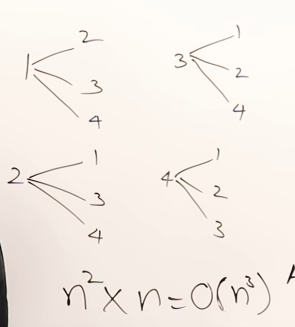
# Floyd Warshall

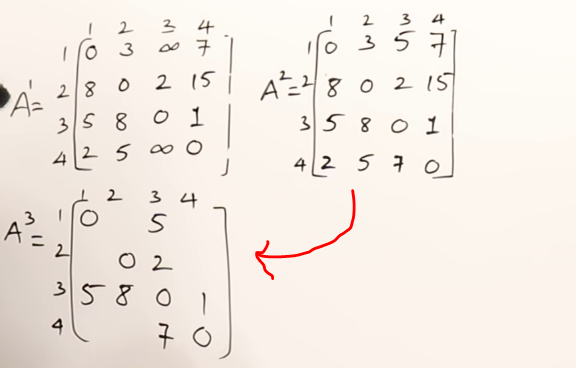
Floyd Warshall 🡪 shortest path from every node to their neighboring nodes.   
Dijkstra’s algorithm 🡪 shortest path from one source to the neighboring destinations.  
If we run Dijkstra’s algorithm on all nodes then I will get the shortest path from every node to their neighboring node.

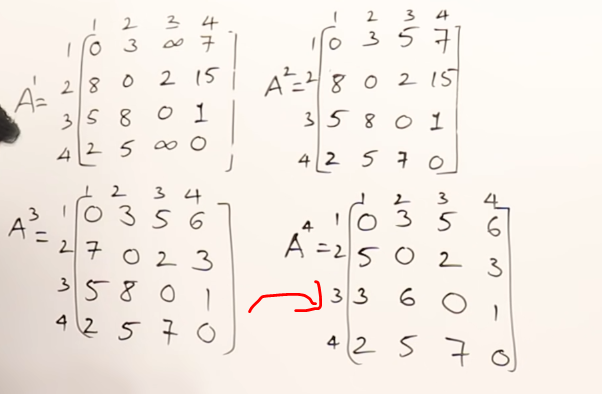


Dijkstra’s algorithm 🡪 O(n^2)   
Nodes 🡪 n  
So the total time complexity 🡪 n^2 \* n 🡪 O(n^3)

Step-1 (via vertex-1) 🡪 Take intermediate vertex-1

Step-2 (via vertex-2) 🡪 Take intermediate vertex-2

Step-3 (via vertex-3) 🡪 Take intermediate vertex-3  


Step-4 (via vertex-4) 🡪 Take intermediate vertex-4

Terminologies

1) **Residual graph** : It is a graph which in-takes additional possible flow. If there is such path from source to sink, then there is a possibility to add flow.

2) **Residual capacity** : (Edge-flow) 🡪 Original capacity

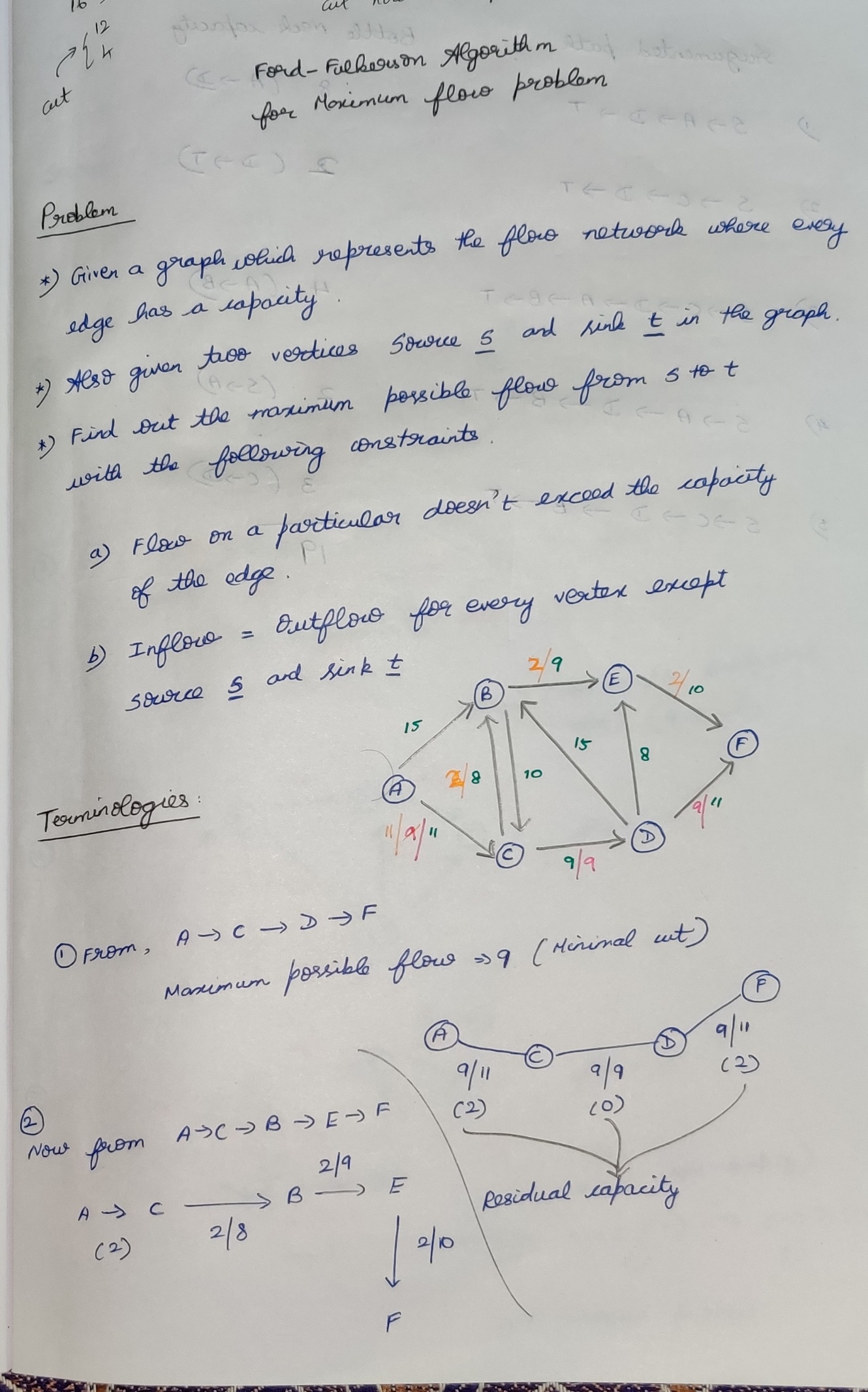
3) **Minimal cut** : Known as **(bottle-neck capacity),** which decides the maximum possible flow from source to sink through an augmented path.

