

Algoritmer og Datastrukturer 2

Gerth Stølting Brodal

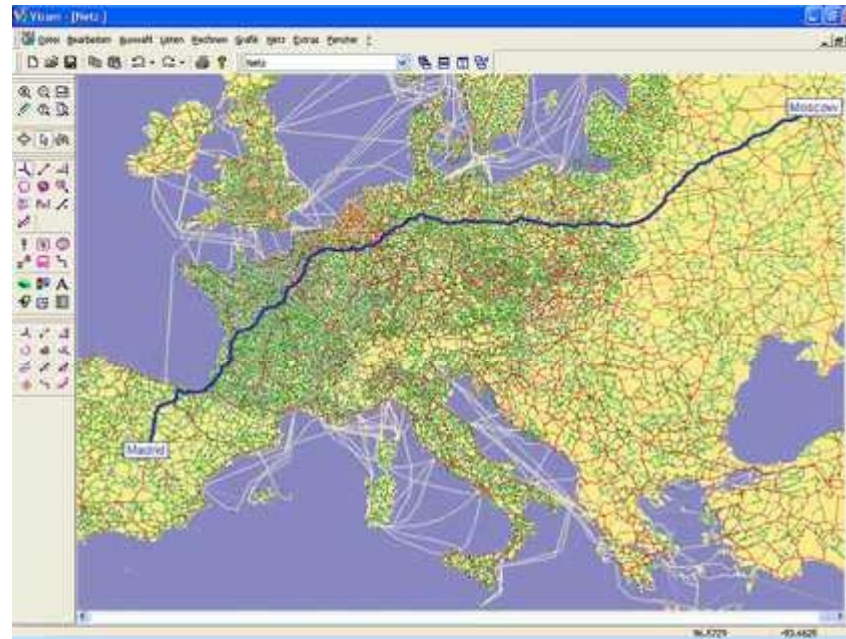
Korteste Veje
[CLRS, kapitel 24]



