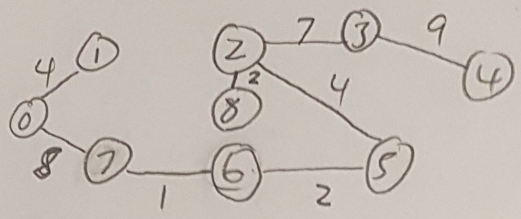
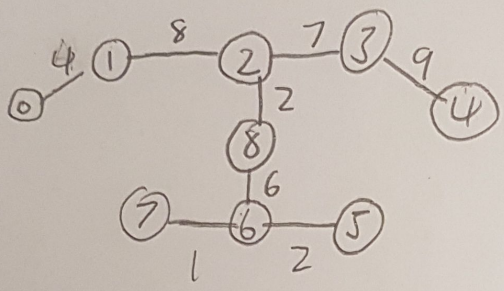
1a. Kruskal



order
7-6
6-5
8-2
0-1
2-5
2-3
0-7
3-4

1b. Prim



order
8-2
8-6
6-7
2-3
3-4
2-1
1-0

2a. It detects cycles by finding the edge with the least possible weight that connects any two tree. It is computed sometimes to detect existence of cycles inside the computed MST and rooting them out. It selects one unknown vertex and uses BFS to explore every vertex reachable to the first vertex. If it is a cycle it will conclude and do it again

2b. First know the edges are n-1 then pick a vertex, goes through the option of the vertex and then some to see it will connect back. If not then pick the connecting vertex and move forward. It also has to detect if each node is reachable from every node. Thats why it has to divide the graph.

3. Prim's fail because it assumes each node is reachable from ever node which may not be true for directed graphs
Krusical fail because it wont be able to detect cycles and also it needs to satisfy MST, The Union-Find method.

4. One has 1 base the other has 2 bases. One has an apex the other one doesn't.

9. Dijkstra is a greedy algorithm so it will only hold if weights are positive.

2. Topological ordering is only possible if the graph has no directed cycles unless it is DAG since it has no directed cycles.

8. This is because Djikstra's Algorithm are mainly used on directed graphs which generally doesn't need to detect cycles.