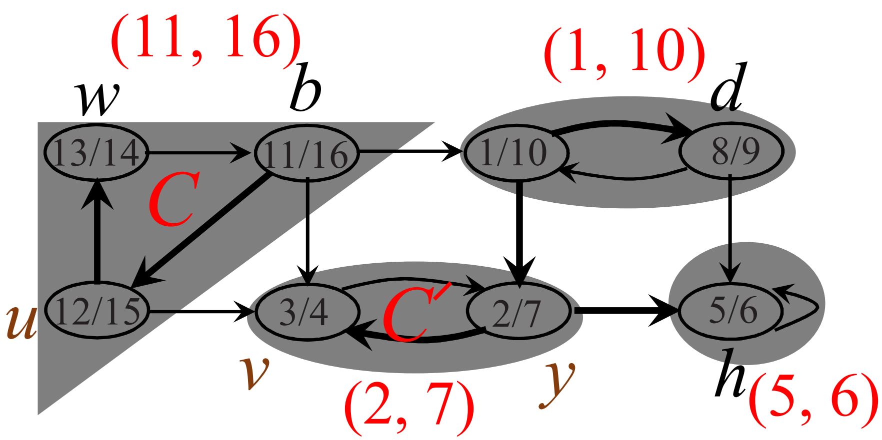


Case 2: $d(C) > d(C')$

Let y be the first vertex in C′
At $y.d$ all vertices in C′ are WHITE and reachable.
Therefore, $y.f = f(C')$

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

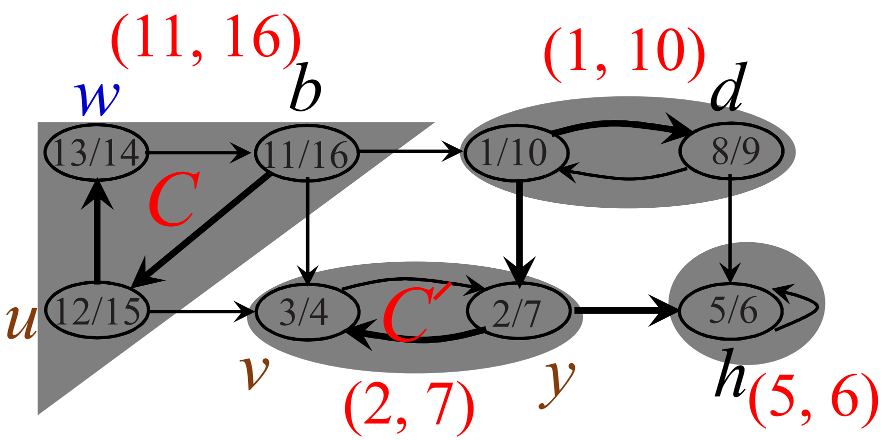


Case 2: $d(C) > d(C')$

Let y be the first vertex in C'
At $y.d$ all vertices in C' are WHITE and reachable.
Therefore, $y.f = f(C')$
At $y.d$, all in C are WHITE but NOT reachable from y as edge is (u, v)

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$

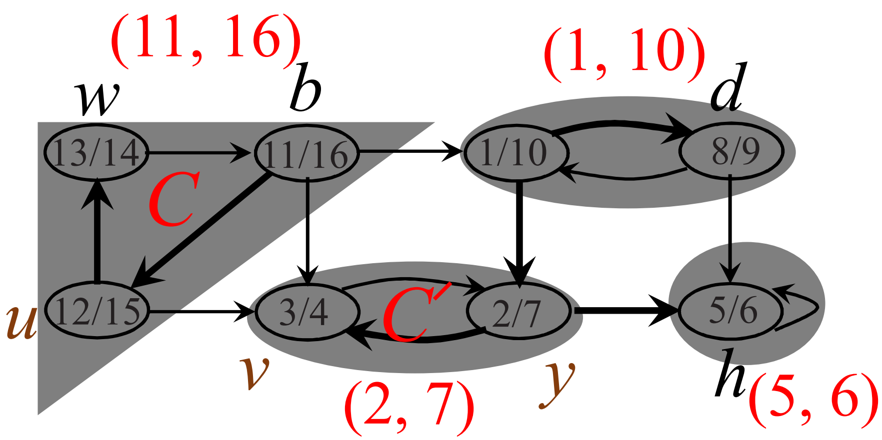


Case 2: $d(C) > d(C')$

Let y be the first vertex in C′
At $y.d$ all vertices in C′ are WHITE and reachable.
Therefore, $y.f = f(C')$
At $y.d$, all in C are WHITE but NOT reachable from y as edge is (u, v)
If $w \in C$, $w.f > y.f$, so is true for others in C