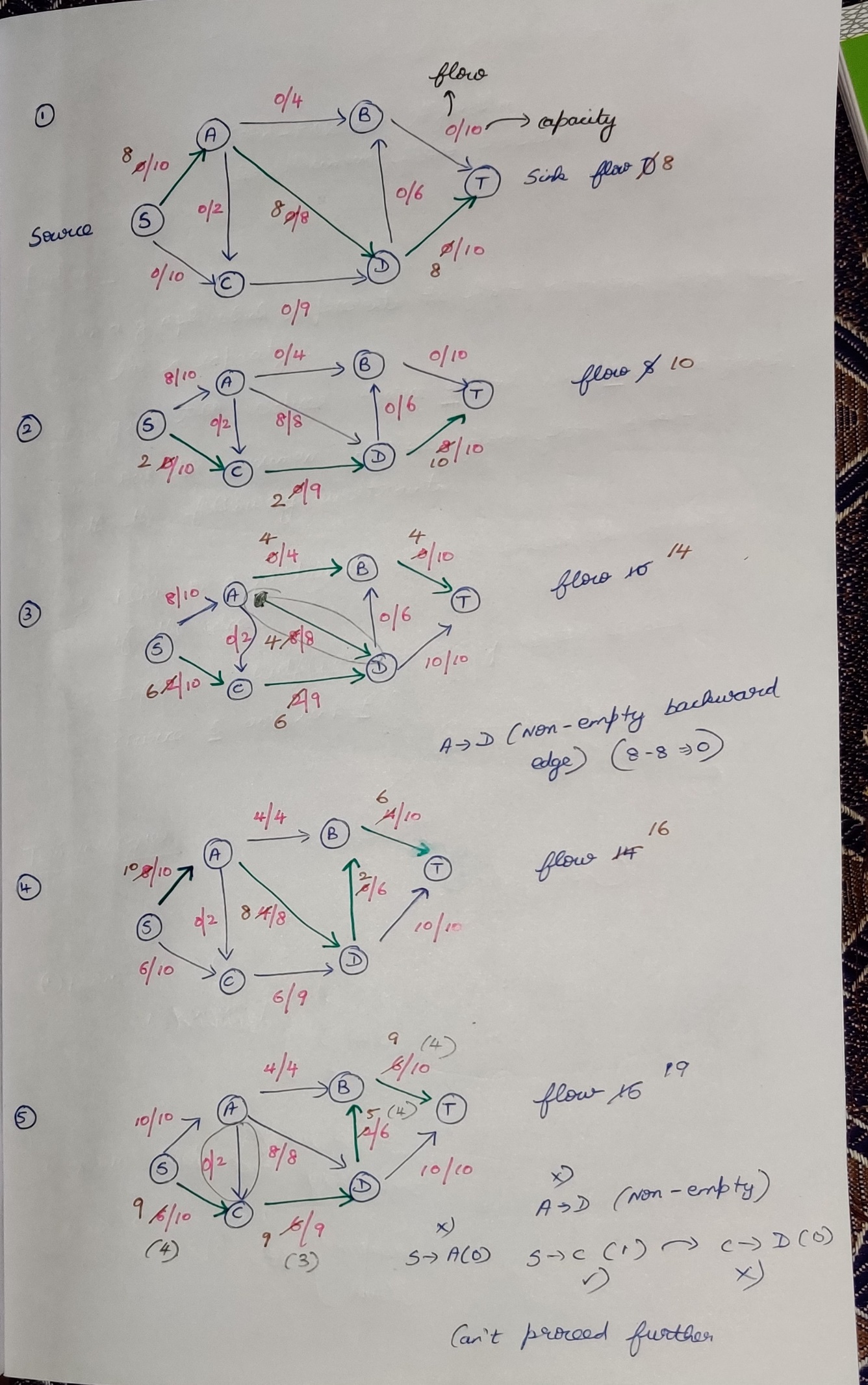
4) **Augmented path:** Can be done in two-ways

