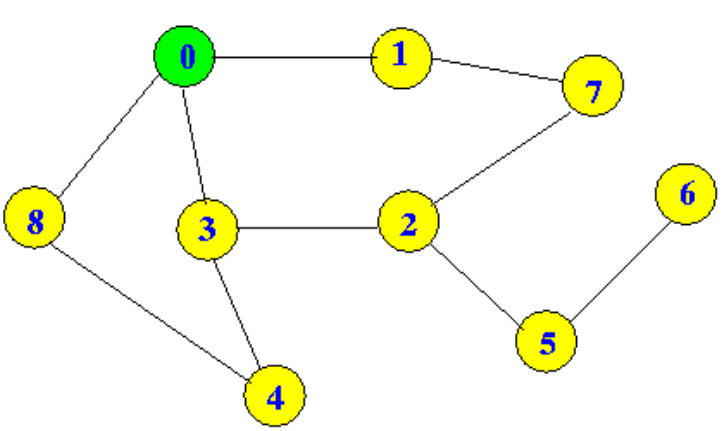
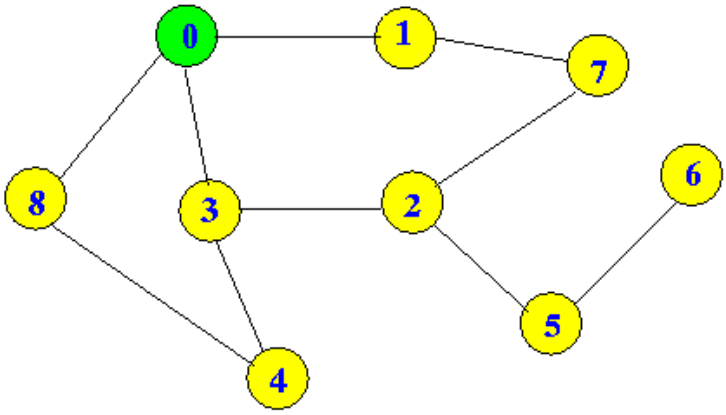
BFS AND DFS

We start from vertex 0.





BFS

DFS

Queue:

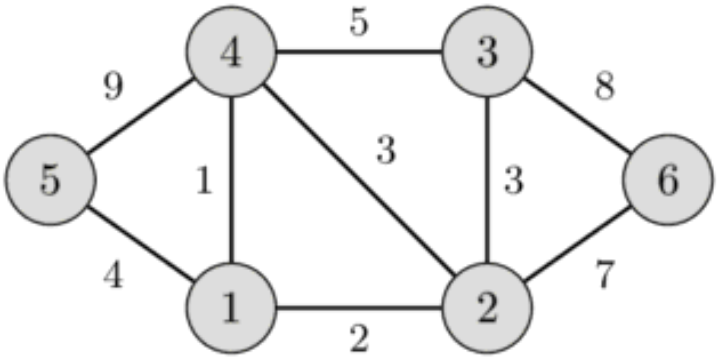
1) 1-3-8
2) 3-8
3) 3-8-7
4) 8-7
5) 8-7-2-4
6) 7-2-4
7) 2-4
8) 4
9) 4-5
10) 5
11) 6
12) -

Stack:

1) 8-3-1
2) 3-1
3) 4-3-1
4) 1
5) 2-1
6) 1
7) 7-5-1
8) 5-1
9) 1
10) 6-1
11) 1
12) -

BFS Order: 0-1-3-8-7-2-4-5-6
DFS Order: 0-8-4-3-2-7-5-6-1

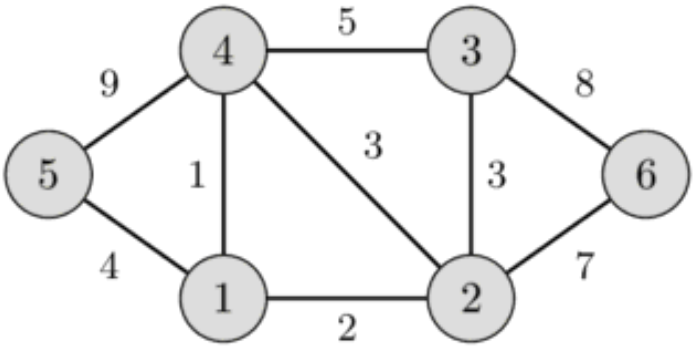
MINIMUM SPANNING TREE



Prim's Algorithm (Start from vertex 6)

