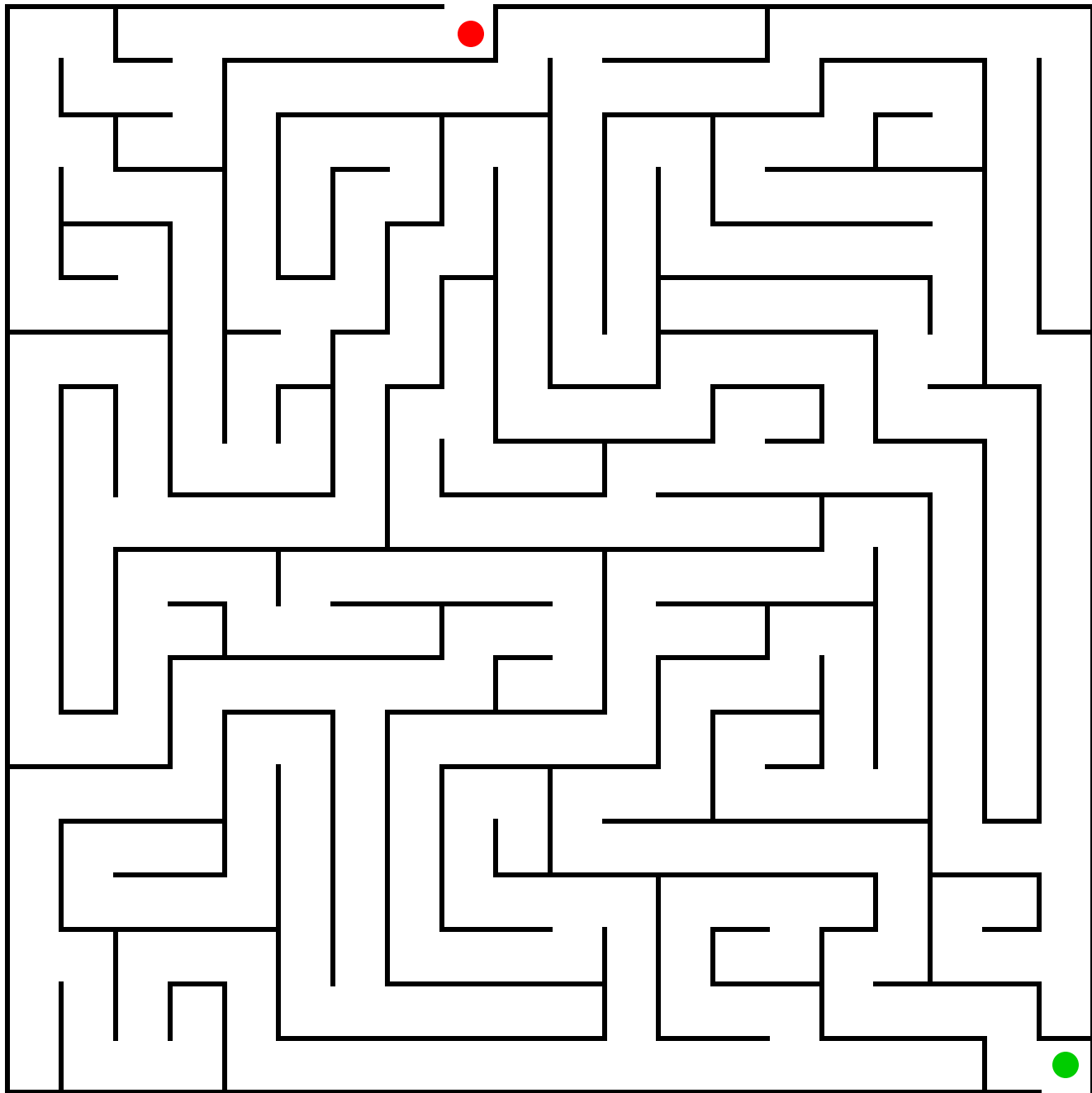


# Algoritmer og Datastrukturer 2

Gerth Stølting Brodal

Topologisk Sortering, Stærke Sammenhængskomponenter  
[CLRS, kapitel 22.4-22.5]





# Dybde Først Søgning (DFS)

DFS( $G$ )

```
1 for each vertex  $u \in G.V$ 
2    $u.color = WHITE$ 
3    $u.\pi = NIL$ 
4    $time = 0$ 
5 for each vertex  $u \in G.V$ 
6   if  $u.color == WHITE$ 
7     DFS-VISIT( $G, u$ )
```

$u.color$

WHITE = knuderne endnu ikke besøgt  
GRAY = knuder på rekursionsstakken  
BLACK = knuderne besøgt

$u.\pi$  = faderen til  $u$  i DFS træet

$u.d$  = "discover time" for  $u$

$u.f$  = "finishing time" for  $u$

DFS-VISIT( $G, u$ )