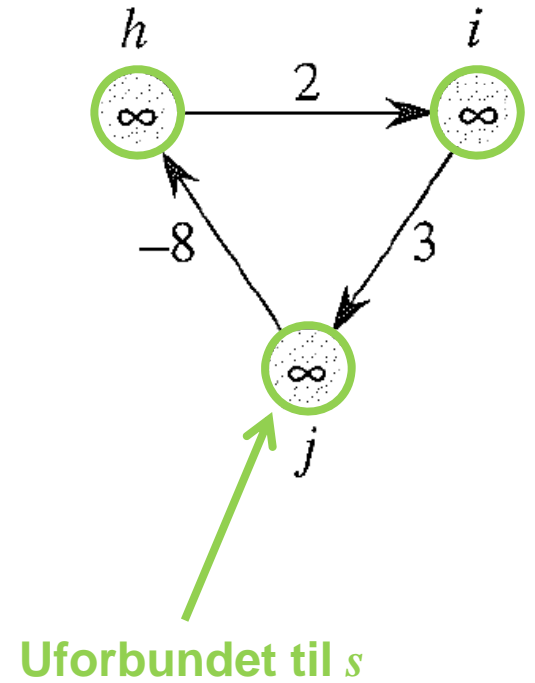
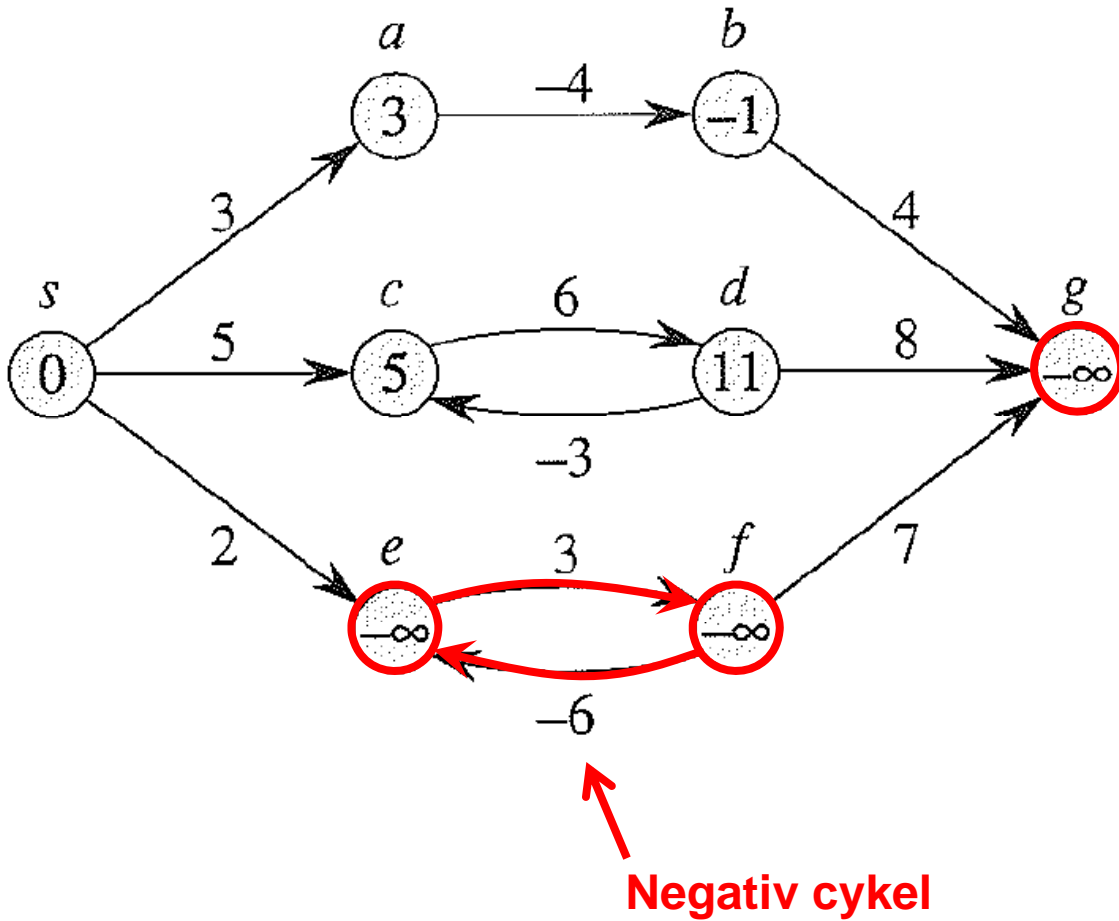
AARHUS UNIVERSITET

Kort over Vest-Europa

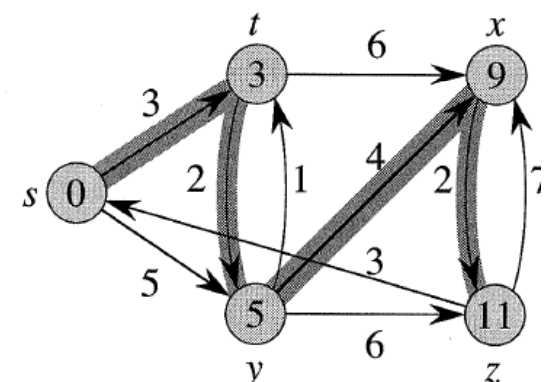
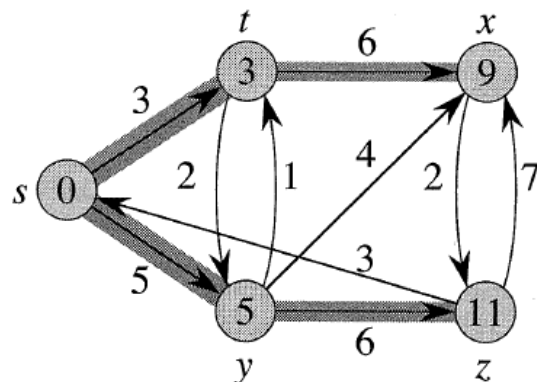
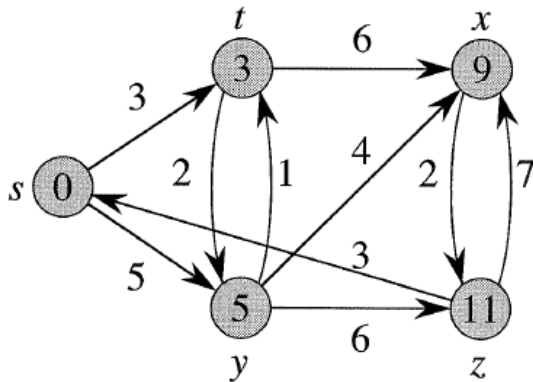
- 18.029.721 knuder
- 42.199.587 orienterede kanter