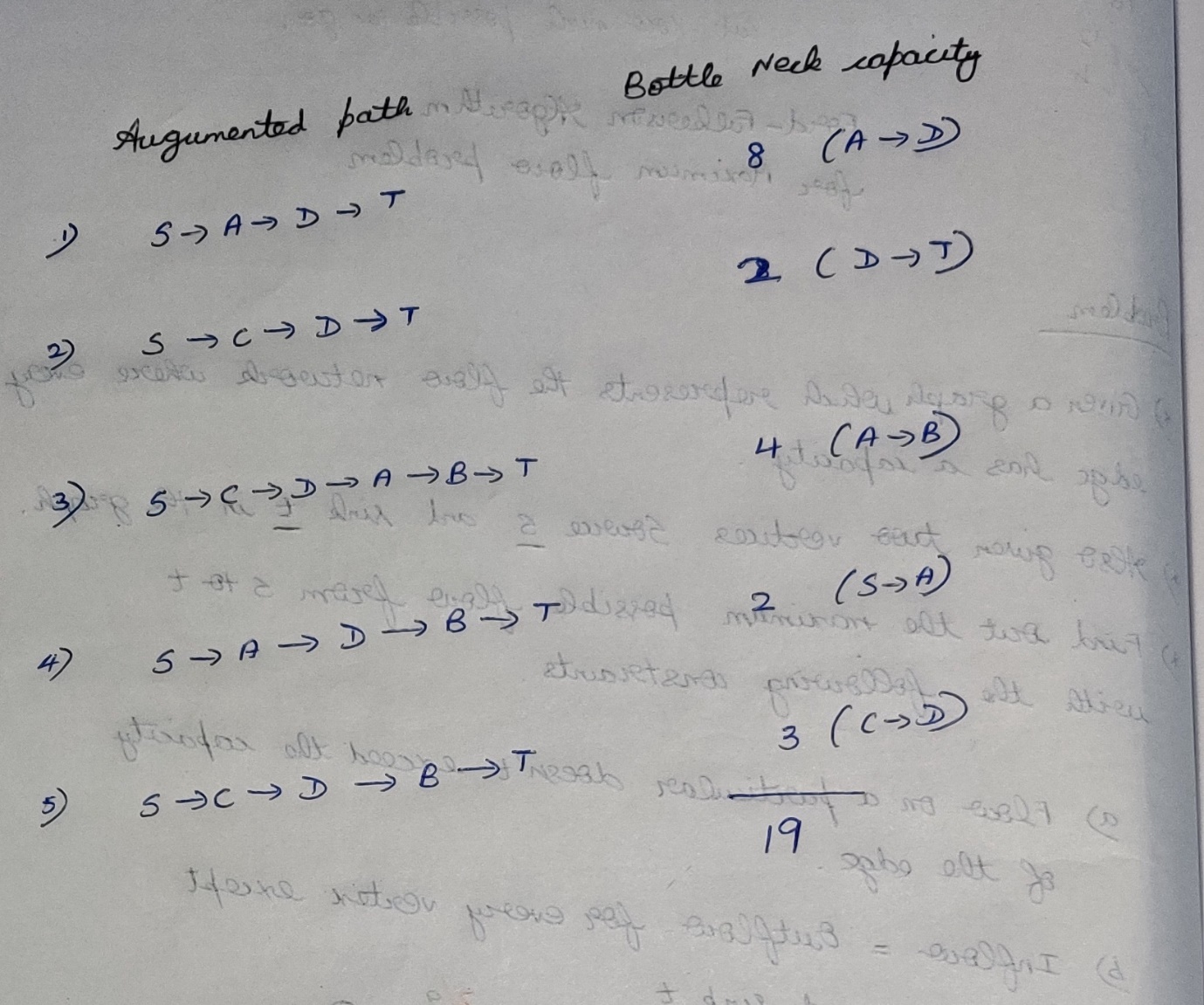
\*) Non-full forward edges  
\*) Non-empty backward edges

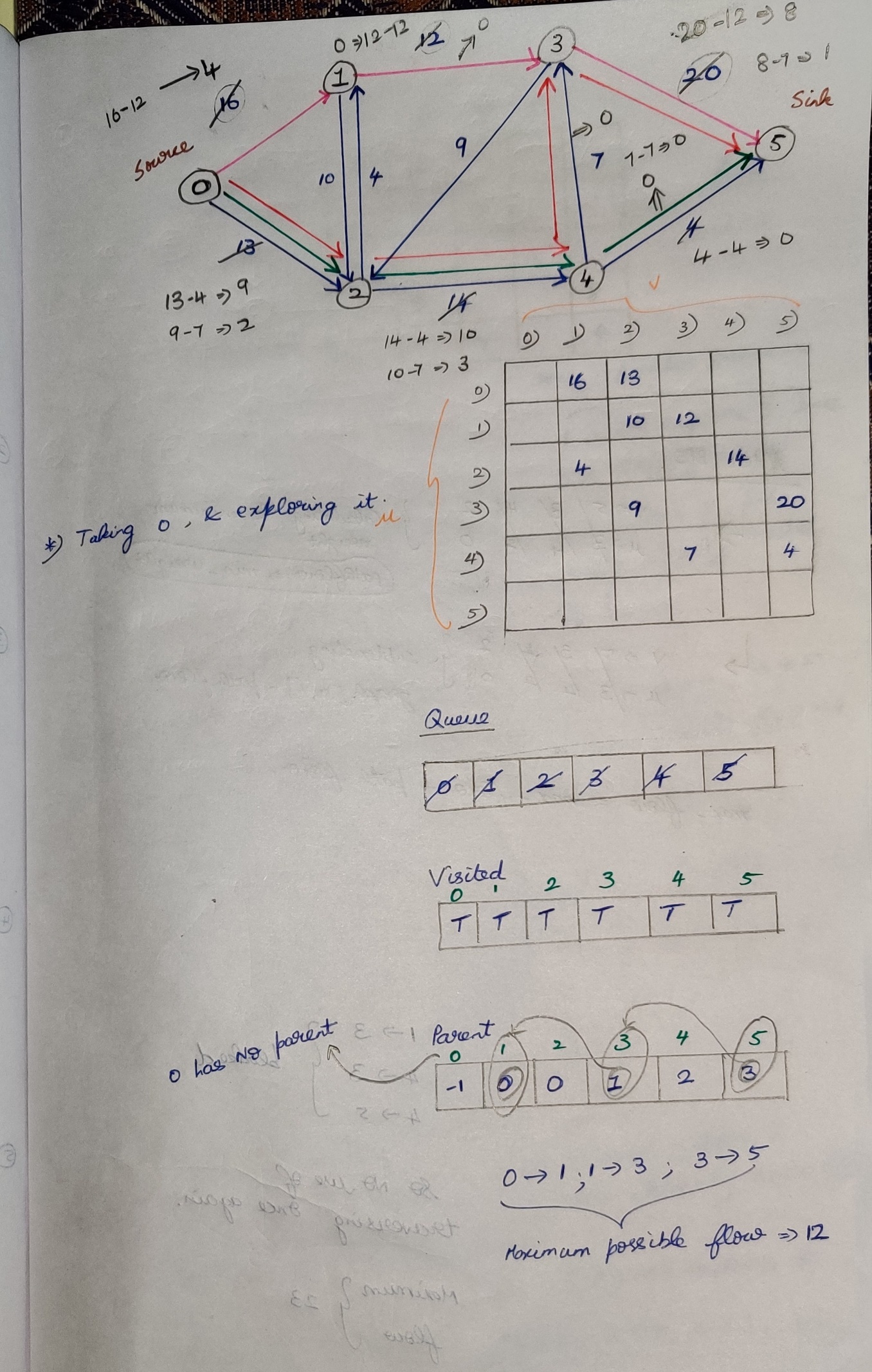
**Algorithm**1) Start with the initial flow as 0.  
2) While there is an augmented path from source to sink, then add this path flow to flow.  
3) Return flow.