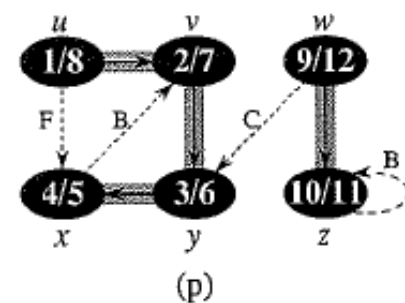
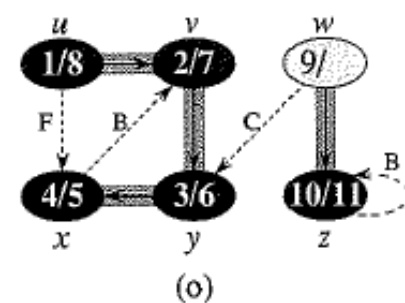
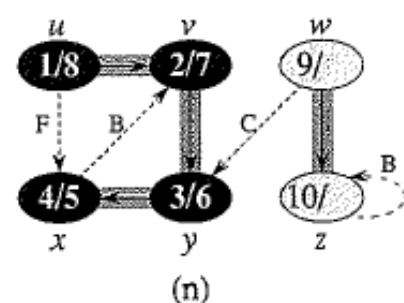
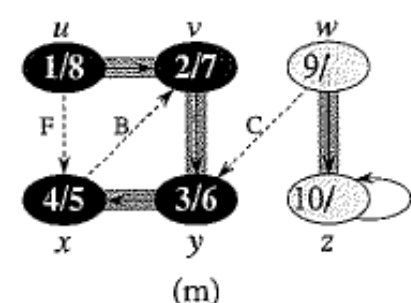
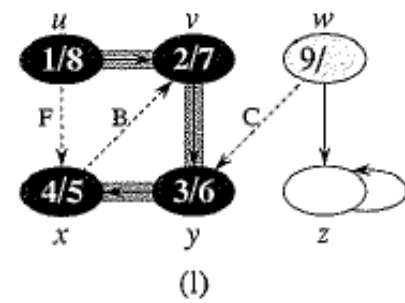
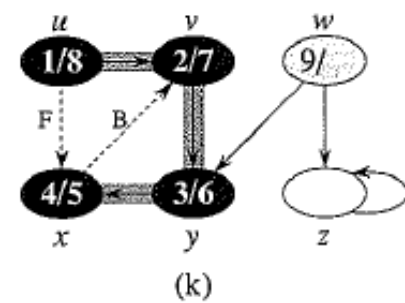
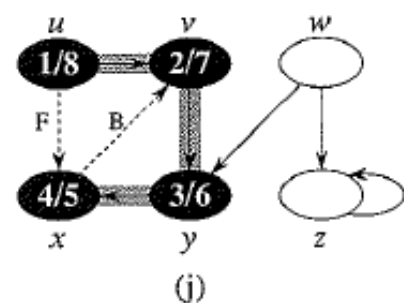
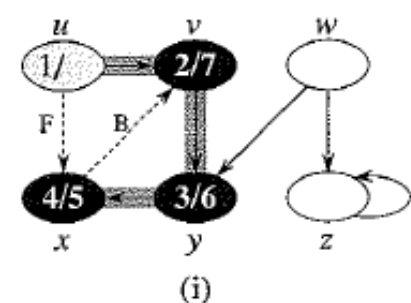
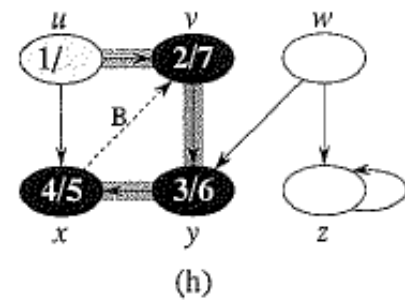
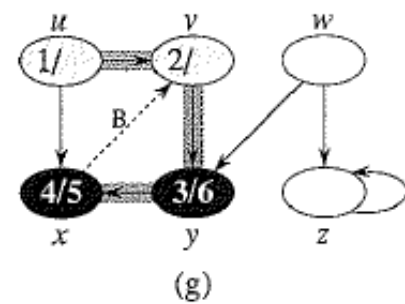
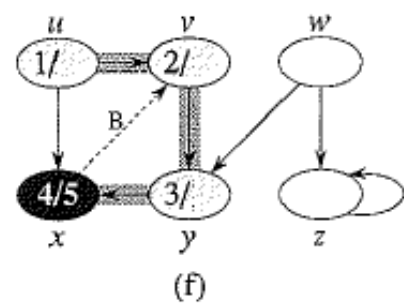
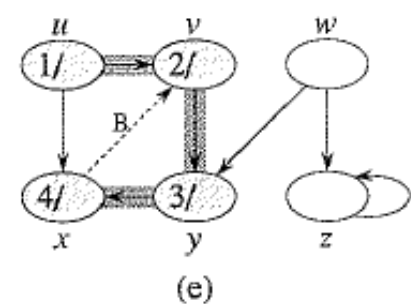
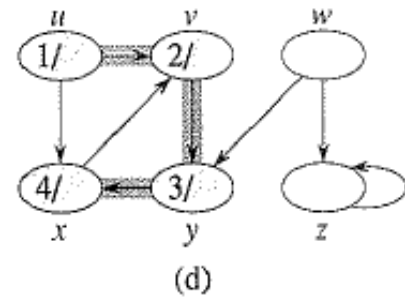
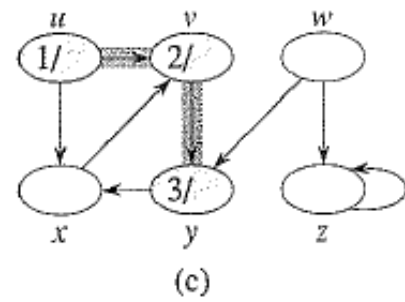
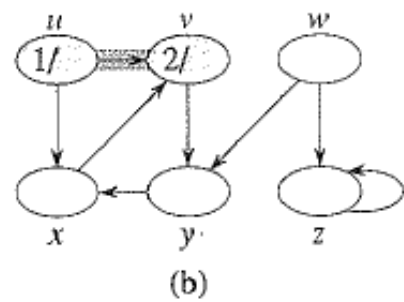
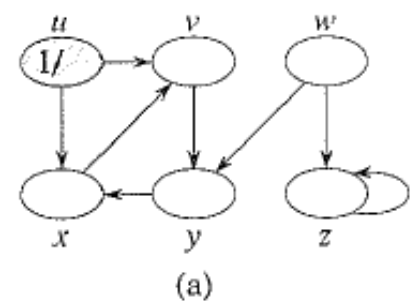
```
1  $time = time + 1$ 
2  $u.d = time$ 
3  $u.color = GRAY$ 
4 for each  $v \in G.Adj[u]$ 
5   if  $v.color == WHITE$ 
6      $v.\pi = u$ 
7     DFS-VISIT( $G, v$ )
8  $u.color = BLACK$ 
9  $time = time + 1$ 
10  $u.f = time$ 
```

