Answer for Assignment 1

Q1:

1. BFS/DFS

   A: False if it uses first-in-first-out rule. As it should first visit G before F. True if use the layer by layer definition.

   B: False if it uses first-in-first-out rule. As it should first visit F before G. True if use the layer by layer definition.

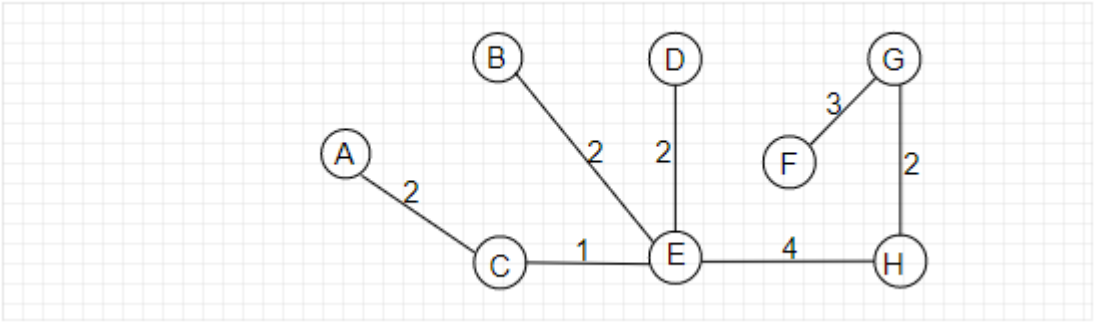
   C: True. It follows the DFS order. False if it uses the first-in-last-out rule.

   D: False. When visiting D, D still has unvisited neighbor G. However, after D, it directly jumps to F.

2. minimum spanning tree

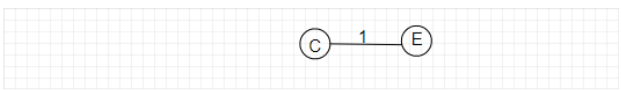
   A: Prime Algorithm

A | AC | ACE | ACEBD | ACEBDH | ACEBDHG | ACEBDHGF

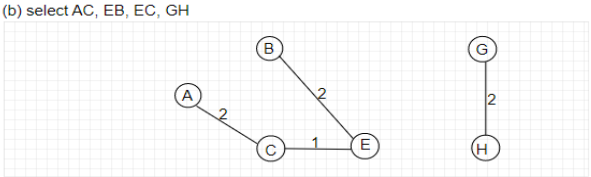


   B: Kruskal's Algorithm

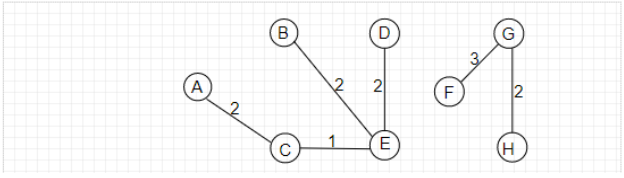
(a) select CE



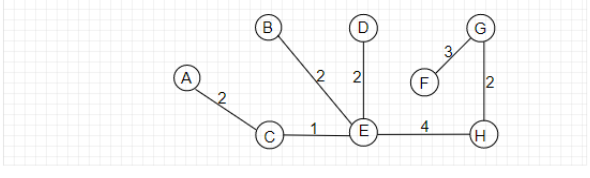
(b) select AC, EB, EC, GH



(c) select GF



(d) select EH



3. Dijkstra's algorithm

1: from D, there are B, E,G

Vertex	Visited	Distance	Previous
A	F	∞	Ø
B	F	5	D
C	F	∞	Ø
D	T	0	Ø
E	F	2	D
F	F	∞	Ø
G	F	8	D
H	F	∞	Ø

2: select E, there are B, C, F,H unvisited

Vertex	Visited	Distance	Previous
A	F	∞	Ø
B	F	4	E
C	F	3	E
D	T	0	Ø
E	T	2	D
F	F	7	E
G	F	8	D
H	F	6	E