Eksempel: Korteste veje fra s



Eksempel: Korteste veje træer

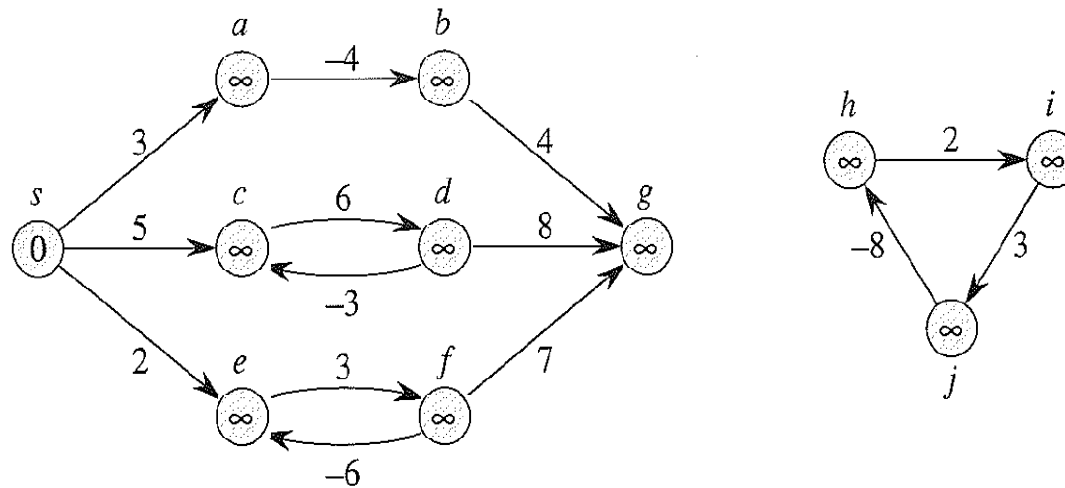


2 forskellige korteste veje træer der repræsenterer stier fra s med samme længde

Korteste Veje Estimator : Initialisering

INITIALIZE-SINGLE-SOURCE(G, s)

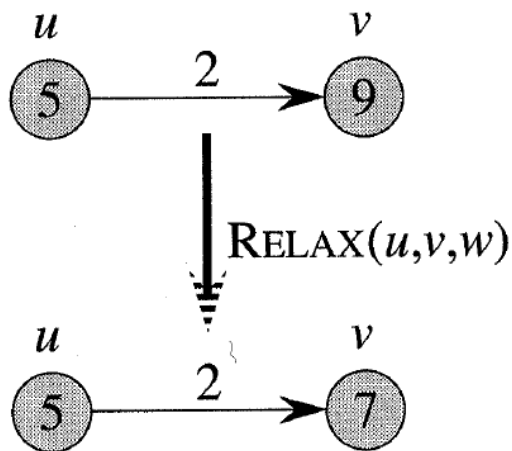
- 1 **for** each vertex $v \in G.V$
- 2 $v.d = \infty$
- 3 $v.\pi = \text{NIL}$
- 4 $s.d = 0$



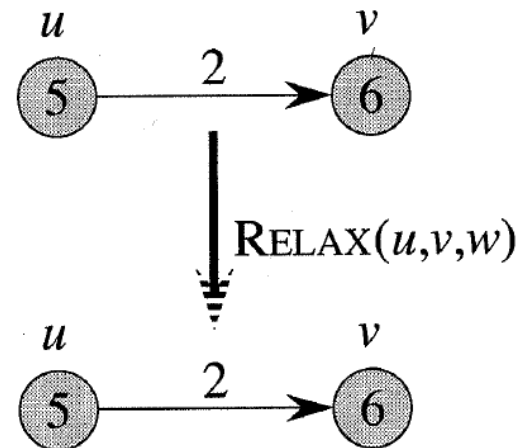
Korteste Veje Estimer : Relax

RELAX(u, v, w)

