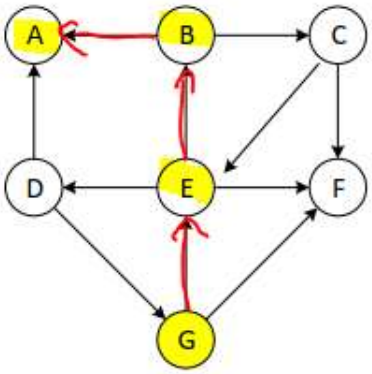
HW6

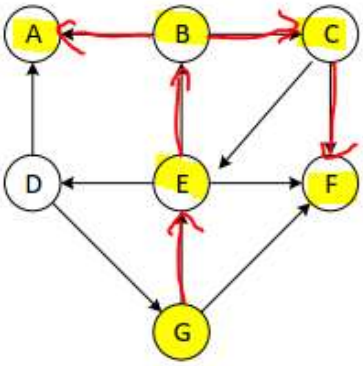
Problem 1

Sequence: G, E, B, A, C, F, D.

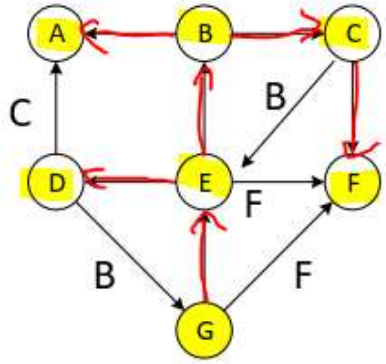
1. Start at vertex G.
2. G -> E -> B -> A.
   Backtrack to B.



3. B -> C -> F.
   Backtrack to E (F -> C -> B -> E).



4. E -> D.
   Finished.



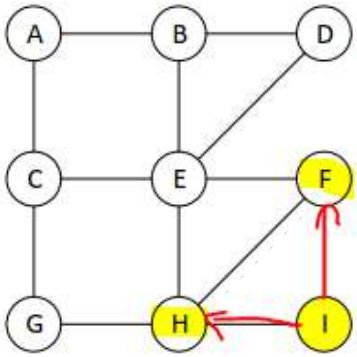
Tree edge = the red edges are the tree edges (the red arrows).
Forward edge = F.
Back edge = B.
Cross edge = C.

Kasperzack Stefan

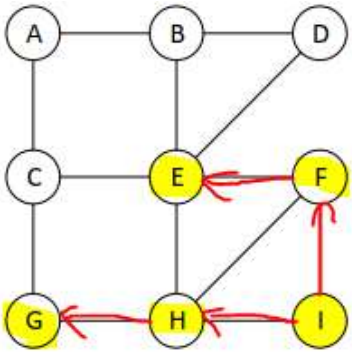
Problem 2

Sequence: I, F, H, E, G, B, C, D, A.

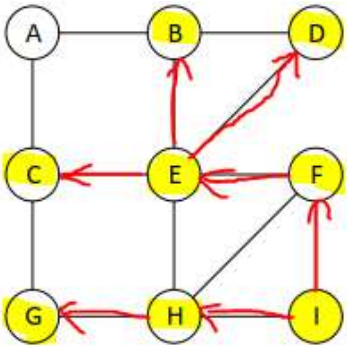
1. Start at vertex I.
2. Explore vertices that are one-edge away from I: F, H.



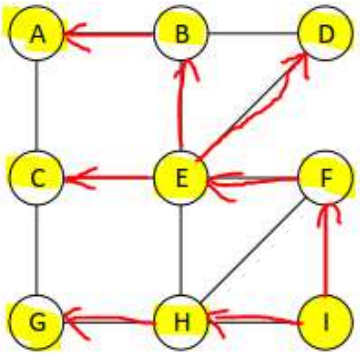
3. Explore vertices that are two-edge away from I: E, G.



4. Explore vertices that are three-edge away from I: B, C, D.



5. Explore vertices that are four-edge away from I: A.
   Finished.



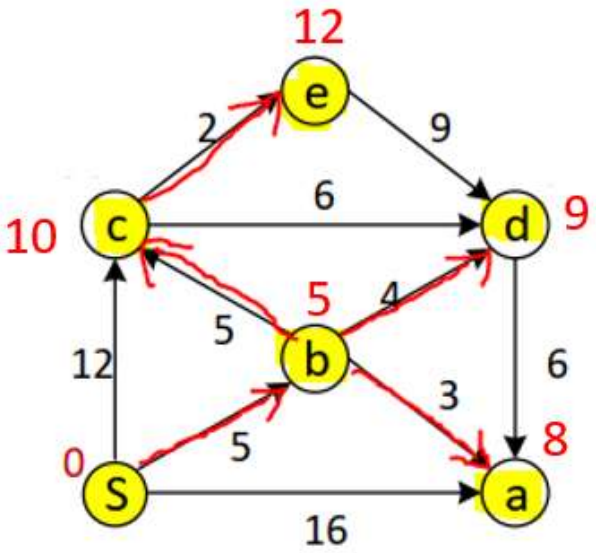
Problem 3

Problem 3-(1)