3: select C, there are A, B unvisited

Vertex	Visited	Distance	Previous
A	F	5	C
B	F	4	E
C	T	3	E
D	T	0	Ø
E	T	2	D
F	F	7	E
G	F	8	D
H	F	6	E

4: select B, there is A unvisited

Vertex	Visited	Distance	Previous
A	F	5	C
B	T	4	E
C	T	3	E
D	T	0	Ø
E	T	2	D
F	F	7	E
G	F	8	D
H	F	6	E

5: select A

Vertex	Visited	Distance	Previous
A	T	5	C
B	T	4	E
C	T	3	E
D	T	0	Ø
E	T	2	D
F	F	7	E
G	F	8	D
H	F	6	E

6: select H, there is G unvisited

Vertex	Visited	Distance	Previous
A	T	5	C
B	T	4	E
C	T	3	E
D	T	0	Ø
E	T	2	D
F	F	7	E
G	F	8	D
H	T	6	E

7: select F, there is G unvisited

Vertex	Visited	Distance	Previous
A	T	5	C
B	T	4	E
C	T	3	E
D	T	0	Ø
E	T	2	D
F	T	7	E
G	F	8	D
H	T	6	E

8: select G

Vertex	Visited	Distance	Previous
A	T	5	C
B	T	4	E
C	T	3	E
D	T	0	Ø
E	T	2	D
F	T	7	E
G	T	8	D
H	T	6	E

Q2:

1. transitive closure

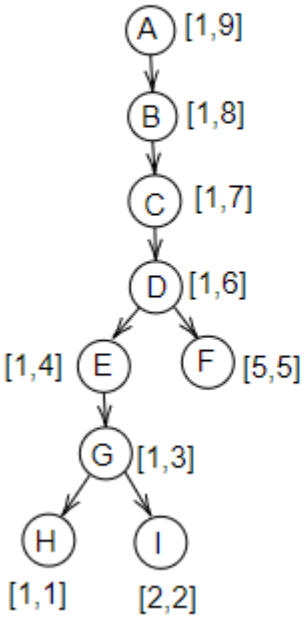
empty means zero.

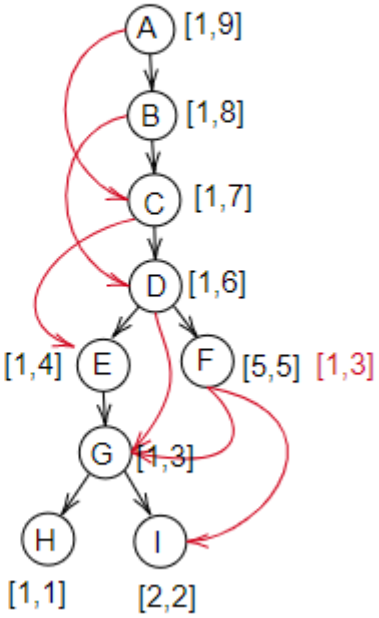
	A	B	C	D	E	F	G	H	I
A	1	1	1	1	1	1	1	1	1
B		1	1	1	1	1	1	1	1
C			1	1	1	1	1	1	1
D				1	1	1	1	1	1
E					1		1	1	1
F						1	1	1	1
G							1	1	1
H								1	
I									1

(A, F)√, (C, G)√, (E, F)×

2. "optimal" spanning tree

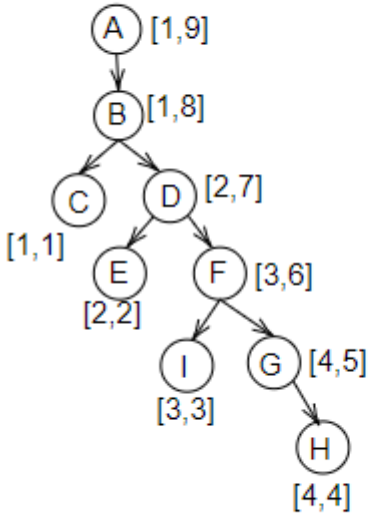
traversing vertices in post order:
H → I → G → E →F → D → C → B → A.

