

Tarjan's SCC Algorithm

Fundamental Algorithms

Rodica Potolea, Camelia Lemnaru and Ciprian Oprea

Technical University of Cluj-Napoca
Computer Science Department

TARJAN(G)

```

1  global  $index = 0$ 
2  global  $S = \text{EMPTY-STACK}$ 
3  global  $nrComponents = 0$ 
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12       $nrComponents = nrComponents + 1$ 
13      repeat
14           $v = \text{POP}(S)$ 
15           $v.onStack = \text{FALSE}$ 
16           $v.comp = nrComponents$ 
17      until  $v == u$ 

```

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

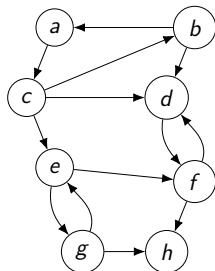
STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```

call stack



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

```

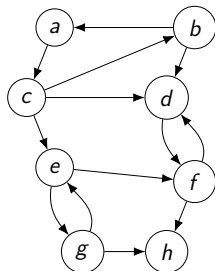
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      nrComponents = nrComponents + 1
13      repeat
14           $v = \text{POP}(S)$ 
15           $v.onStack = \text{FALSE}$ 
16           $v.comp = nrComponents$ 
17      until  $v == u$ 

```

call stack S

index : 0

nrComponents : 0



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

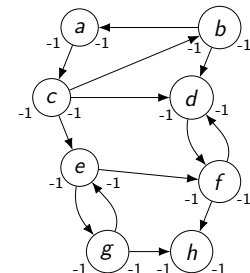
STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```

call stack S



index : 0

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

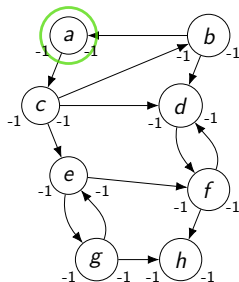
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 0

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

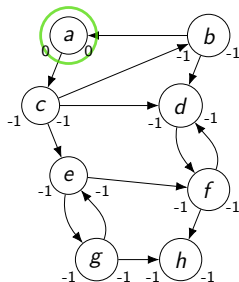
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 0

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

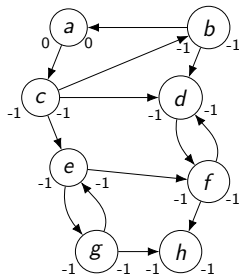
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 1

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

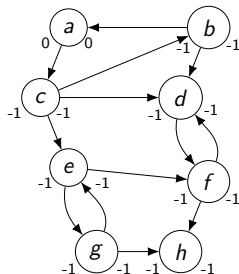
```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = \text{index}$ 
2   $\text{index} = \text{index} + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = \text{nrComponents}$ 
17     until  $v == u$ 

```



call stack

S

index : 1

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

```

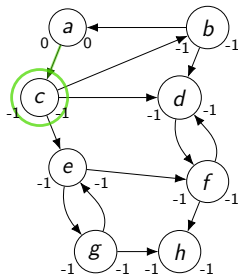
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 1

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

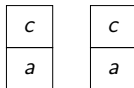
```

STRONG-CONNECT(G, u)

```

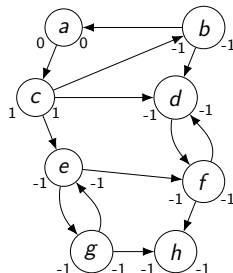
1   $u.index = u.lowLink = \text{index}$ 
2   $\text{index} = \text{index} + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 2

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

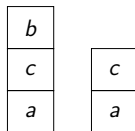
```

STRONG-CONNECT(G, u)

```

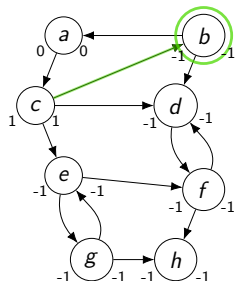
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 2

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

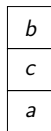
```

STRONG-CONNECT(G, u)

