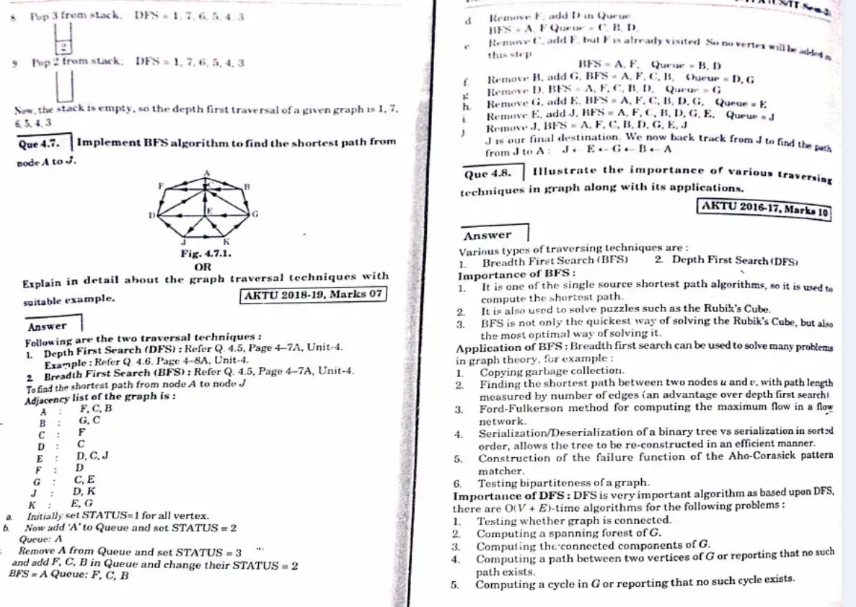
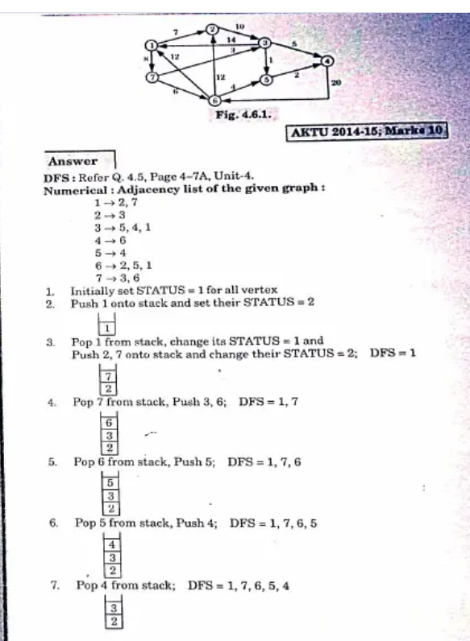
Representation of graph:  
Adjacency list  
adjacency matrix  
incidence matrix[1,-1,0]

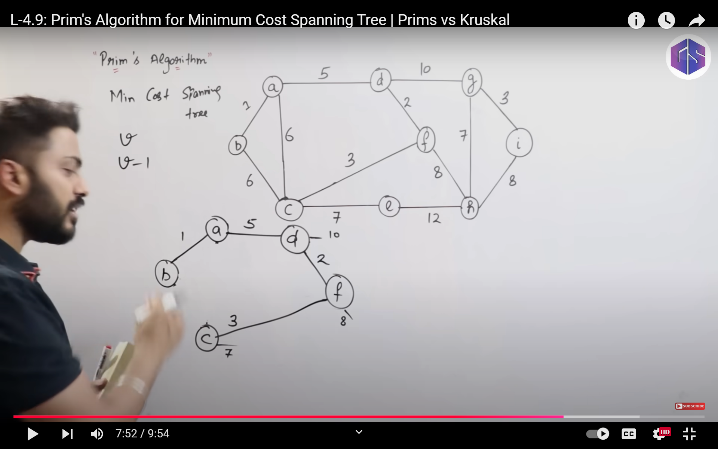
BFS- queue(BQ)(level order traversal) – used in undirected graphs – shortest path  
DFS- stack(DS)(depth wise travel before backtracking travels full path) – used in directed – cycle detection  
  