Lemma 22.14

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E$, where $u \in C$ and $v \in C'$
Then $f(C) > f(C')$



Case 2: $d(C) > d(C')$

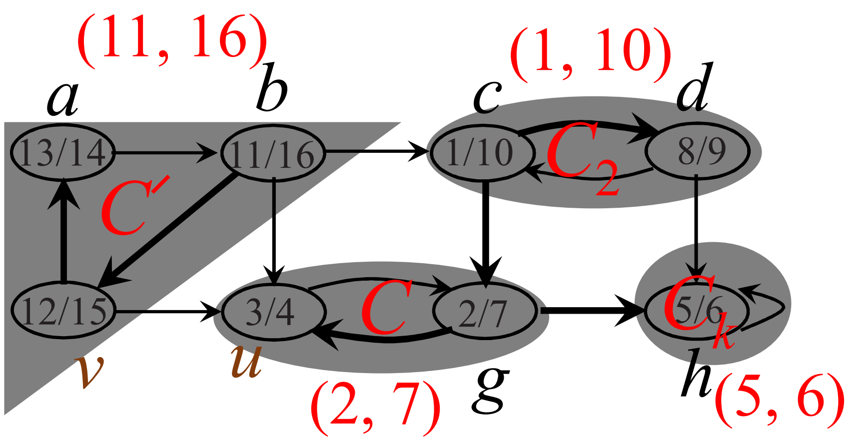
Let y be the first vertex in C′
At $y.d$ all vertices in C′ are WHITE and reachable.
Therefore, $y.f = f(C')$
At $y.d$, all in C are WHITE but NOT reachable from y as edge is (u, v)
If $w \in C$, $w.f > y.f$, so is true for others in C
Therefore, $f(C) > f(C')$

Corollary 22.15

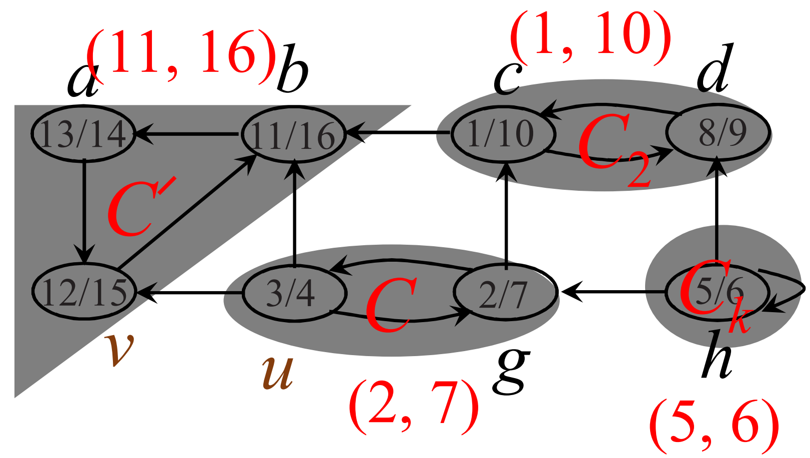
Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E^T$, where $u \in C$ and $v \in C'$
Then $f(C) < f(C')$

Corollary 22.15

Let C and C' be distinct SCCs in directed graph $G = (V, E)$,
Let an edge $(u, v) \in E^T$, where $u \in C$ and $v \in C'$
Then $f(C) < f(C')$



Graph, G



Graph, G^T

Theorem 22.16

The ***Algorithm*:** STRONGLY-CONNECTED-COMPONENTS procedure correctly computes the SCCs of the directed graph G provided as its input.

Theorem 22.16

The ***Algorithm*:** STRONGLY-CONNECTED-COMPONENTS procedure correctly computes the SCCs of the directed graph G provided as its input.

Proof by induction on k (no. of SCCs generated)