- 1 **if** $v.d > u.d + w(u, v)$
- 2 $v.d = u.d + w(u, v)$
- 3 $v.\pi = u$



**Kortere afstand
til v fundet**



**Forbedrer ikke
afstanden til v**

Bellman-Ford:

Korteste Veje i Grafer med Negative Vægte

BELLMAN-FORD(G, w, s)

1 INITIALIZE-SINGLE-SOURCE(G, s)

2 **for** $i = 1$ **to** $|G.V| - 1$

3 **for** each edge $(u, v) \in G.E$

4 RELAX(u, v, w)

5 **for** each edge $(u, v) \in G.E$

6 **if** $v.d > u.d + w(u, v)$

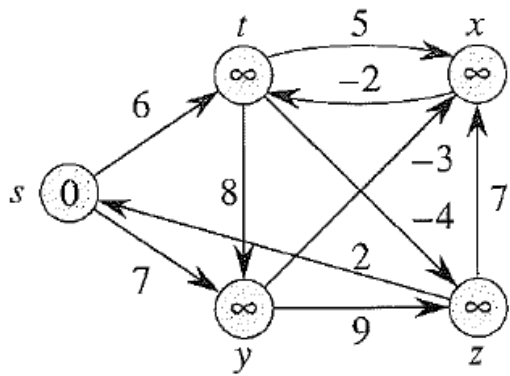
7 **return** FALSE

8 **return** TRUE

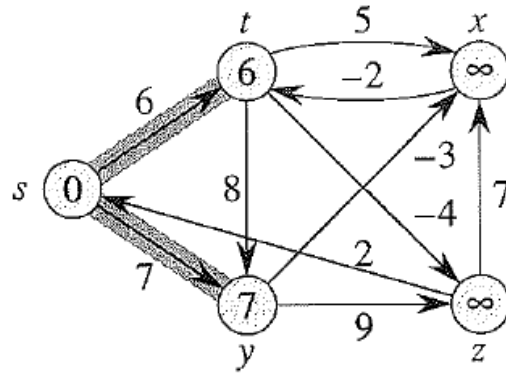
Check for
negativ
cykel

Tid $O(nm)$

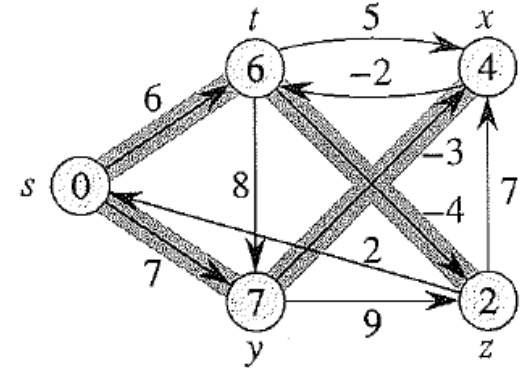
Bellman-Ford: Eksempel



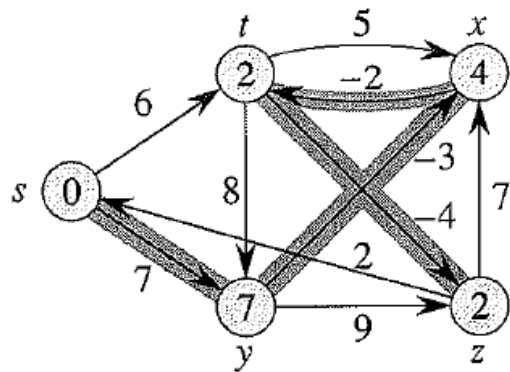
(a)



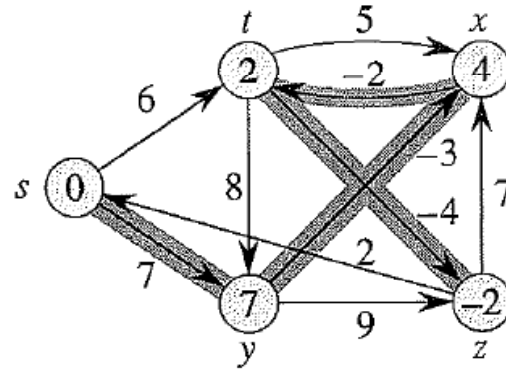
(b)