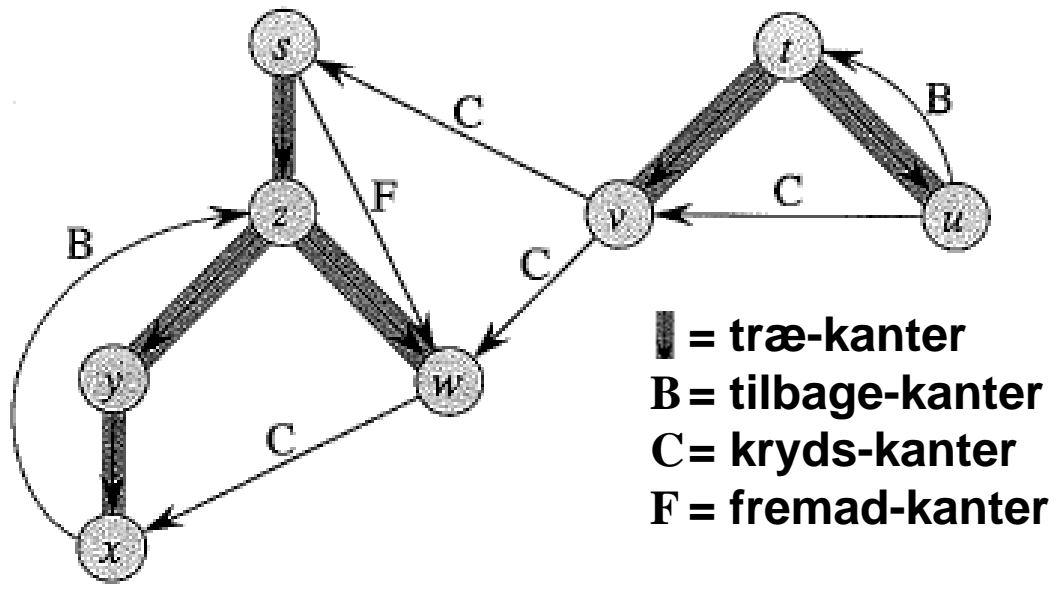
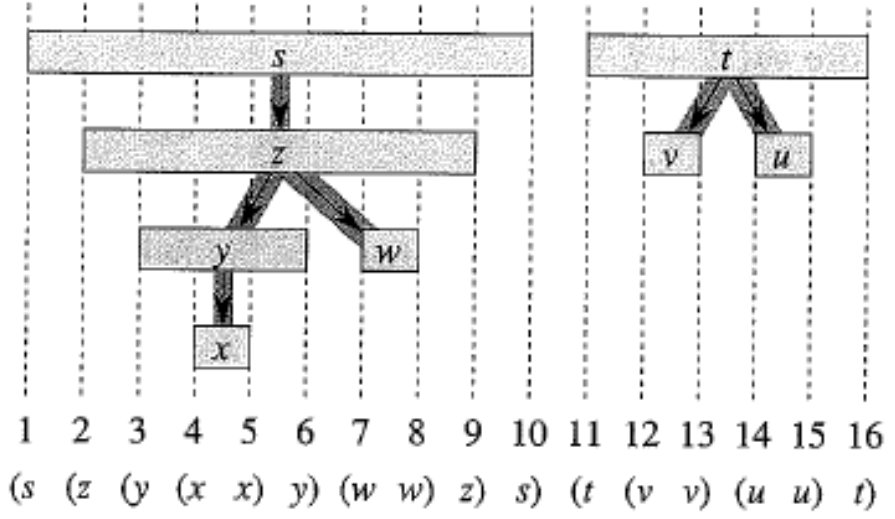
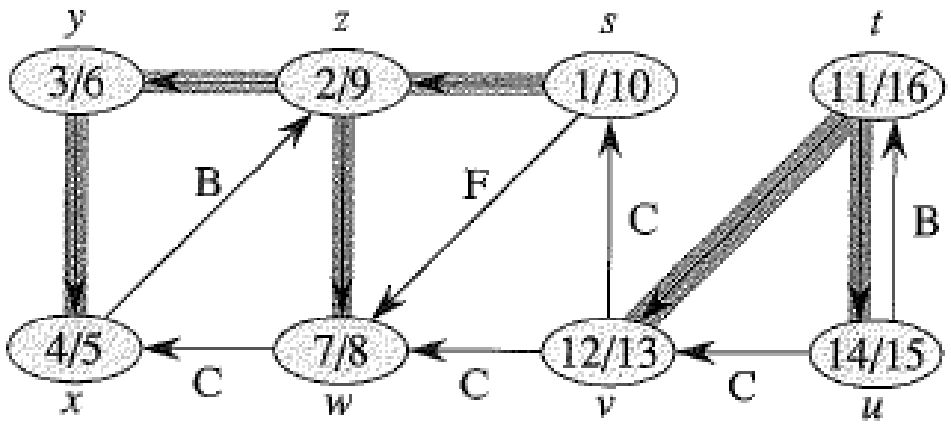
// white vertex  $u$  has just been discovered