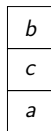
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

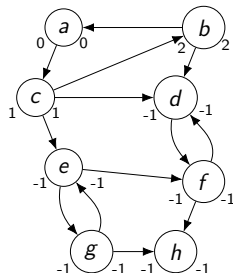
```



call stack



S



index : 3

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = \text{index}$ 
2   $\text{index} = \text{index} + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      nrComponents = nrComponents + 1
13      repeat
14           $v = \text{POP}(S)$ 
15           $v.onStack = \text{FALSE}$ 
16           $v.comp = nrComponents$ 
17      until  $v == u$ 

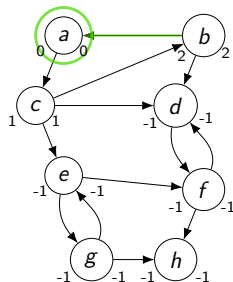
```

b
c
a

call stack

b
c
a

S



index : 3

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

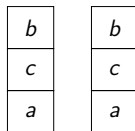
```

STRONG-CONNECT(G, u)

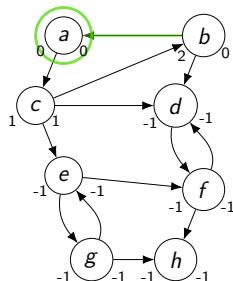
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12       $nrComponents = nrComponents + 1$ 
13      repeat
14           $v = POP(S)$ 
15           $v.onStack = FALSE$ 
16           $v.comp = nrComponents$ 
17      until  $v == u$ 

```



call stack S



$index : 3$

$nrComponents : 0$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

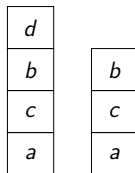
```

STRONG-CONNECT(G, u)

```

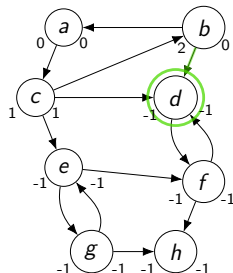
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



$index : 3$

$nrComponents : 0$



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

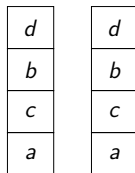
```

STRONG-CONNECT(G, u)

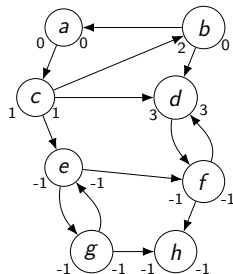
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 4

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

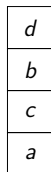
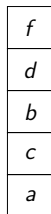
```

STRONG-CONNECT(G, u)

```

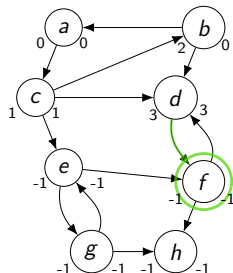
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 4

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

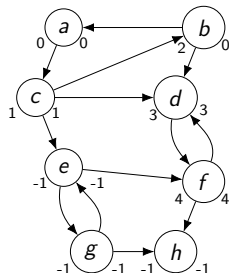
```

f
d
b
c
a

call stack

f
d
b
c
a

S



index : 5

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

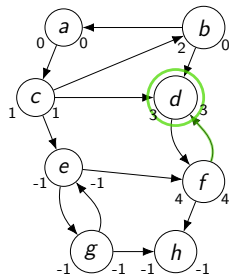
```

f
d
b
c
a

call stack

f
d
b
c
a

S



index : 5

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

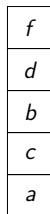
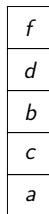
```

STRONG-CONNECT(G, u)

```

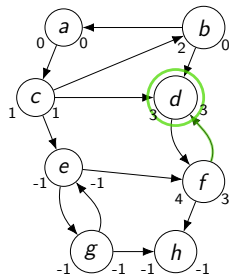
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12       $nrComponents = nrComponents + 1$ 
13      repeat
14           $v = POP(S)$ 
15           $v.onStack = FALSE$ 
16           $v.comp = nrComponents$ 
17      until  $v == u$ 