WORKING:

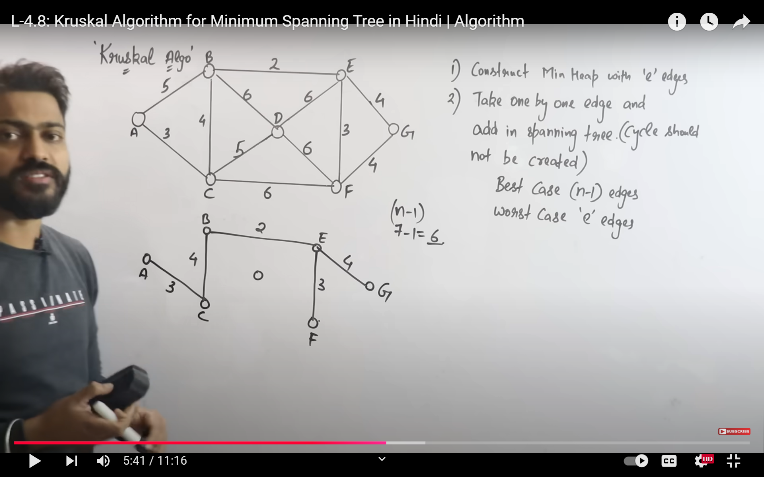


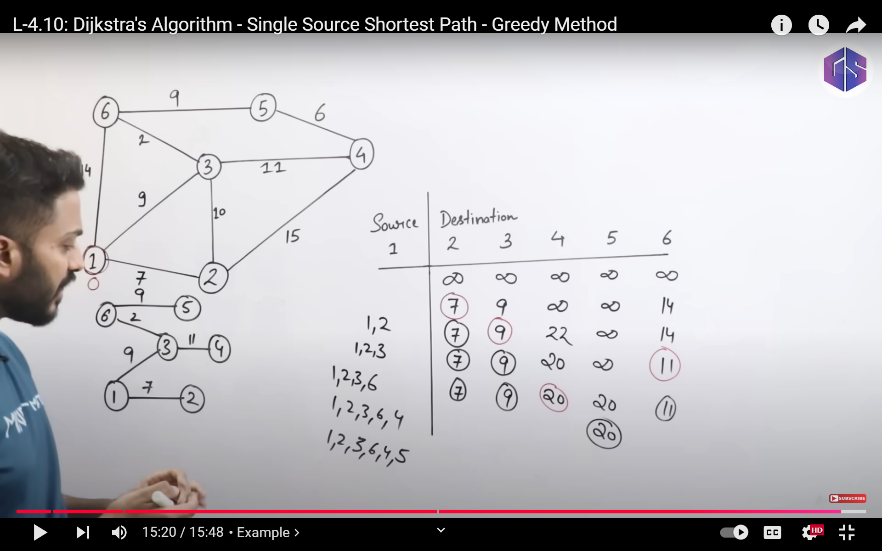
**When to Use BFS vs DFS?**

* Use **BFS** when the shortest path or shallow solutions are needed.
* Use **DFS** when exploring all possible solutions or when memory usage is a concern.