// explore edge ( $u, v$ )

// blacken  $u$ ; it is finished

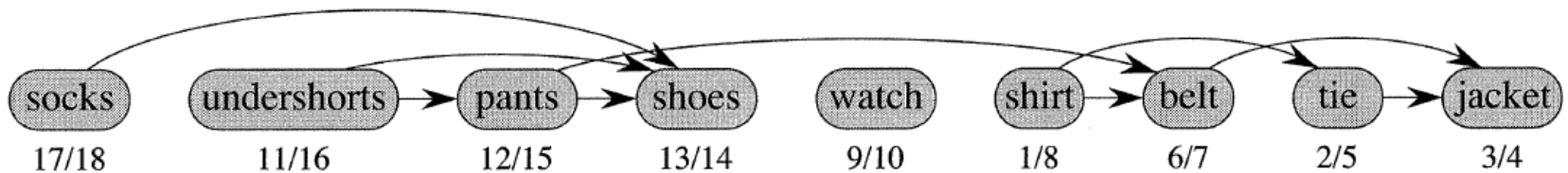
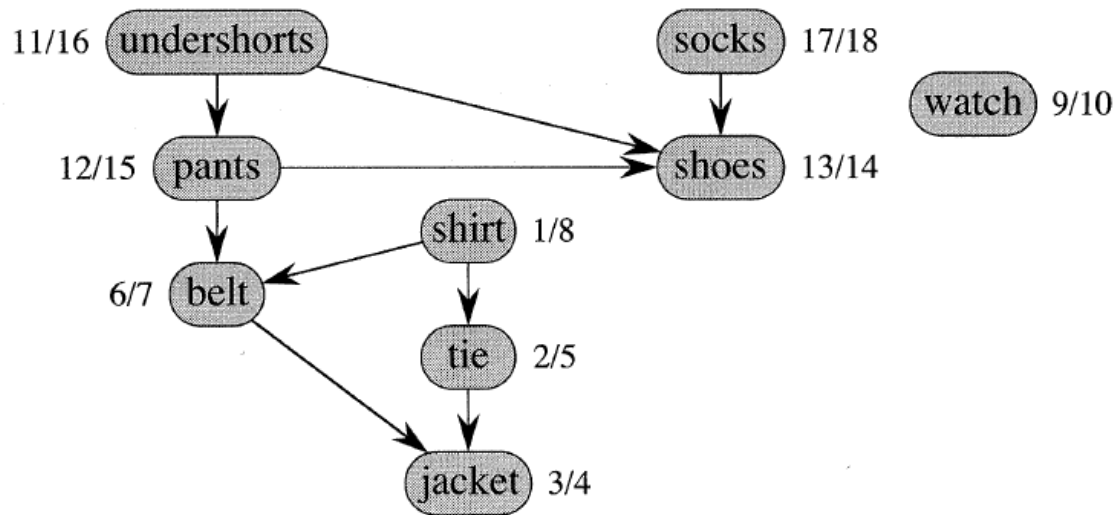
Tid  $O(n+m)$