Active Set (dest,cost):

- (2,7), (3,8) -> Pick edge (2-6)
- (3,3), (1,2), (4,3) -> Pick edge (1-2)
- (4,1), (5,4), (3,3) -> Pick edge (1-4)
- (5,4), (3,3) -> Pick edge (2-3)
- (5,4) -> Pick edge (1-5)
- n-1=5 edges picked: stop)

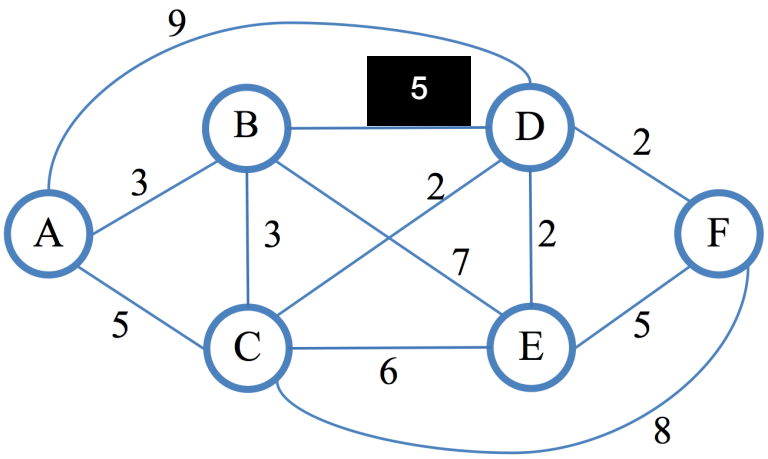


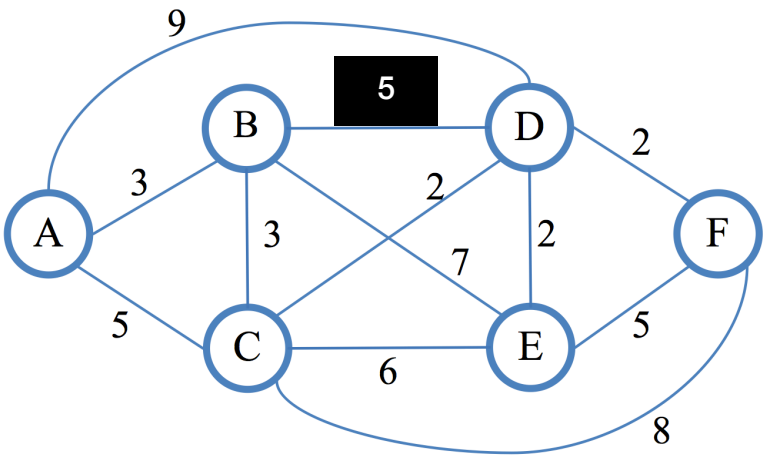
Kruskal's Algorithm

Edges
(1-4): 1
(1-2): 2
(2-4): 3
(2-3): 3
(1-5): 4
(3-4): 5
(2-6): 7
(3-6): 8
(4-5): 9

We pick (1-4), then (1-2). We cannot pick (2-4) because it creates a cycle. We pick (2-3), then (1-5). We cannot pick (3-4) because it creates a cycle. We pick (2-6). We picked 5 edges and we are done.

SHORTEST PATH ALGORITHMS





Dijkstra's Algorithm (break ties alphabetically)

Active Set:

1) (B,3), (C,5), (D,9)-> Pick B
2) (C,5), (D,8), (E,10) -> Pick C
3) (D,7), (E,10), (F,13)-> Pick D
4) (E,9), (F,9) -> Pick E
5) (F,9) -> Pick F
6) -

Shortest paths:
A-B:3, A-C:5,A-D:7,A-E:9,A-F:9

Bellman-Ford's Algorithm

1 hop: A-B:3, A-C:5, A-D:9, A-E:∞, A-F:∞
2 hops: A-B:3, A-C:5, A-D:7, A-E:10, A-F:11
3 hops: A-B:3, A-C:5, A-D:7, A-E:9, A-F:9
4 hops: A-B:3, A-C:5, A-D:7, A-E:9, A-F:9-> no change, algorithm terminates