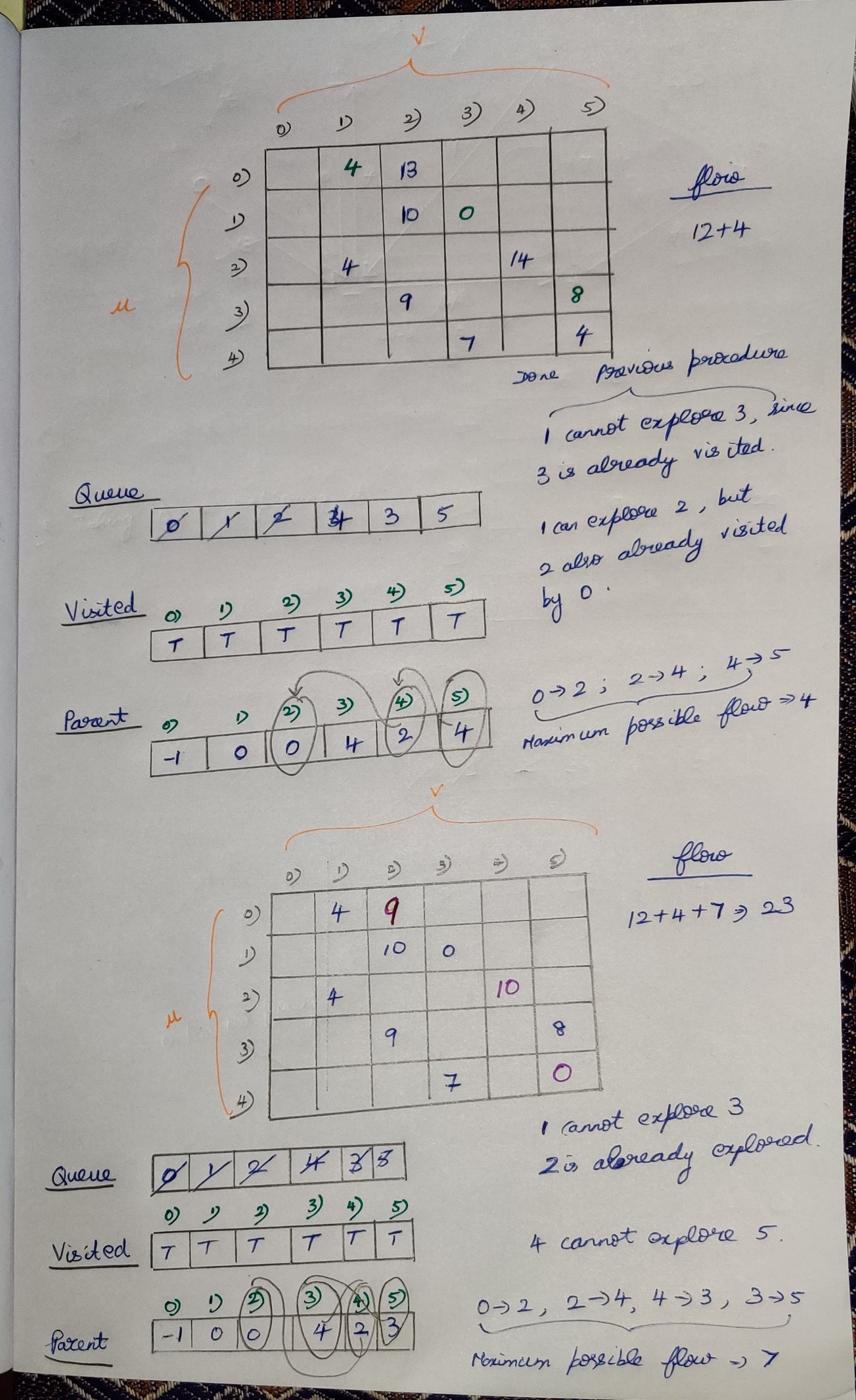
# Ford-Fulkerson Algorithm for maximum flow