```



call stack

S



index : 5

nrComponents : 0



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

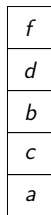
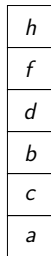
```

STRONG-CONNECT(G, u)

```

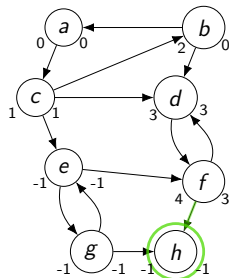
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



$index : 5$

$nrComponents : 0$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

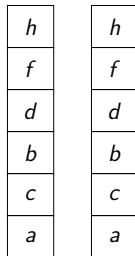
```

STRONG-CONNECT(G, u)

```

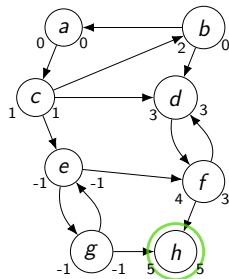
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 6

nrComponents : 0

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

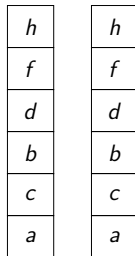
```

STRONG-CONNECT(G, u)

```

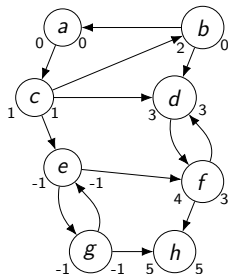
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13  repeat
14      $v = POP(S)$ 
15      $v.onStack = FALSE$ 
16      $v.comp = nrComponents$ 
17  until  $v == u$ 

```



call stack

S



index : 6

nrComponents : 1



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

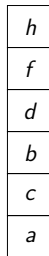
```

STRONG-CONNECT(G, u)

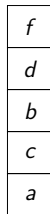
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

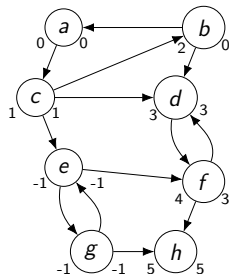
```



call stack



S



$index : 6$

$nrComponents : 1$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

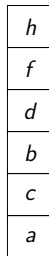
```

STRONG-CONNECT(G, u)

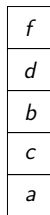
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

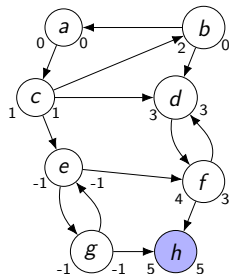
```



call stack



S



$index : 6$

$nrComponents : 1$



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

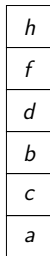
```

STRONG-CONNECT(G, u)

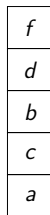
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

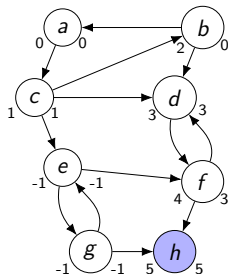
```



call stack



S



$index : 6$

$nrComponents : 1$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

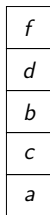
```

STRONG-CONNECT(G, u)

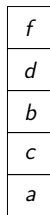
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

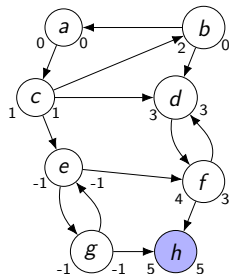
```



call stack



S



index : 6

nrComponents : 1

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

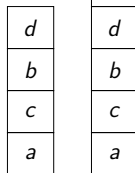
```

STRONG-CONNECT(G, u)

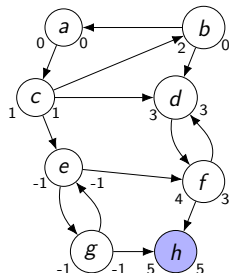
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13  repeat
14      $v = POP(S)$ 
15      $v.onStack = FALSE$ 
16      $v.comp = nrComponents$ 
17  until  $v == u$ 

```



call stack S



index : 6

nrComponents : 2



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

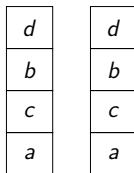
```