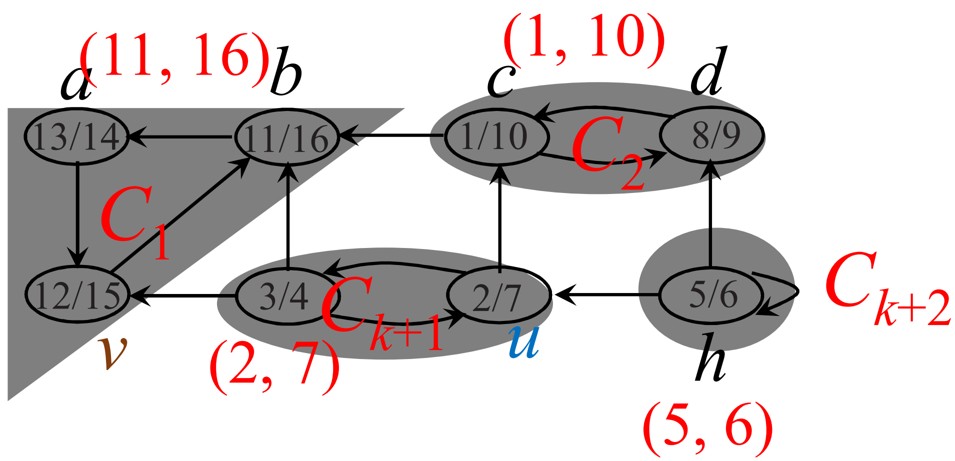
At $k = 0$, it is true.

Theorem 22.16

The ***Algorithm*:** STRONGLY-CONNECTED-COMPONENTS procedure correctly computes the SCCs of the directed graph G provided as its input.

Proof by induction on k (no. of SCCs generated)

Assume that it is true for k, prove it for k +1.



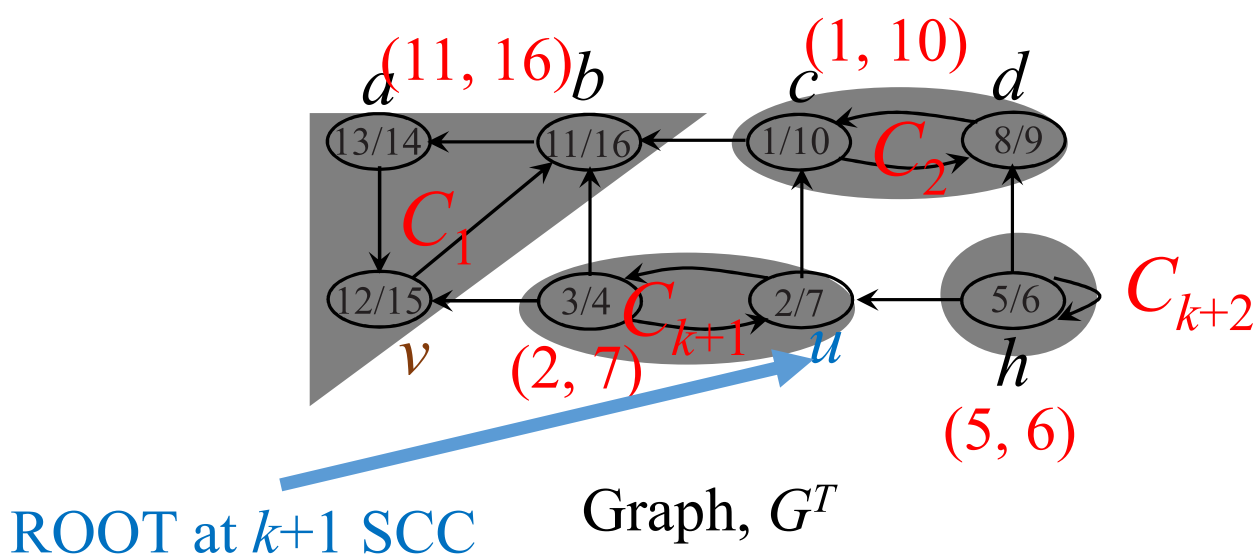
Graph, G^T

Theorem 22.16

The ***Algorithm*:** STRONGLY-CONNECTED-COMPONENTS procedure correctly computes the SCCs of the directed graph G provided as its input.

Proof by induction on k (no. of SCCs generated)

Assume that it is true for k, prove it for $k+1$.