(c)



(d)



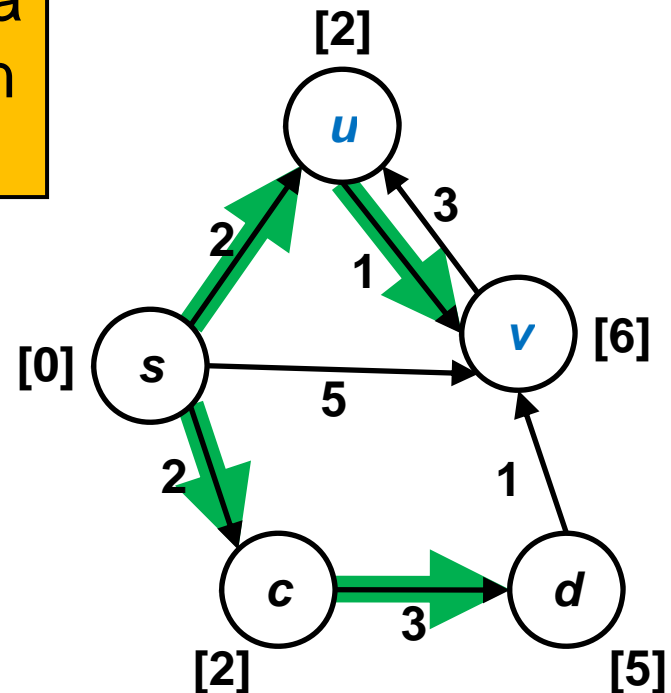
(e)

Sætning

Betragt et (ukendt) korteste veje træ T hvori (u,v) er en kant.

Antag den aktuelle $d[u]$ er den korteste afstand til u .

$\text{Relax}(u,v,w)$ medfører at $d[v]$ også er en kortest afstand til v (hvis den ikke allerede var det).



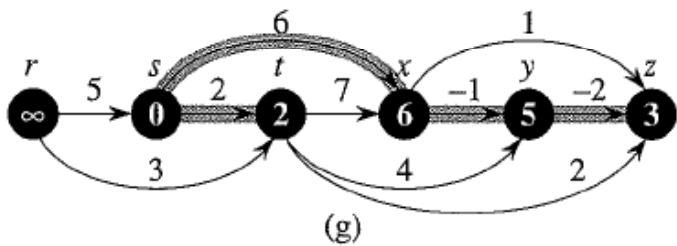
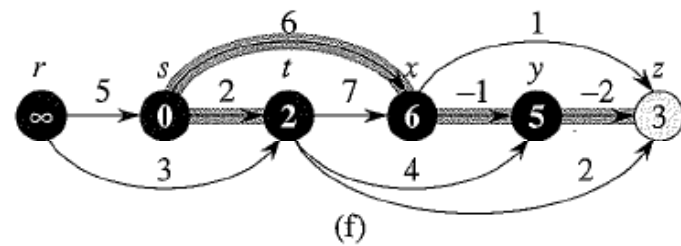
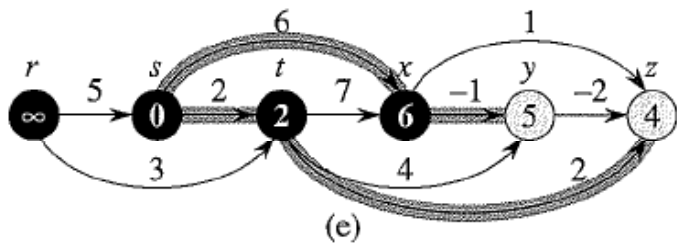
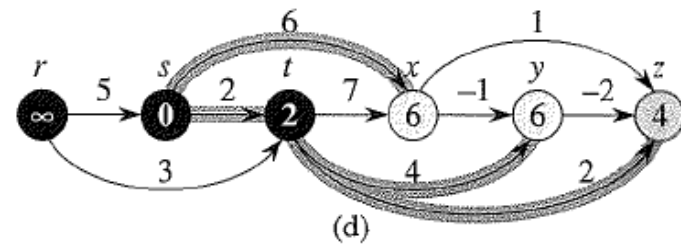
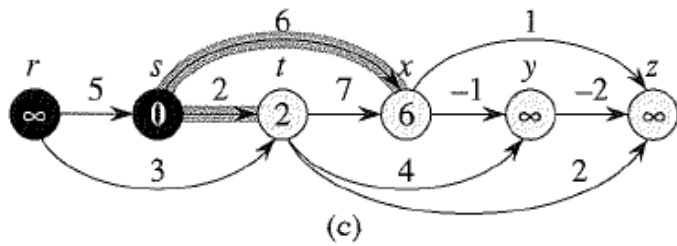
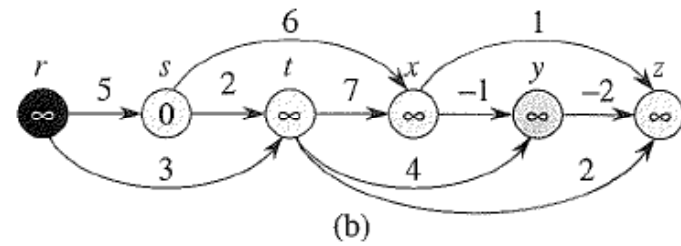
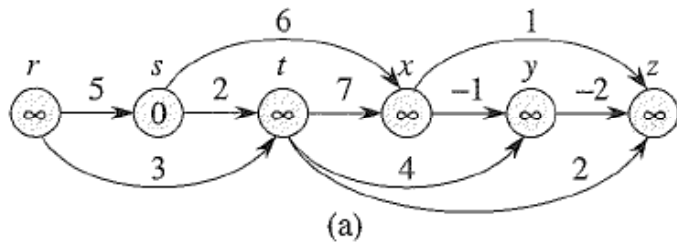
Korteste Veje i Acycliske Grafer