**█** = træ-kanter  
**B** = tilbage-kanter  
**C** = kryds-kanter  
**F** = fremad-kanter

# Acykliske Grafer: Topologisk Sortering



**Alle kanter går fra venstre-mod-højre**

Microsoft Excel - Copy of SheepFlock

File Edit View Insert Format Tools Data Window Help Adobe PDF

Type a question for help

H18 fx =B18\*G18

	A	B	C	D	E	F	G	H	I
3	<b>I. Description of animals in flock during the year.</b>								
4	Ewes in flock:	700			[Green cells are those you can change.]				
5	Lambing rate:	4	times per	3	years =	1.33	times/year.		
6	Lambs weaned/lambing:	1.5	Days of lactation/lambing:		60				
7	Adult death loss per year:	3%	Days in lactation/year:		80				
8	Postweaning lamb loss:	2%	Lambs weaned per ewe per year:			2.0			
9	Ewe culling rate:	15%	Ram culling rate:		50%				
10	Rams/100 ewes:	1	(Only 1/3 of ewes bred per season under STAR system.)						Inventory
11			Weaning	Market	Final	Price	Value	or sale	
12		Number	wt, lb	wt, lb	wt, lb	\$/lb	per head	value	
13	Ewes	700			150	\$1.00	\$150	\$105,000	
14	Rams	8			200	\$2.00	\$400	\$3,200	
15	Ewe lamb rplcmnts	126	30		100	\$1.25	\$125	\$15,750	
16	Ram lamb rplcmnts	5	40		130	\$2.00	\$260	\$1,300	
17	Ewe lambs sold	560	30	70		\$1.10	\$77	\$43,120	
18	Ram lambs sold	681	40	70		\$1.10	\$77	\$52,437	
19	Cull ewes sold	105		150		\$0.30	\$45	\$4,725	
20	Cull rams sold	5		200		\$0.30	\$60	\$300	
21	Fleece weight per adult	708			6	\$0.30	\$1.80	\$1,274	
22							Inventory:	\$125,250	
23							Sales:	\$101,856	