STRONG-CONNECT(G, u)

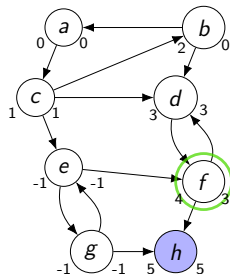
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 6$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

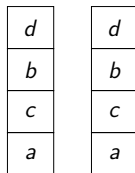
```

STRONG-CONNECT(G, u)

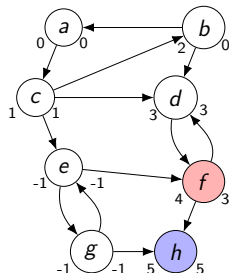
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 6$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

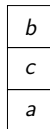
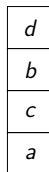
```

STRONG-CONNECT(G, u)

```

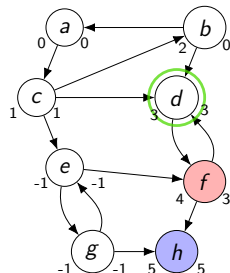
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 6

nrComponents : 2

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

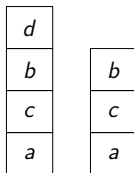
```

STRONG-CONNECT(G, u)

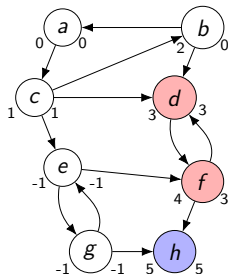
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 6$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

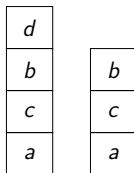
```

STRONG-CONNECT(G, u)

```

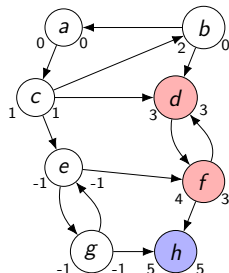
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



$index : 6$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

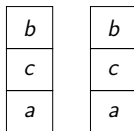
```

STRONG-CONNECT(G, u)

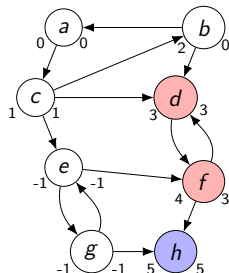
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 6$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

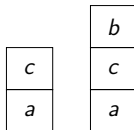
```

STRONG-CONNECT(G, u)

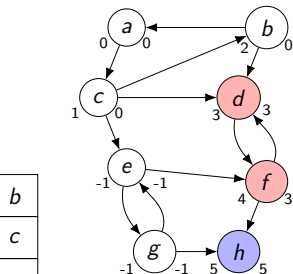
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack



index : 6

nrComponents : 2

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

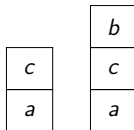
```

STRONG-CONNECT(G, u)

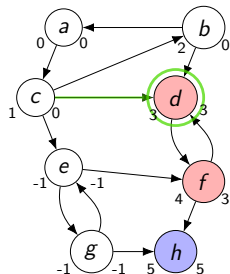
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 6$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

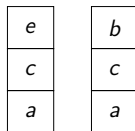
```

STRONG-CONNECT(G, u)

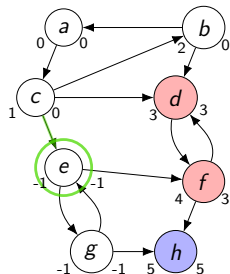
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 6$

$nrComponents : 2$



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

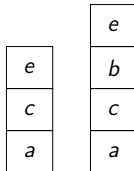
```

STRONG-CONNECT(G, u)

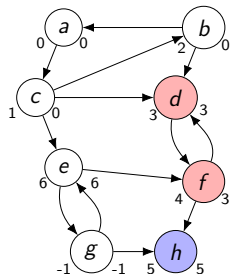
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 7$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

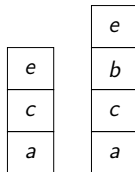
```