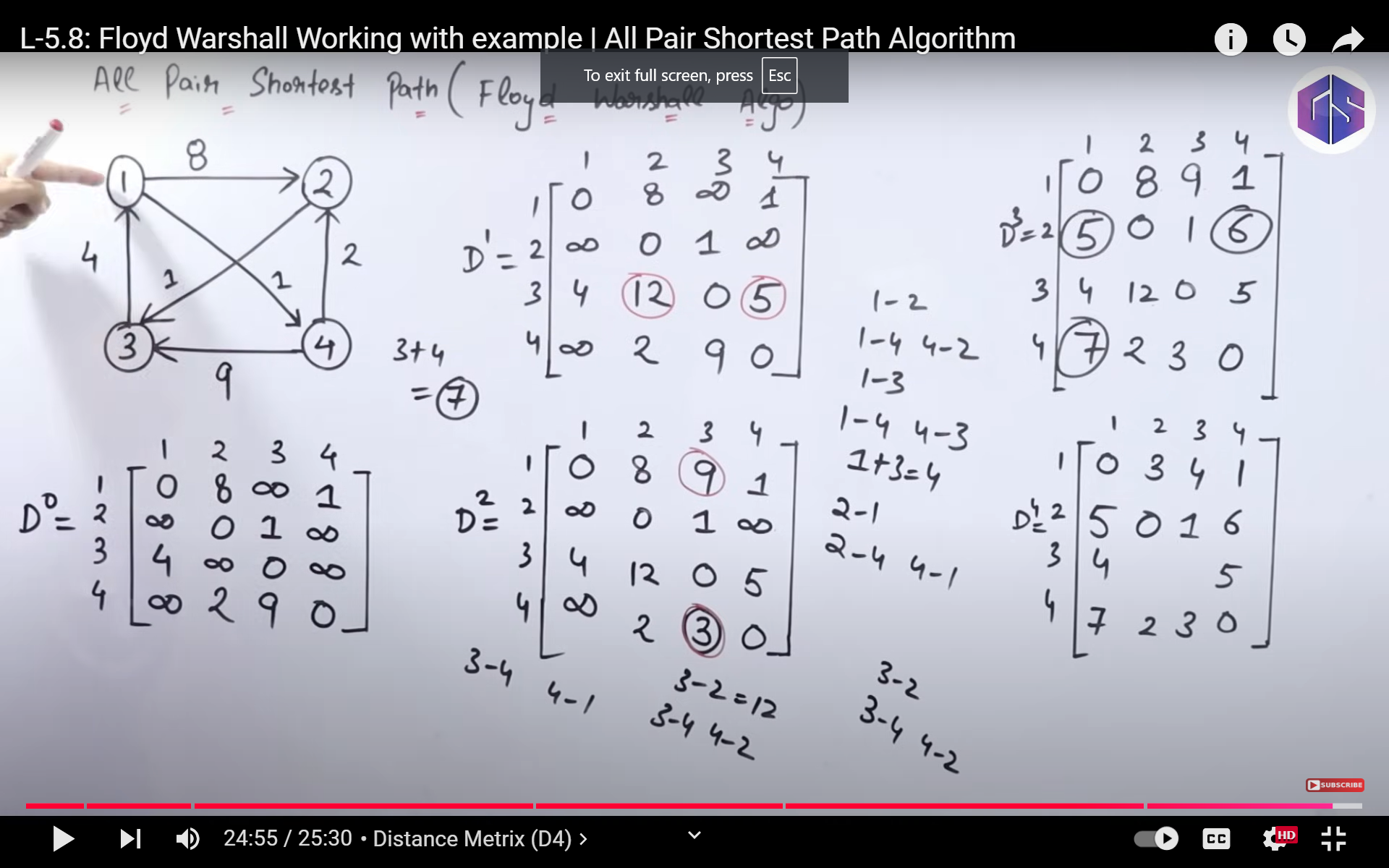
List is always given in adjacenc list type(int n=adj.size();

PRIM’S ALGORITHM:  
to find minimum code spanning tree (ST is if no of vertex is v then no of edges is v-1).  
in prims we can start from any vertex and It should not form any cycle prims connected chlta whereas Kruskal disconnected chlta last mei connected bnta hai  


  
prims mei pichli edges ki length ko bhi mind mei rkhna hai.

DIJKSTRA ALGORITHM:  
single source shortest path: it finds shortest route from a point to all destinations   
example: map, DNA mapping.  
it uses greedy approach.  
**complexity-ElogV**  


How belmon ford works we have to relax every edge v-1 number of times. While Dijkstra fix after one relaxation of a vertex. Dijkstra fails bcz it does not give minimum distance.

FLLOYD WARSHALL ALGORITHM(Directed weighted graph)  


D4 tak ate ate sare nodes ko use kr skte hain.  
**complexity-VElogV**