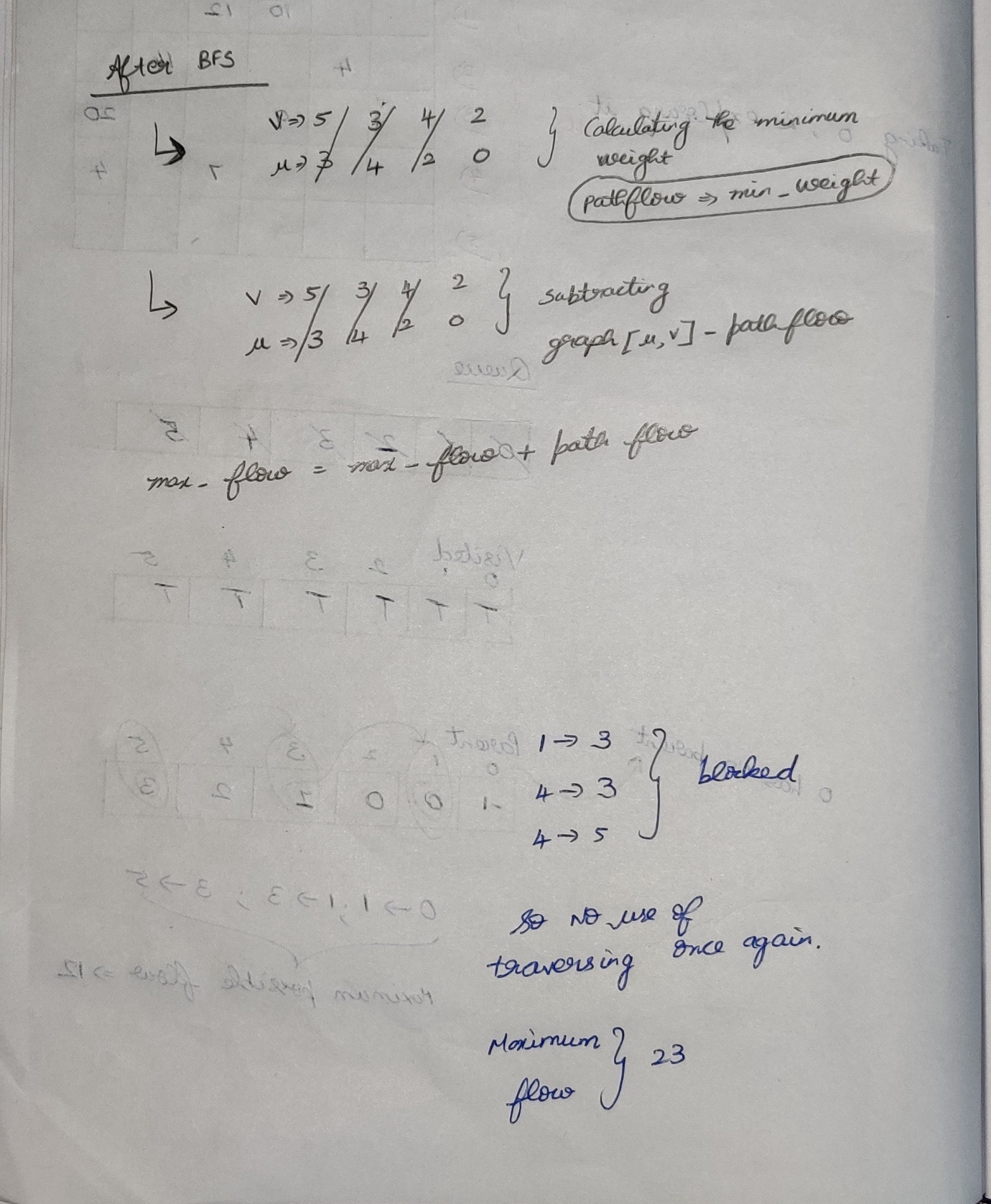


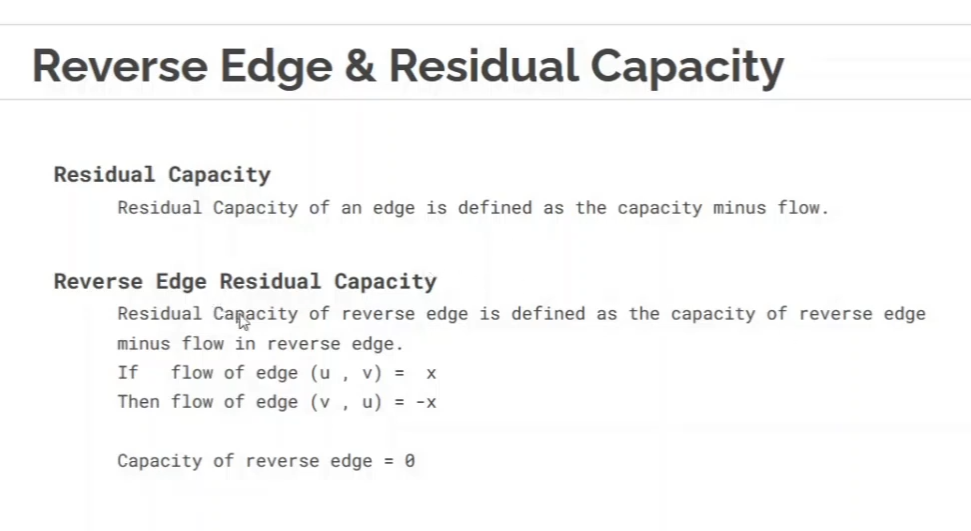


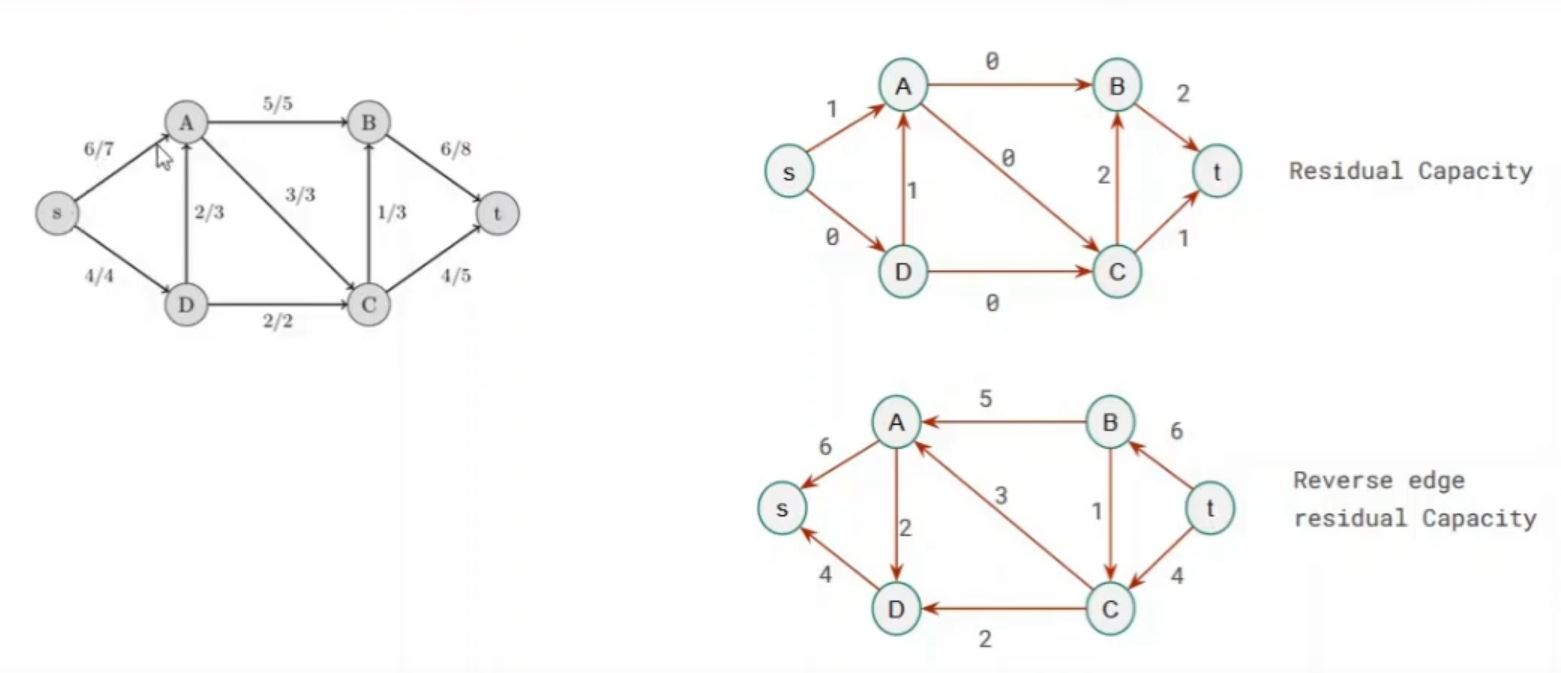