STRONG-CONNECT(G, u)

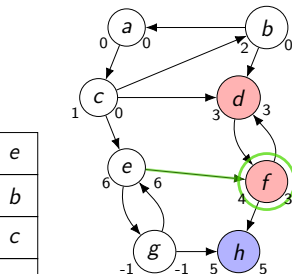
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack



index : 7

nrComponents : 2



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

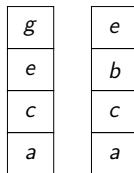
```

STRONG-CONNECT(G, u)

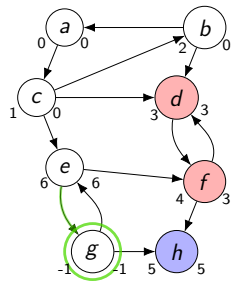
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 7$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

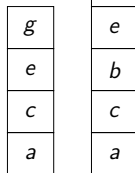
```

STRONG-CONNECT(G, u)

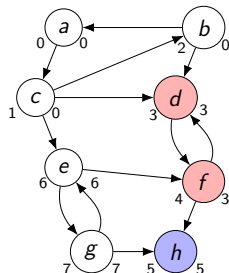
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 8

nrComponents : 2

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

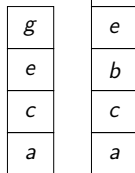
```

STRONG-CONNECT(G, u)

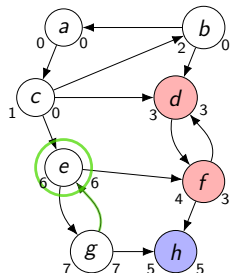
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 8$

$nrComponents : 2$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

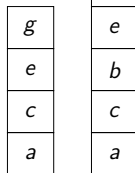
```

STRONG-CONNECT(G, u)

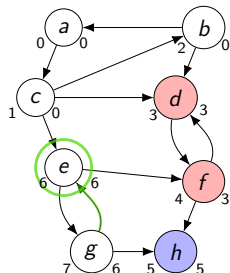
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      nrComponents = nrComponents + 1
13      repeat
14           $v = \text{POP}(S)$ 
15           $v.onStack = \text{FALSE}$ 
16           $v.comp = nrComponents$ 
17      until  $v == u$ 

```



call stack S



index : 8

nrComponents : 2



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

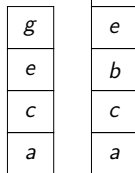
```

STRONG-CONNECT(G, u)

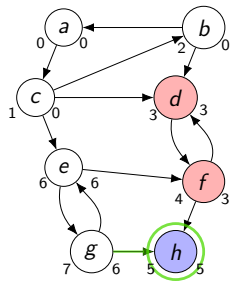
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 8$

$nrComponents : 2$



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

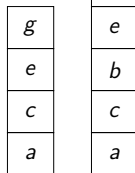
```

STRONG-CONNECT(G, u)

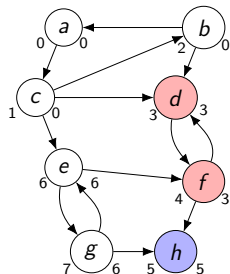
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 8

nrComponents : 2

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

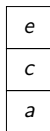
```

STRONG-CONNECT(G, u)

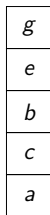
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13  repeat
14      $v = POP(S)$ 
15      $v.onStack = FALSE$ 
16      $v.comp = nrComponents$ 
17  until  $v == u$ 

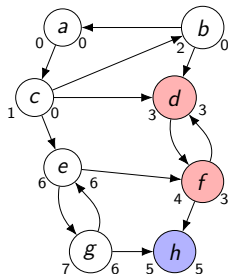
```



call stack



S



index : 8

nrComponents : 3

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

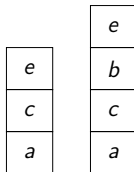
```

STRONG-CONNECT(G, u)

```

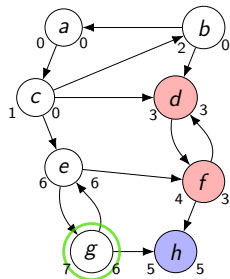
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 8

nrComponents : 3

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

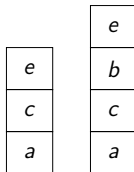
```