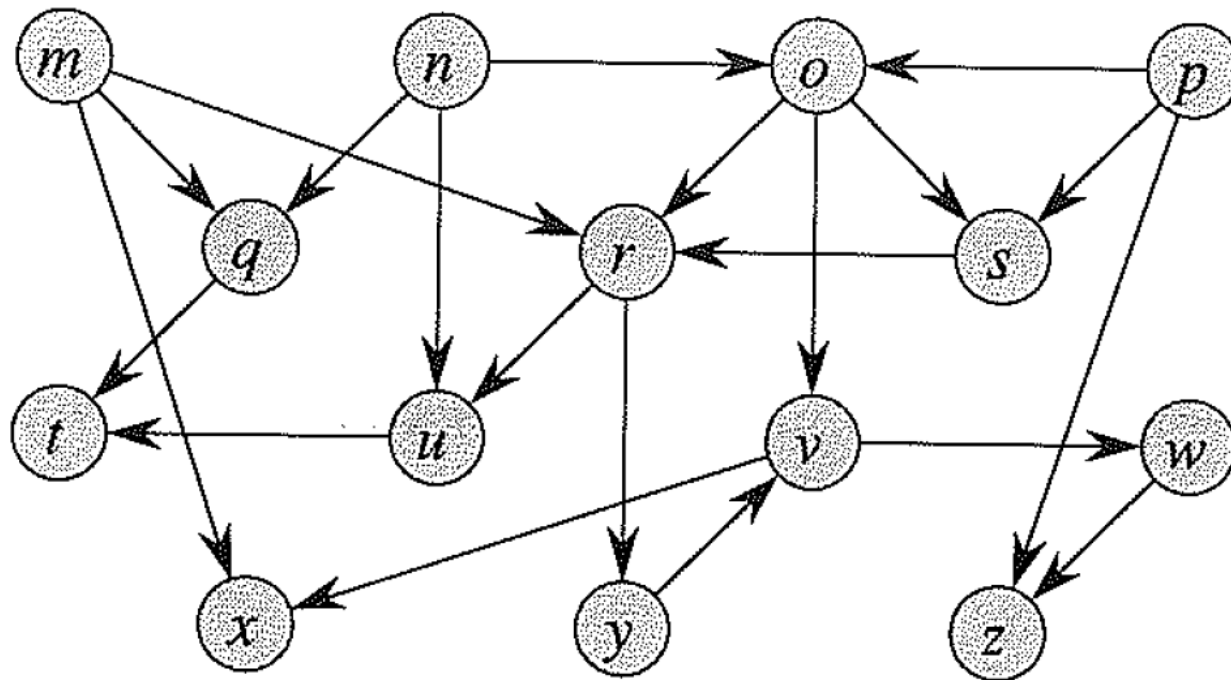
Sheep flock /

Ready

Topologisk sortering = en rækkefølge hvor vi kan beregne cellernes indhold



# Topologisk Sortering (I)

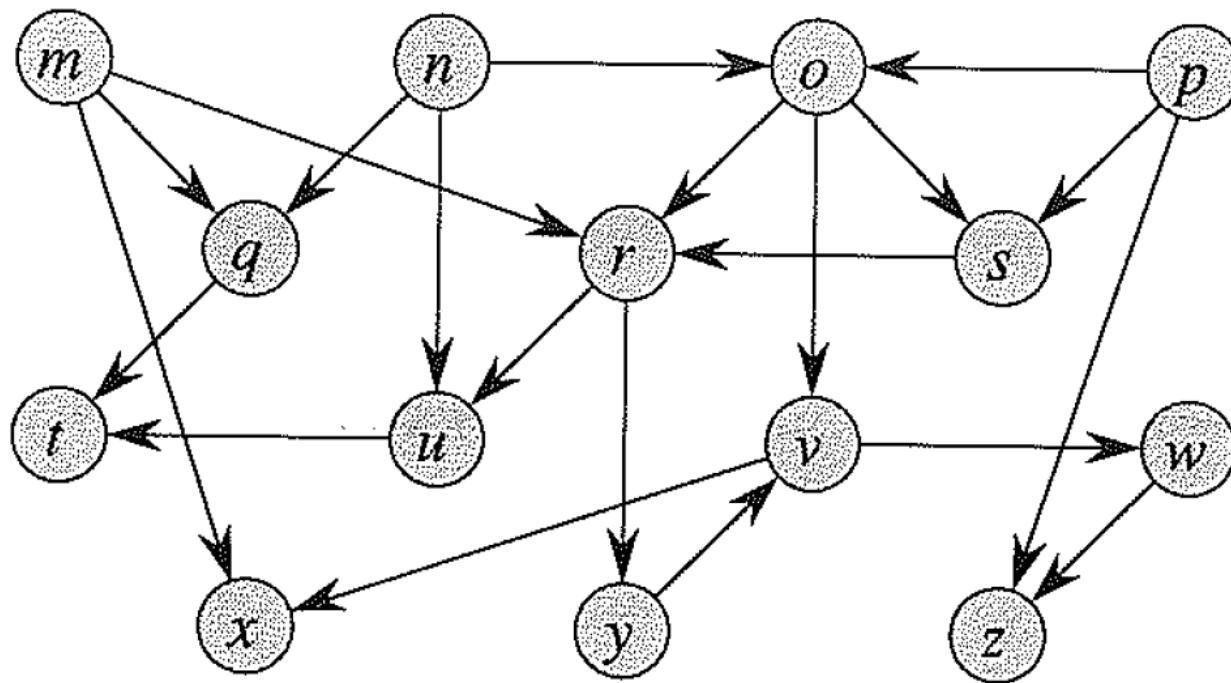


**Algoritme:** Grådigt slet en knude med indgrad 0 (og udgående kanter), og tilføj knuden sidst i den topologiske orden

Tid  $O(m+n)$



# Topologisk Sortering (II)



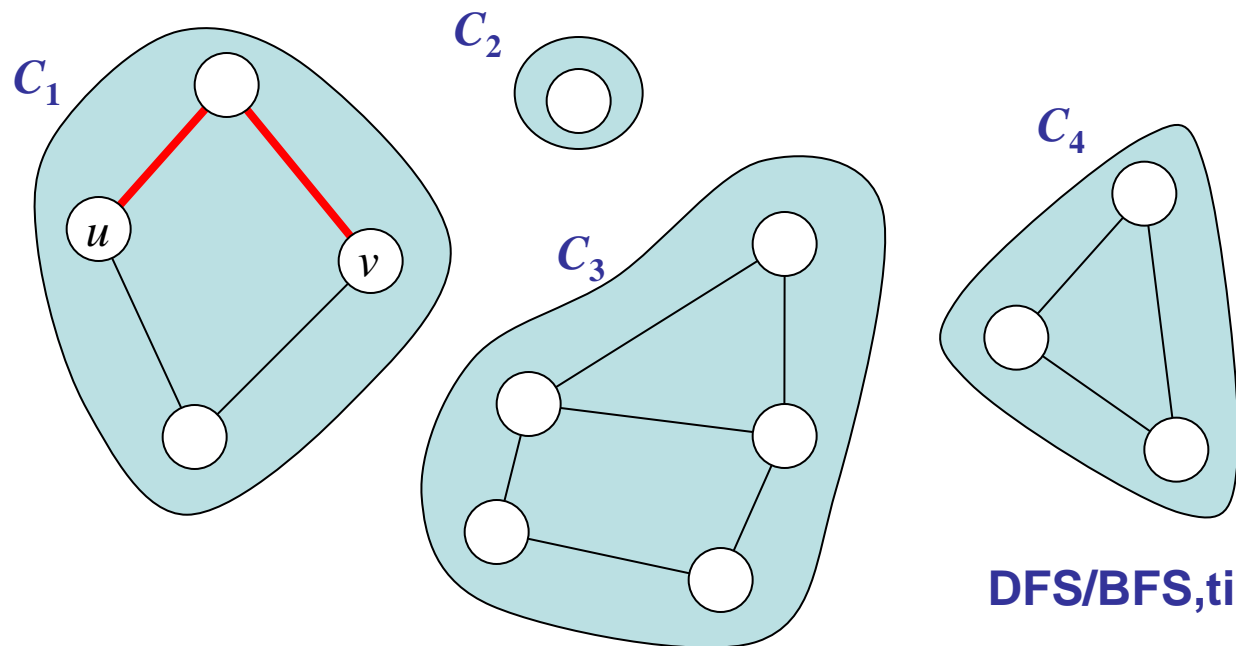
TOPOLOGICAL-SORT( $G$ )

