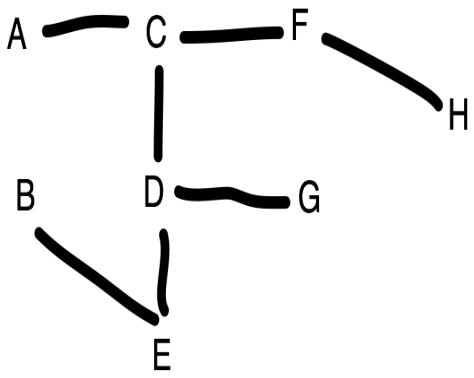
Part 1 – Theory

Problem 1:

(A, C):2 OK
(D, G):2 OK
(D, E):3 OK
(F, H):5 OK
(C, D):7 OK
(C, F):7 OK
(B, E):8 OK
(F, G):8 Reject
(C, G):9 Reject
(B, D):10 Reject

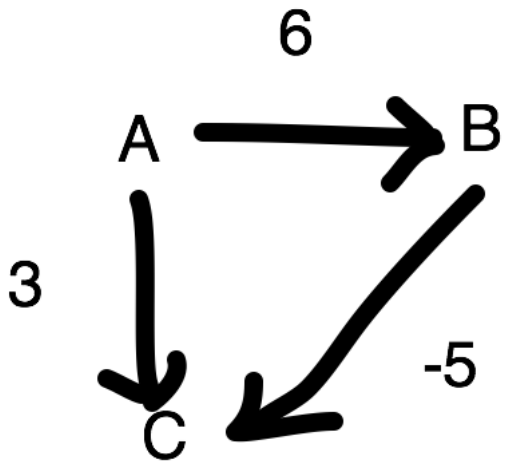


A B C D E F G H
A–C B D E F G H
A–C D–G B E F H
A–C E–D–G B F H
A–C E–D–G F–H B
A–C- D – G F–H B
|
E

A – C – F – H B
|
D – G
|
E

A – C – F – H
|
D – G
|
B – E

Problem 2:



From A to B, the Dijkstra will choose (A, B) directly, set (A, B) = 6. From A to C, the Dijkstra base on greedy algorithm it will choose A to C directly, A -> C. which set (A, C) =3. But it is not the shortest path from A to C. The shortest path from A to C is A->B->C, the total length is 1.

Problem 3:

For example, we could use two color to assigned the vertices.

1. Assigned the green to the start vertices. Add to the set V1.
2. Assigned the red to its neighbor vertices. Add to the set V2.

3. Assigned the green to neighbor’s neighbor vertices. Add to the set V1.
4. Keep the loop, until all the vertices is assigned the color.
5. If in the V1 there are both have green and red color or in the V2 there are both have green and red color, this graph will not be bipartite graph. Or much more easily to see, if the neighbor’s color is same as the color for current vertices, this graph will also not be bipartite graph.