STRONG-CONNECT(G, u)

```

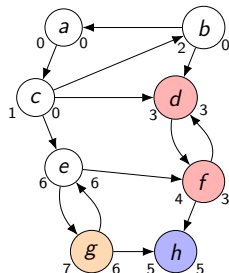
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 8

nrComponents : 3

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

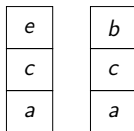
```

STRONG-CONNECT(G, u)

```

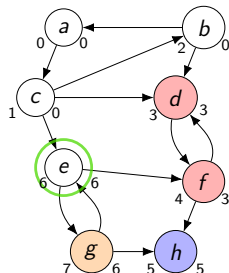
1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack

S



index : 8

nrComponents : 3

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

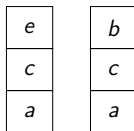
```

STRONG-CONNECT(G, u)

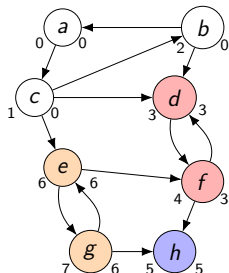
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 8$

$nrComponents : 3$

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

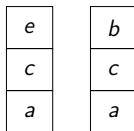
```

STRONG-CONNECT(G, u)

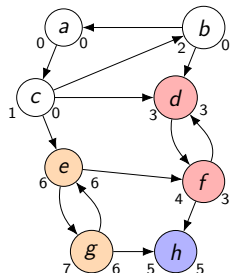
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



$index : 8$

$nrComponents : 3$



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

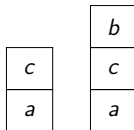
```

STRONG-CONNECT(G, u)

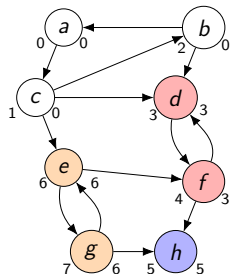
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 8

nrComponents : 3

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

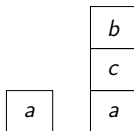
```

STRONG-CONNECT(G, u)

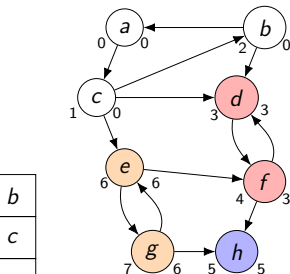
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12     nrComponents = nrComponents + 1
13  repeat
14      $v = \text{POP}(S)$ 
15      $v.onStack = \text{FALSE}$ 
16      $v.comp = nrComponents$ 
17  until  $v == u$ 

```



call stack



index : 8

nrComponents : 4

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

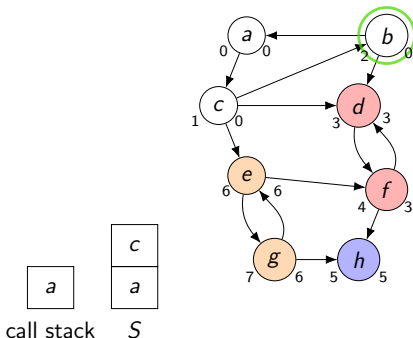
```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



index : 8

nrComponents : 4



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

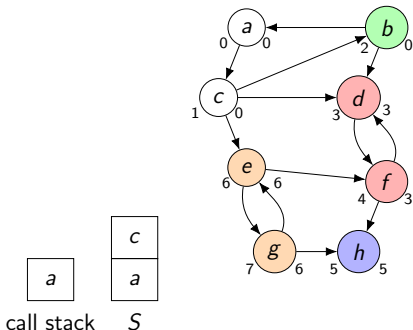
```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