Initial	Vertex	s	a	b	c	d	e		comes into cloud
	D	0	∞	∞	∞	∞	∞		-
1. iteration	Vertex	s	a	b	c	d	e		s
	D	0	16	5	12	∞	∞		
2. iteration	Vertex	s	a	b	c	d	e		b
	D	0	8	5	10	9	∞		
3. iteration	Vertex	s	a	b	c	d	e		a
	D	0	8	5	10	9	∞		
4. iteration	Vertex	s	a	b	c	d	e		d
	D	0	8	5	10	9	∞		
5. iteration	Vertex	s	a	b	c	d	e		c
	D	0	8	5	10	9	12		
6. iteration	Vertex	s	a	b	c	d	e		e
	D	0	8	5	10	9	12		

Problem 3-(2)

In red: shortest path from S to every other node generated by the algorithm.



6. iteration	Vertex	s	a	b	c	d	e
	D	0	8	5	10	9	12

Kasperzack Stefan

Problem 4

Problem 4-(1)

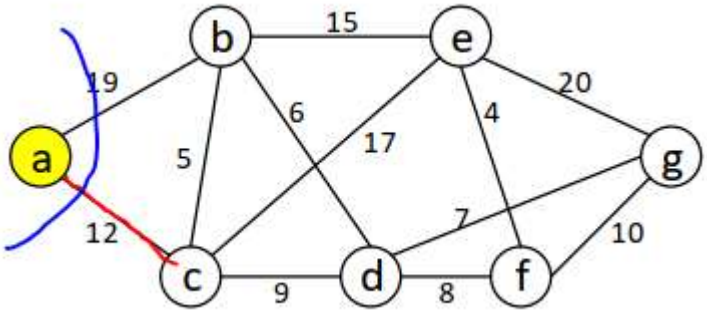
Sequence: a, c, b, d, g, f, e.

Red: min-weight edge.

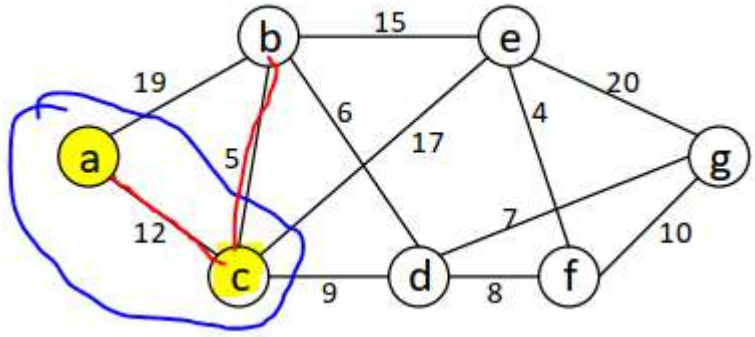
Yellow: in the cloud.

Blue: cloud boundary.

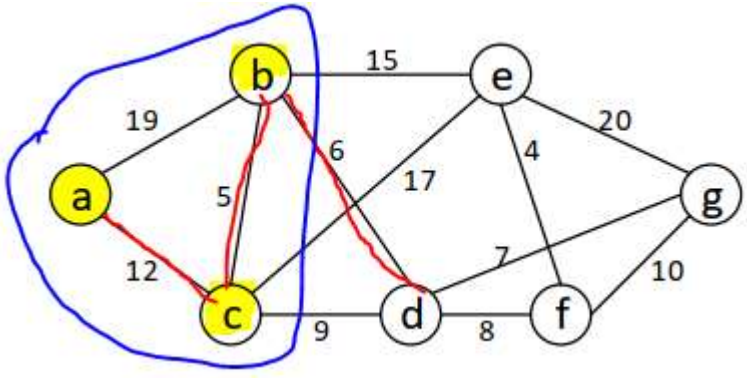
1. Brought into the cloud: a.