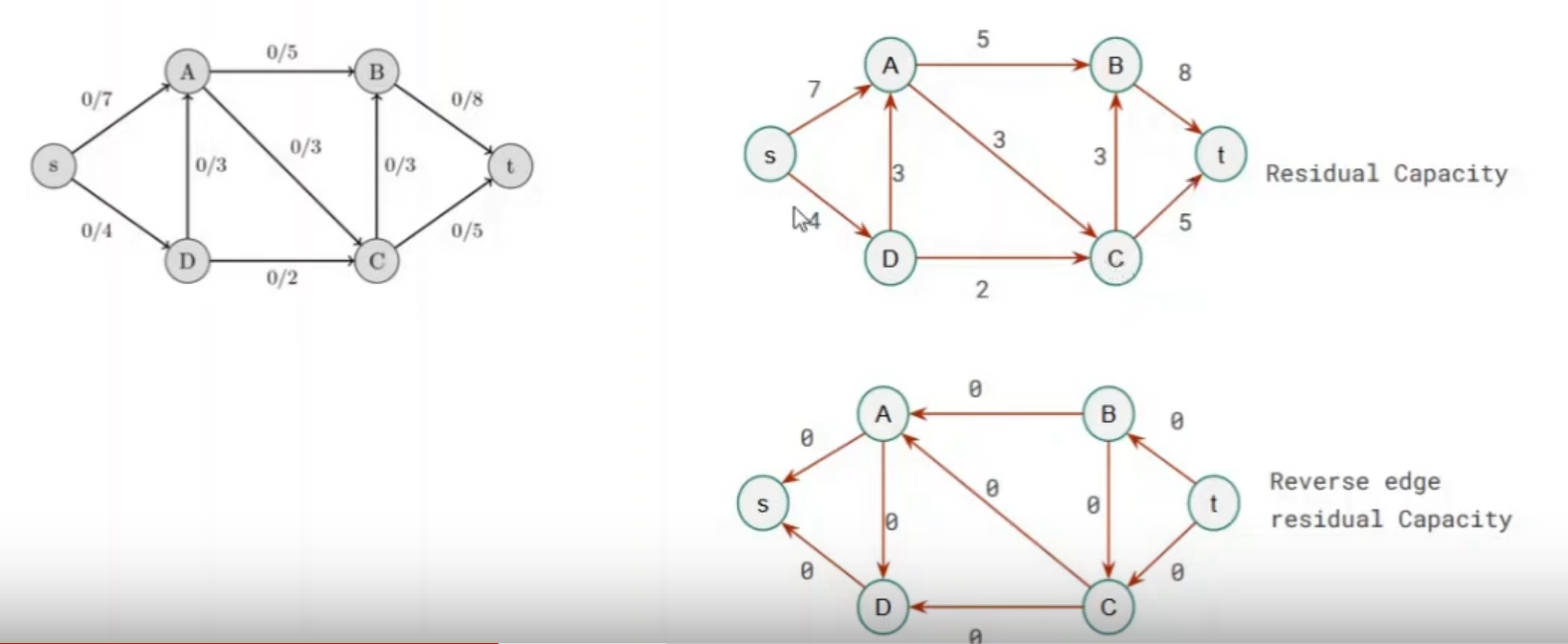






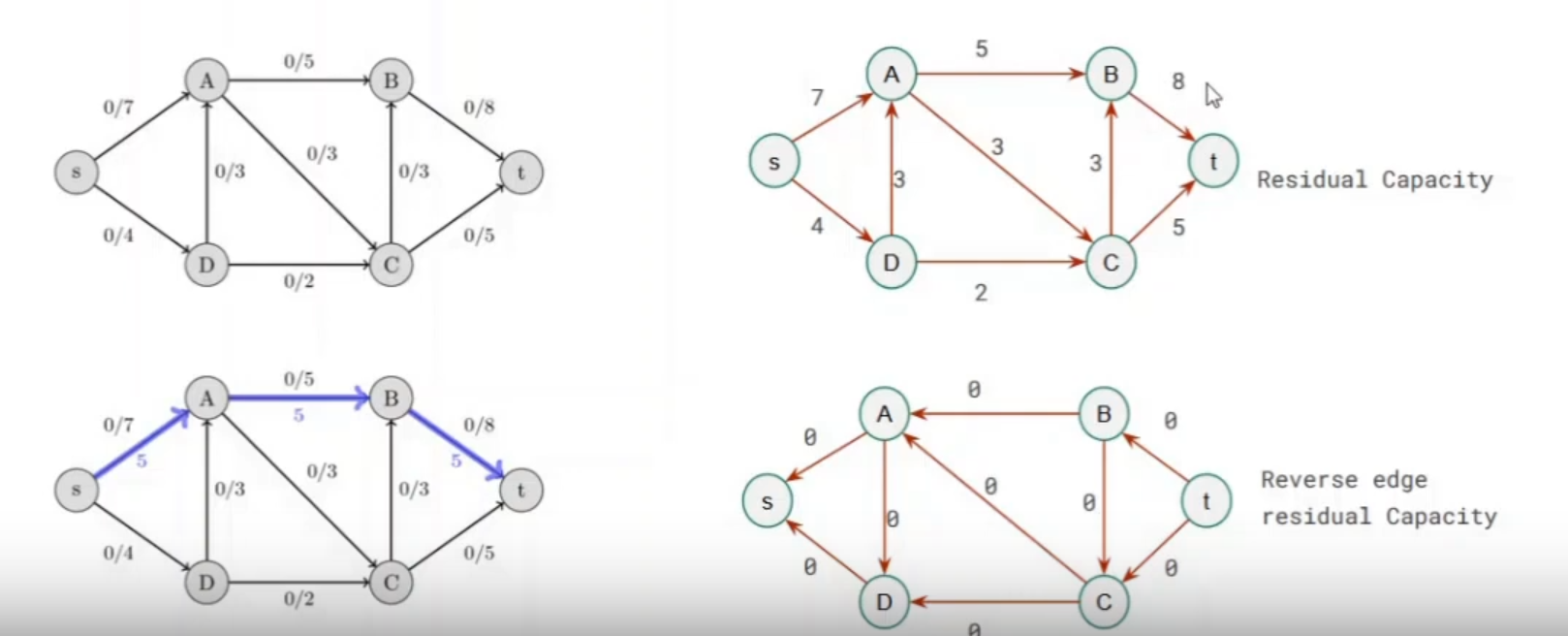


Flow in the normal graph = Capacity in the reverse edge.

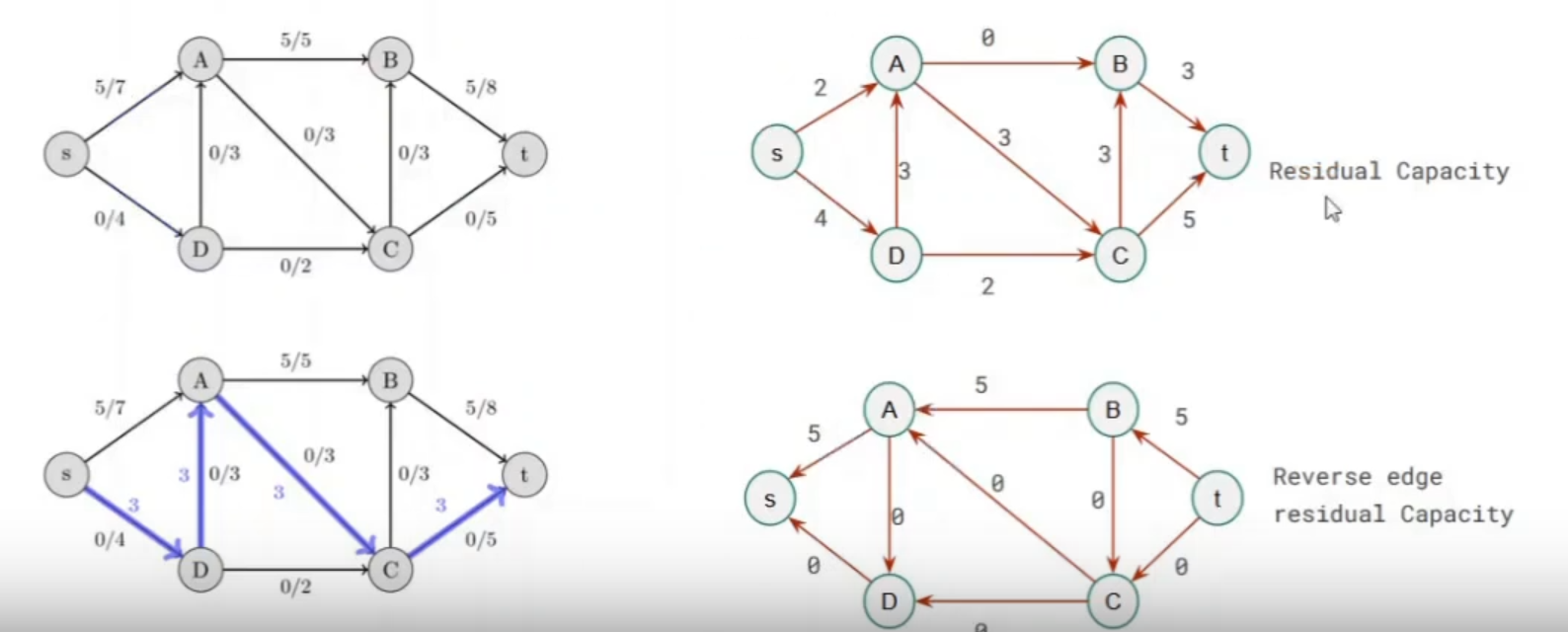


Initially in the reverse edge residual graph, all the flow are 0. So there is no possibility of going in the reverse edge residual graph.