Graph, G^T

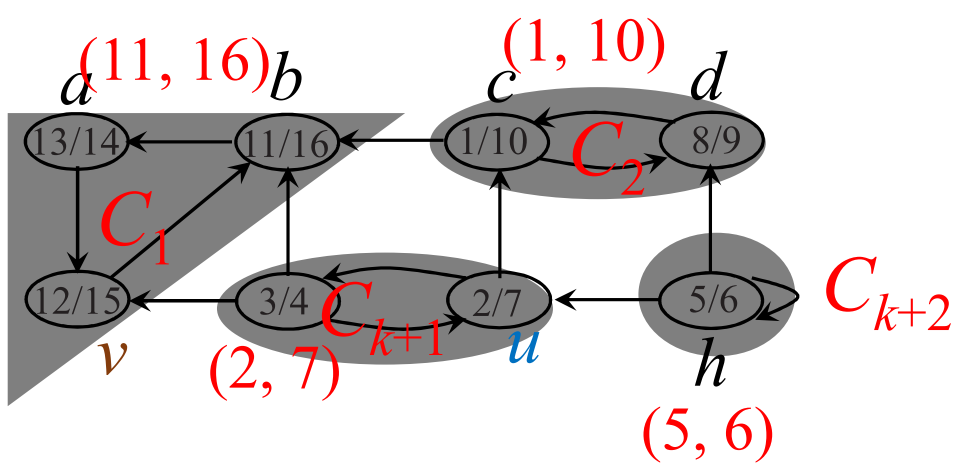
Theorem 22.16

The ***Algorithm*: STRONGLY-CONNECTED-COMPONENTS** procedure correctly computes the SCCs of the directed graph G provided as its input.

Proof by induction on k (no. of SCCs generated)

Assume that it is true for k, prove it for $k+1$

all vertices in C_{k+1} are WHITE at $u.d$ and are REACHABLE.
all vertices in C_{k+1} are descendant of u



Graph, G^T

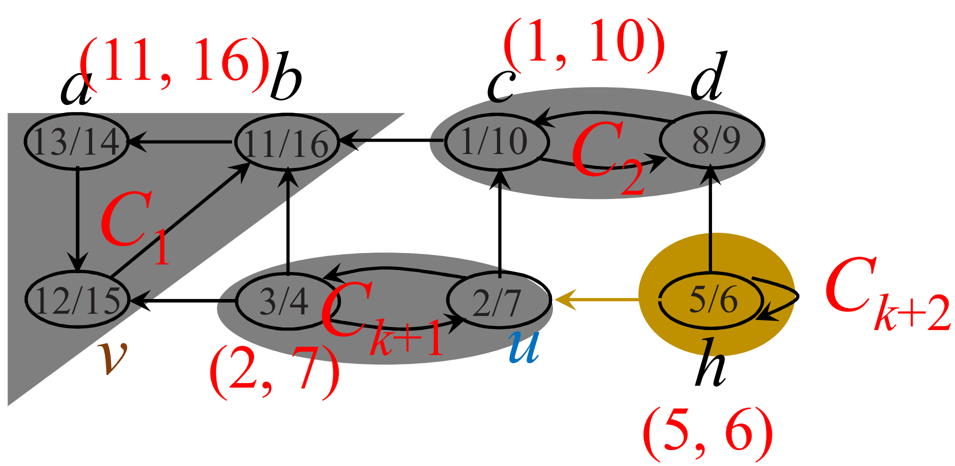
Theorem 22.16

The ***Algorithm*:** STRONGLY-CONNECTED-COMPONENTS procedure correctly computes the SCCs of the directed graph G provided as its input.

Proof by induction on k (no. of SCCs generated)

Assume that it is true for k, prove it for $k+1$

all vertices in C_{k+1} are WHITE at $u.d$ and are REACHABLE
all vertices in C_{k+1} are descendant of u
There is NO path from u to vertices of C_{k+2}, C_{k+3}, C_{k+4}, …
They are not reachable



Graph, G^T

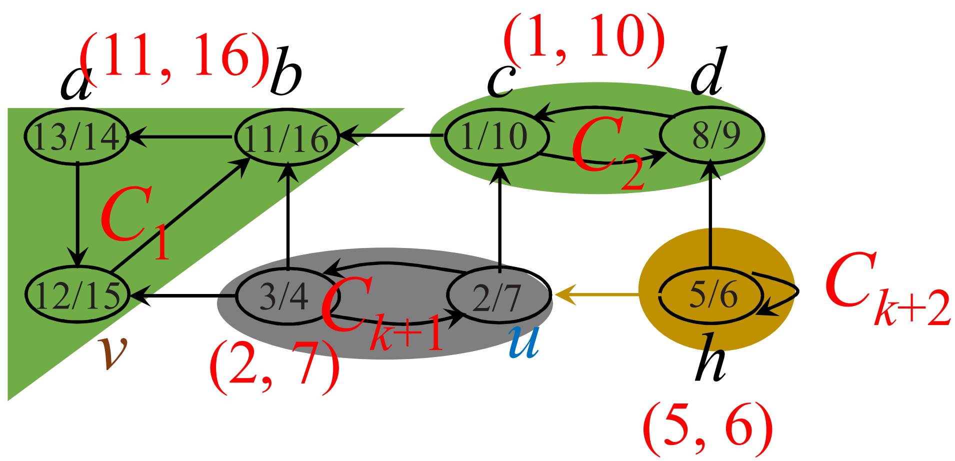
Theorem 22.16

The ***Algorithm*:** STRONGLY-CONNECTED-COMPONENTS procedure correctly computes the SCCs of the directed graph G provided as its input.