DAG-SHORTEST-PATHS(G, w, s)

- 1 topologically sort the vertices of G
- 2 INITIALIZE-SINGLE-SOURCE(G, s)
- 3 **for** each vertex u , taken in topologically sorted order
- 4 **for** each vertex $v \in G.Adj[u]$
- 5 RELAX(u, v, w)

Tid $O(n+m)$

Acykliske Grafer : Eksempel



Dijkstra:

Korteste Veje i Grafer uden Negative Vægte

DIJKSTRA(G, w, s)

1 INITIALIZE-SINGLE-SOURCE(G, s)

2 $S = \emptyset$

3 $Q = G.V$

$Q =$ prioritets kø (besøger knuderne
efter stigende afstand fra s)

4 **while** $Q \neq \emptyset$

5 $u = \text{EXTRACT-MIN}(Q)$

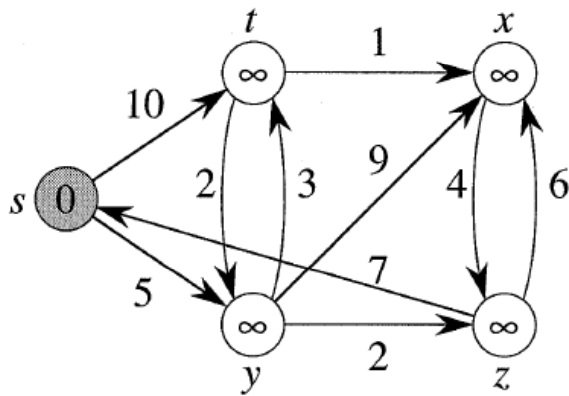
6 $S = S \cup \{u\}$

7 **for** each vertex $v \in G.Adj[u]$

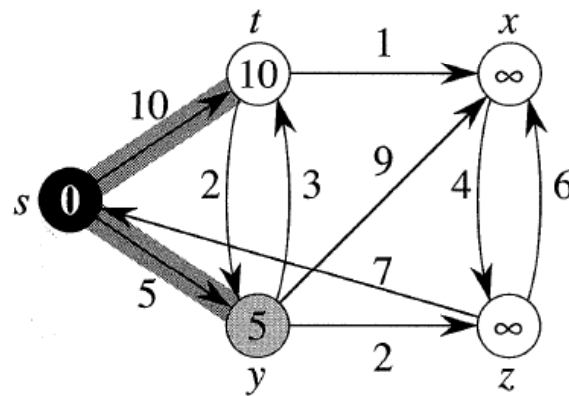
8 RELAX(u, v, w)