- 1 call DFS( $G$ ) to compute finishing times  $v.f$  for each vertex  $v$
- 2 as each vertex is finished, insert it onto the front of a linked list
- 3 **return** the linked list of vertices

Tid  $O(m+n)$

# Sammenhængskomponenter

Opdeling af knuderne i en **uorienteret** graf i **komponenter**  $C_1, \dots, C_k$ , således at  $u$  og  $v$  er i  $C_i$  hvis og kun hvis der er en **sti** mellem  $u$  og  $v$



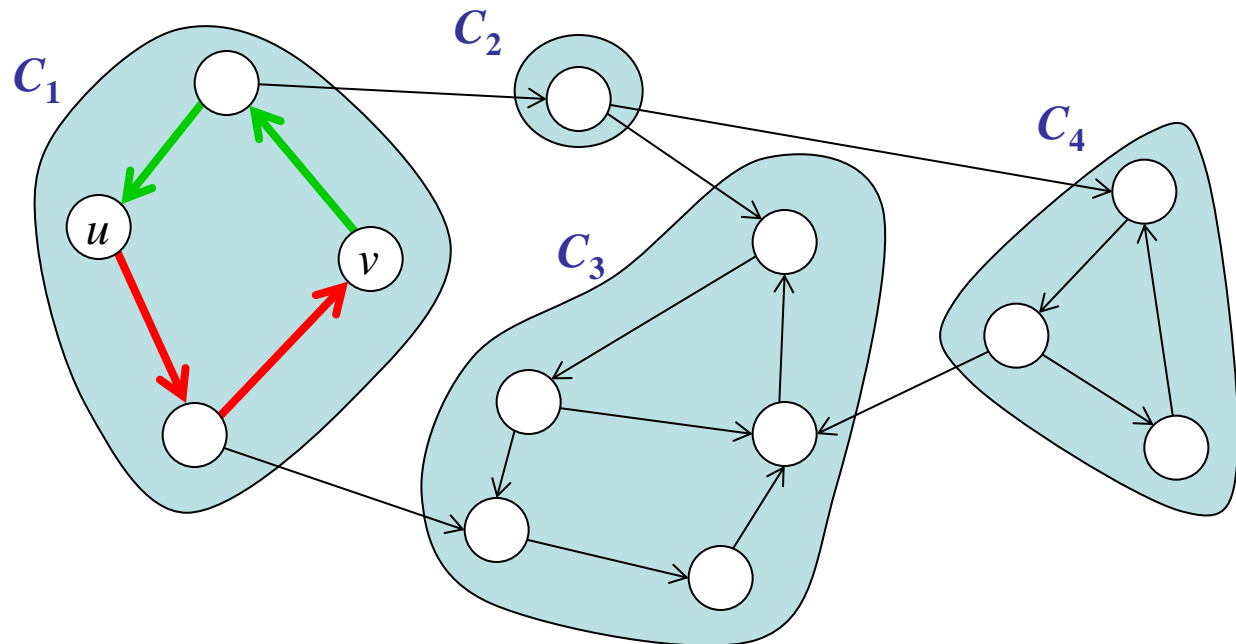
DFS/BFS, tid  $O(m+n)$

# Stærke Sammenhængskomponenter

Opdeling af knuderne i en **orienteret** graf i **komponenter**  $C_1, \dots, C_k$ , således at