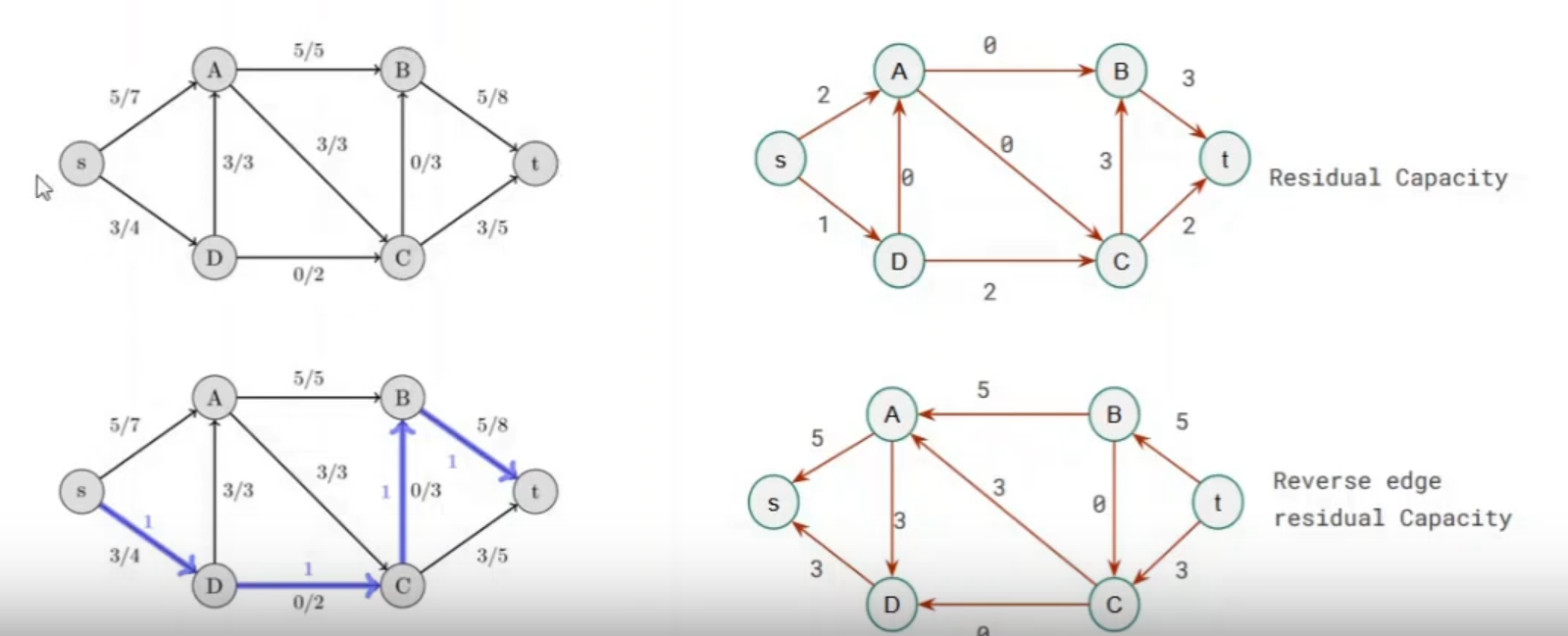
Path-1 🡪 s to A to B to t ( flow = 5)

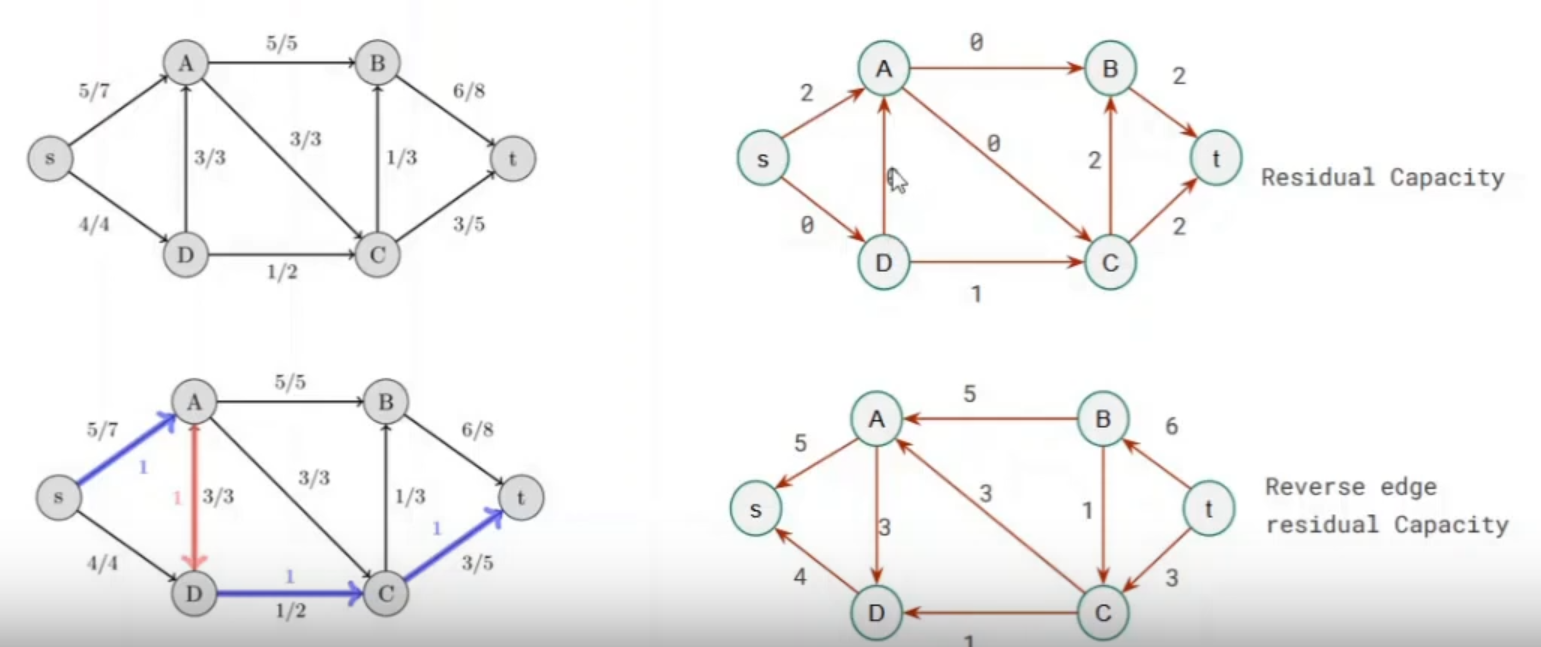


Path-2 🡪 s to D to A to C to t (flow = 5 + 3 = 8)



Path-3 🡪 s to D to C to B to t (flow = 5 + 3 + 2 = 10)



D to C is minimum. So the path is 1.  


Decrease by 1 🡪 A to D