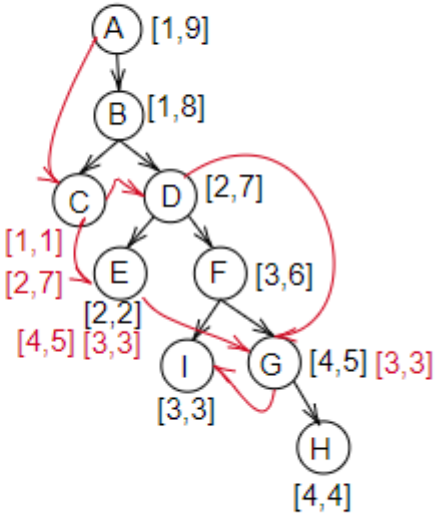




3. random spanning tree

traversing vertices in post order:
C → E → I → H → G → F → D → B →A.





4. which tree is the better index

A:The optimal tree cover is the tree cover such that the resulting compression scheme has the least storage cost (number of intervals needed). The optimal tree cover is better as it has less intervals.

B: If you connect the adjacent intervals, like [2,2] [3,3],[4,5] into [2,5], then the random spanning tree has less intervals. The random spanning tree is better.

Either is OK

5. total-order-based 2-hop cover

the degree of each vertex:

	A	B	C	D	E	F	G	H	I
	2	3	4	5	3	3	5	1	2

1: choose vertex D:

	A	B	C	D	E	F	G	H	I
L_in				D	D	D	D	D	D
L_out	D	D	D	D					

2: choose vertex G:
(A,G), (B, G), (C, G) are covered by D

	A	B	C	D	E	F	G	H	I
L_in				D	D	D	D, G	D,G	D,G
L_out	D	D	D	D	G	G	G		

3: choose vertex C:
(C,E), (C, F), (C, H), (C, I) are covered by D

	A	B	C	D	E	F	G	H	I
L_in			C	D	D	D	D, G	D,G	D,G
L_out	D,C	D,C	D,C	D	G	G	G		

4: choose vertex B:
(B,E), (B, F), (B, H), (B, I) are covered by D

	A	B	C	D	E	F	G	H	I
L_in		B	C	D	D	D	D, G	D,G	D,G
L_out	D,C,B	D,C,B	D,C	D	G	G	G		

5: choose vertex E:
(A,E), (E, H), (E, I) are covered by D, G

	A	B	C	D	E	F	G	H	I
L_in		B	C	D	D,E	D	D, G	D,G	D,G
L_out	D,C,B	D,C,B	D,C	D	G,E	G	G		

6: choose vertex F:
(A,F), (F, H), (F, I) are covered by D, G

	A	B	C	D	E	F	G	H	I
L_in		B	C	D	D,E	D,F	D, G	D,G	D,G
L_out	D,C,B	D,C,B	D,C	D	G,E	G,F	G		

7: choose vertex A:
(A, H), (A, I) are covered by D

	A	B	C	D	E	F	G	H	I
L_in	A	B	C	D	D,E	D,F	D, G	D,G	D,G
L_out	D,C,B,A	D,C,B	D,C	D	G,E	G,F	G		

8: choose vertex I:

	A	B	C	D	E	F	G	H	I
L_in	A	B	C	D	D,E	D,F	D, G	D,G	D,G,I
L_out	D,C,B,A	D,C,B	D,C	D	G,E	G,F	G		I

9: choose vertex H:

	A	B	C	D	E	F	G	H	I
L_in	A	B	C	D	D,E	D,F	D, G	D,G,H	D,G,I
L_out	D,C,B,A	D,C,B	D,C	D	G,E	G,F	G	H	I