$u$  og  $v$  er i  $C_i$  hvis og kun hvis der både er

- en **sti fra  $u$  til  $v$**  og
- en **sti fra  $v$  til  $u$**



# Stærke Sammenhængskomponenter

## STRONGLY-CONNECTED-COMPONENTS ( $G$ )

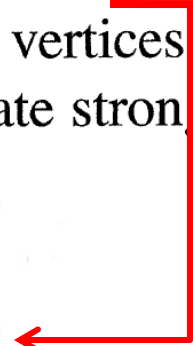
- 1 call DFS( $G$ ) to compute finishing times  $u.f$  for each vertex  $u$
- 2 compute  $G^T$
- 3 call DFS( $G^T$ ), but in the main loop of DFS, consider the vertices in order of decreasing  $u.f$  (as computed in line 1)
- 4 output the vertices of each tree in the depth-first forest formed in line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

DFS( $G$ )

```

1 for each vertex  $u \in G.V$ 
2    $u.color = WHITE$ 
3    $u.\pi = NIL$ 
4  $time = 0$ 
5 for each vertex  $u \in G.V$ 
6   if  $u.color == WHITE$ 
7     DFS-VISIT( $G, u$ )

```



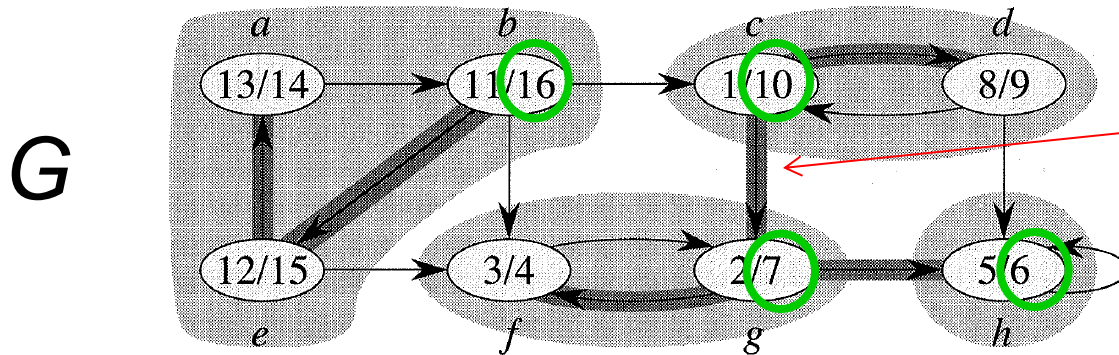
DFS-VISIT( $G, u$ )

```

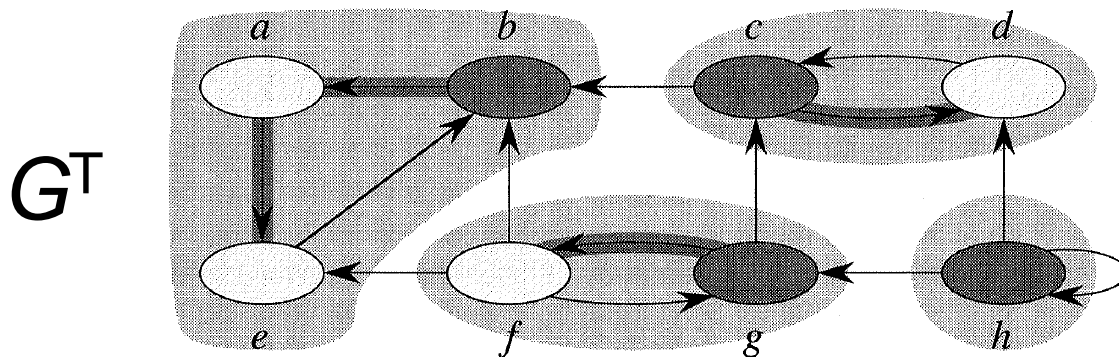
1  $time = time + 1$            // white vertex  $u$  has just been discovered
2  $u.d = time$ 
3  $u.color = GRAY$ 
4 for each  $v \in G.Adj[u]$      // explore edge  $(u, v)$ 
5   if  $v.color == WHITE$ 
6      $v.\pi = u$ 
7     DFS-VISIT( $G, v$ )
8  $u.color = BLACK$          // blacken  $u$ ; it is finished
9  $time = time + 1$ 
10  $u.f = time$ 

```

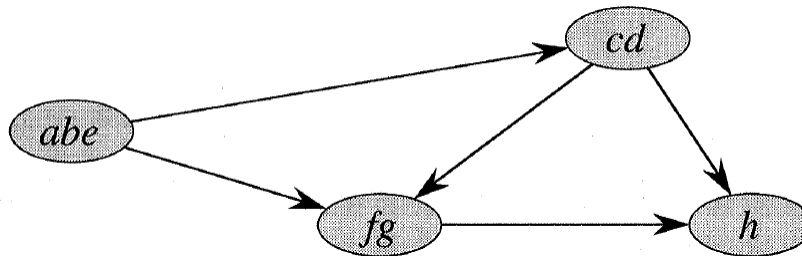
# Stærke Sammenhængskomponenter



DFS trækanter mellem to stærke sammenhængskomponenter



De største finishing tider i hver komponent udgør en (omvendt) topologisk sortering af komponenterne



Tid  $O(m+n)$