Tid $O((n+m) \cdot \log n)$
eller $O(n^2+m)$

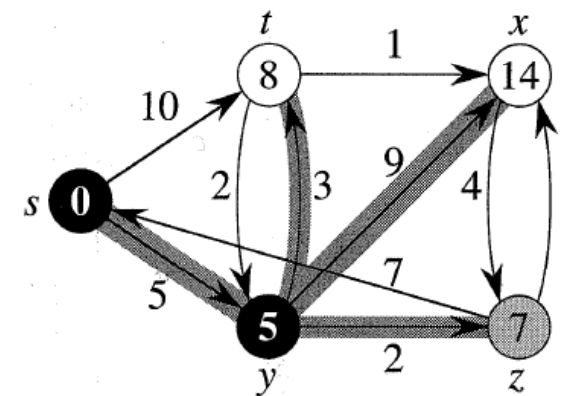
Dijkstra : Eksempel



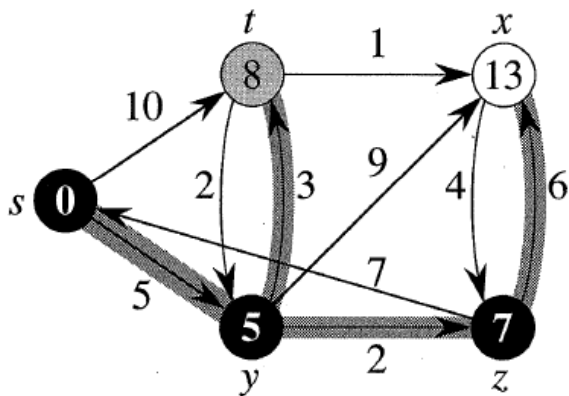
(a)



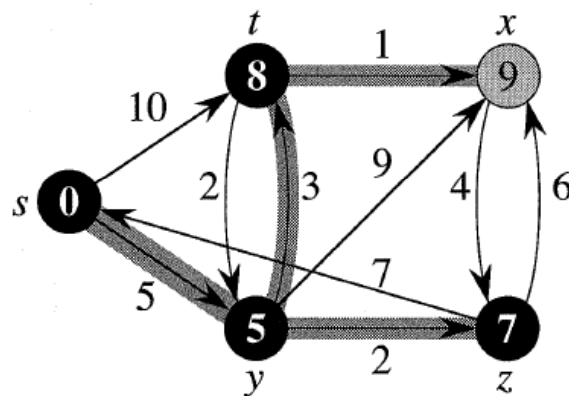
(b)



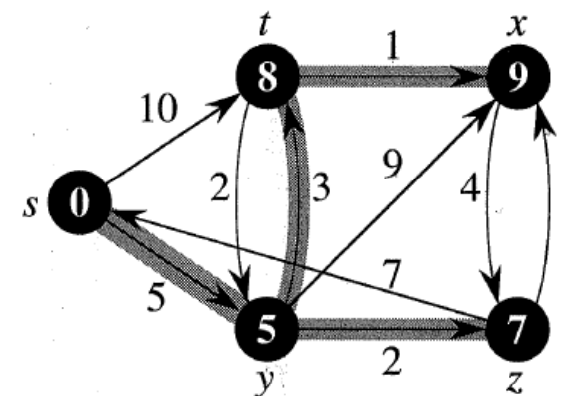
(c)



(d)



(e)



(f)

Opsummering

		SSSP En-til-alle korteste veje
Acykliske grafer (positive og negative vægte)		$O(n+m)$
Generelle grafer	Kun positive vægte	Dijkstra $O((n+m) \cdot \log n)$ $O(n^2+m)$
	Positive og negative vægte	Bellman-Ford $O(m \cdot n)$

Relaxer hver kant præcis én gang