Proof by induction on k (no. of SCCs generated)

Assume that it is true for k, prove it for $k+1$

all vertices in C_{k+1} are WHITE at $u.d$ and are REACHABLE
all vertices in C_{k+1} are descendant of u
There are PATHs from u to vertices of $C_1, C_2, \ldots, C_k$
But They are ALREADY VISITED



Graph, G^T

Theorem 22.16

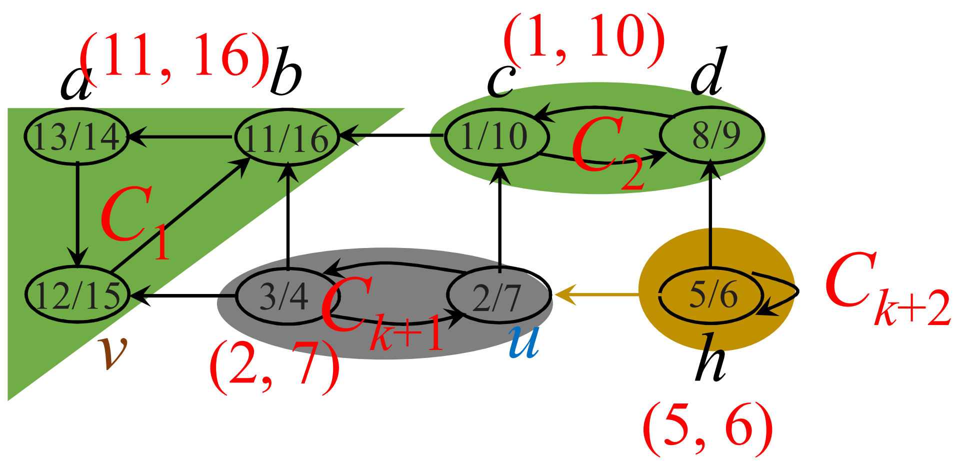
The ***Algorithm*:** STRONGLY-CONNECTED-COMPONENTS procedure correctly computes the SCCs of the directed graph G provided as its input.

Proof by induction on k (no. of SCCs generated)

Assume that it is true for k, prove it for $k+1$

all vertices in C_{k+1} are WHITE at $u.d$ and are REACHABLE
all vertices in C_{k+1} are descendant of u
There are PATHs from u to vertices of $C_1, C_2, \ldots, C_k$

2nd DFS will not visit them again



Graph, G^T

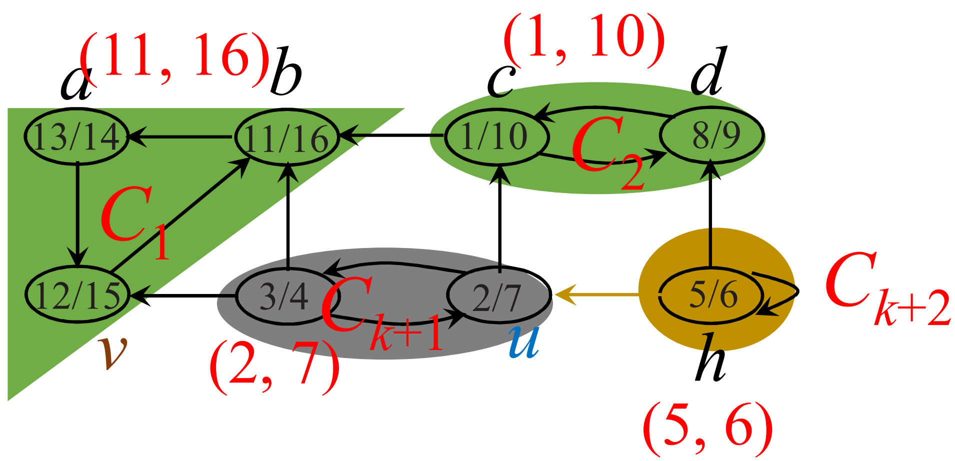
Theorem 22.16

The ***Algorithm*: STRONGLY-CONNECTED-COMPONENTS** procedure correctly computes the SCCs of the directed graph G provided as its input.

Proof by induction on k (no. of SCCs generated)

Assume that it is true for k, prove it for k +1

all vertices in C_{k+1} are WHITE at $u.d$ and are REACHABLE
all vertices in C_{k+1} are descendant of u
There are PATHs from u to vertices of $C_1, C_2, \ldots, C_k$
Therefore, C_{k+1} will be correctly identified and will NOT include any other vertex from other components (Green or Yellow).



Graph, G^T