index : 8

nrComponents : 4

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

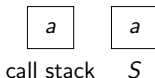
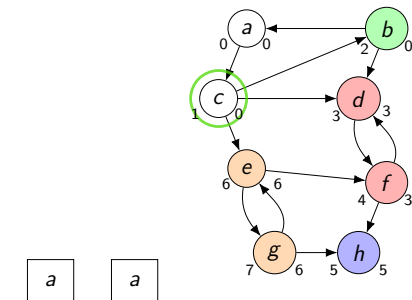
```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = \text{index}$ 
2   $\text{index} = \text{index} + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12       $nrComponents = nrComponents + 1$ 
13      repeat
14           $v = \text{POP}(S)$ 
15           $v.onStack = \text{FALSE}$ 
16           $v.comp = nrComponents$ 
17      until  $v == u$ 

```



index : 8

nrComponents : 4

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

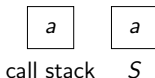
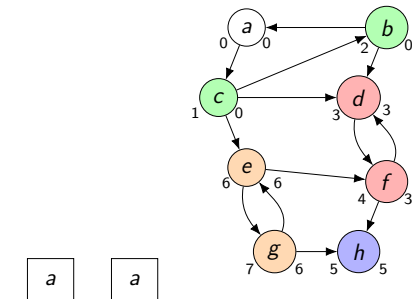
```

STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



index : 8

nrComponents : 4



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

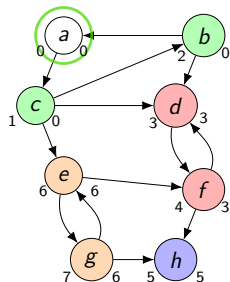
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12       $nrComponents = nrComponents + 1$ 
13      repeat
14           $v = \text{POP}(S)$ 
15           $v.onStack = \text{FALSE}$ 
16           $v.comp = nrComponents$ 
17      until  $v == u$ 

```



call stack S



index : 8

nrComponents : 4

TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

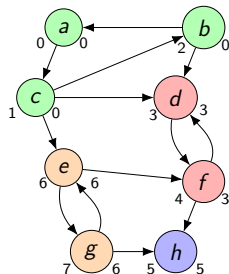
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 8

nrComponents : 4



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = FALSE$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

STRONG-CONNECT(G, u)

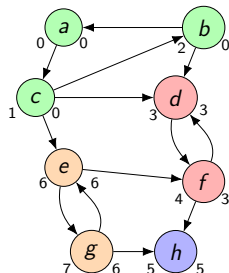
```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = TRUE$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = POP(S)$ 
15          $v.onStack = FALSE$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```



call stack S



index : 8

nrComponents : 4



TARJAN(G)

```

1  global index = 0
2  global S = EMPTY-STACK
3  global nrComponents = 0
4  for each vertex  $u \in G.V$ 
5       $u.index = u.lowLink = -1$ 
6       $u.onStack = \text{FALSE}$ 
7       $u.comp = 0$ 
8  for each vertex  $u \in G.V$ 
9      if  $u.index == -1$ 
10         STRONG-CONNECT( $G, u$ )

```

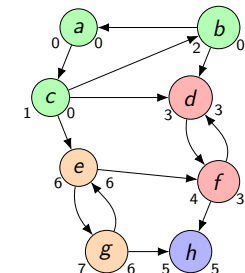
STRONG-CONNECT(G, u)

```

1   $u.index = u.lowLink = index$ 
2   $index = index + 1$ 
3  PUSH( $S, u$ )
4   $u.onStack = \text{TRUE}$ 
5  for each  $v \in G.Adj[u]$ 
6      if  $v.index == -1$ 
7          STRONG-CONNECT( $G, v$ )
8           $u.lowLink = \min(u.lowLink, v.lowLink)$ 
9      elseif  $v.onStack$ 
10          $u.lowLink = \min(u.lowLink, v.index)$ 
11  if  $u.lowLink == u.index$ 
12      $nrComponents = nrComponents + 1$ 
13     repeat
14          $v = \text{POP}(S)$ 
15          $v.onStack = \text{FALSE}$ 
16          $v.comp = nrComponents$ 
17     until  $v == u$ 

```

call stack S



index : 8

nrComponents : 4