2. Brought into the cloud: c.

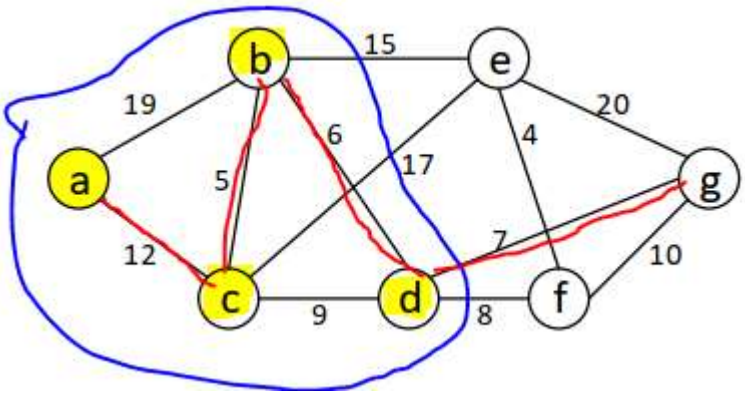


3. Brought into the cloud: b.

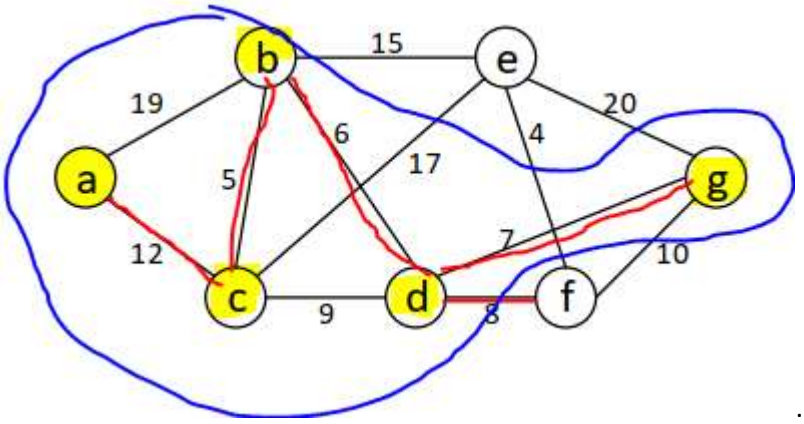


.

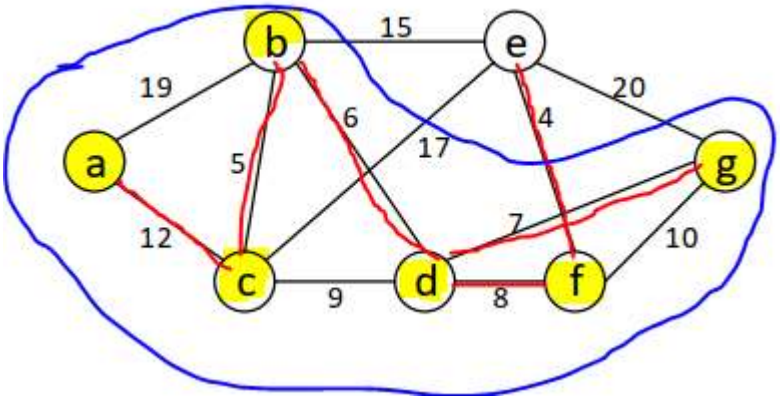
4. Brought into the cloud: d.



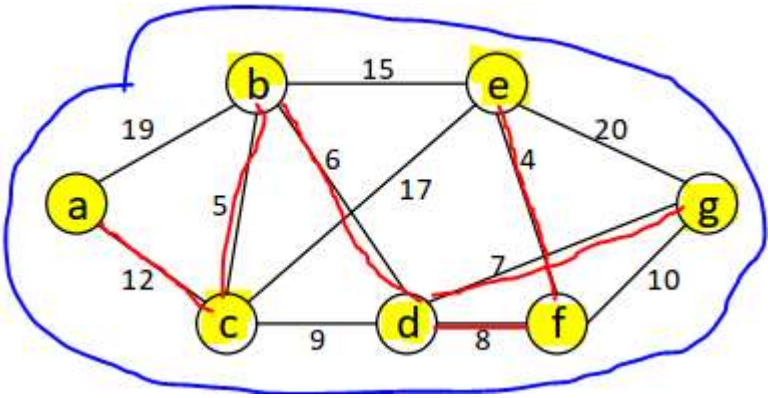
5. Brought into the cloud: g.



6. Brought into the cloud: f.



7. Brought into the cloud: e.



Problem 4-(2)

Minimum spanning tree, set of edges: {(a, c), (c, b), (b, d), (d, g), (d, f), (f, e)}.

Red: min-spanning tree.