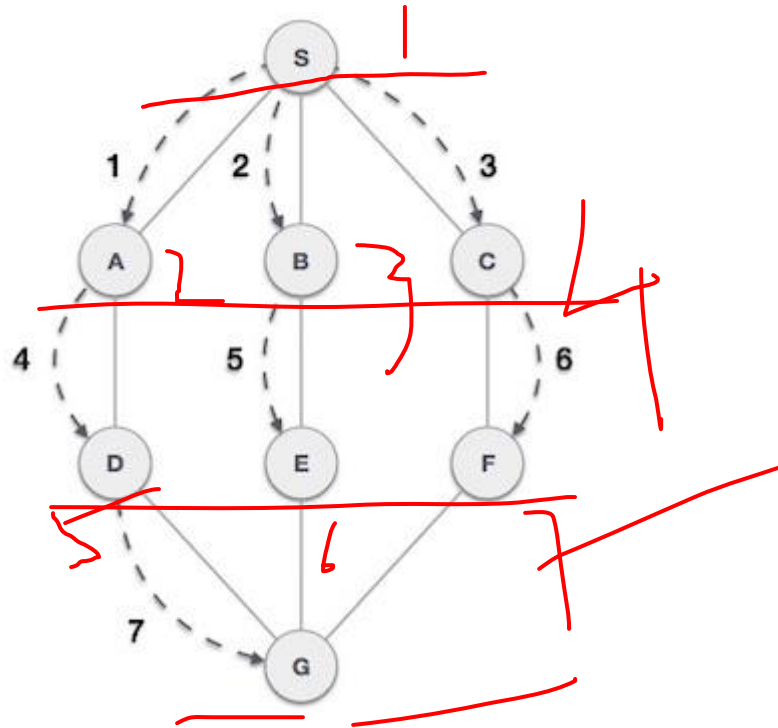


# Graph Traversals - BFS & DFS - Breadth First Search and Depth First Search

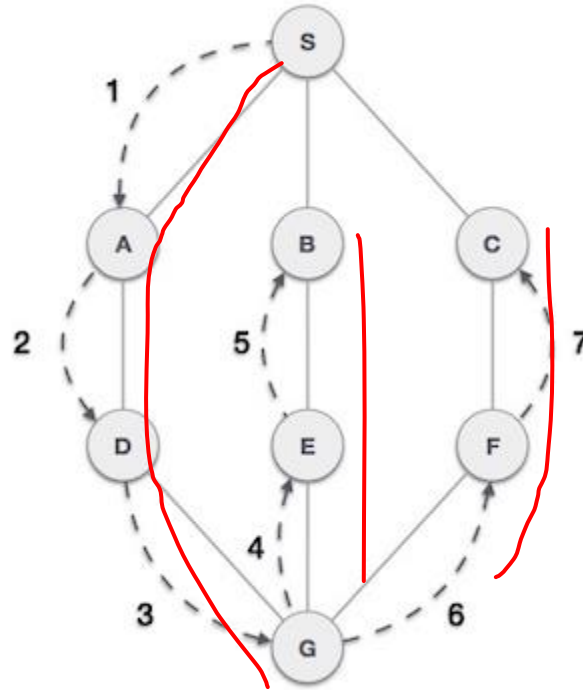
## BFS

Breadth First Search (BFS) algorithm traverses a graph in a breadthward motion and uses a queue to remember to get the next vertex to start a search when a dead end occurs in any iteration.



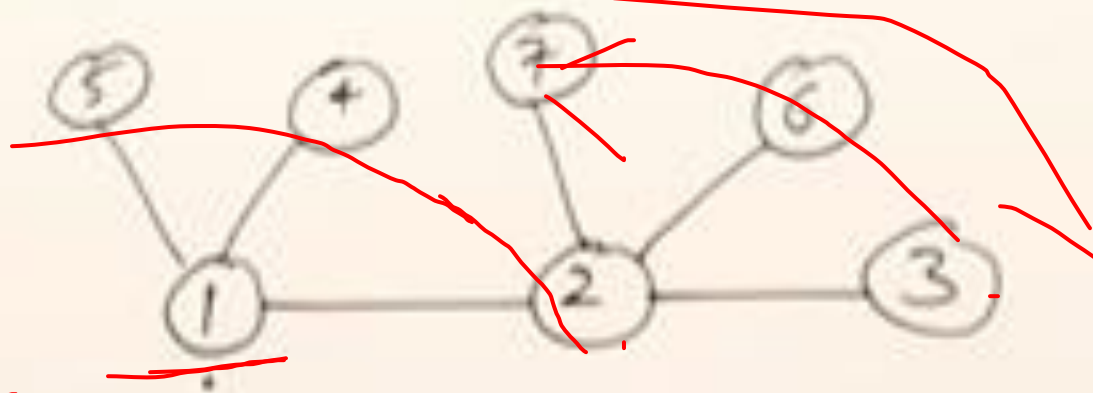
## DFS

Depth First Search (DFS) algorithm traverses a graph in a depthward motion and uses a stack to remember to get the next vertex to start a search when a dead end occurs in any iteration.



# BFS & DFS

- / Visiting a Vertex
- / Exploration of Vertex

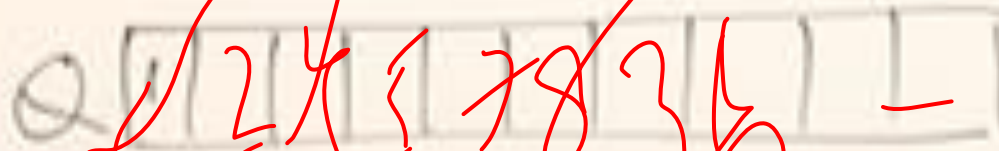


BFS: 1, 2, 4, 5, 7, 3, 6

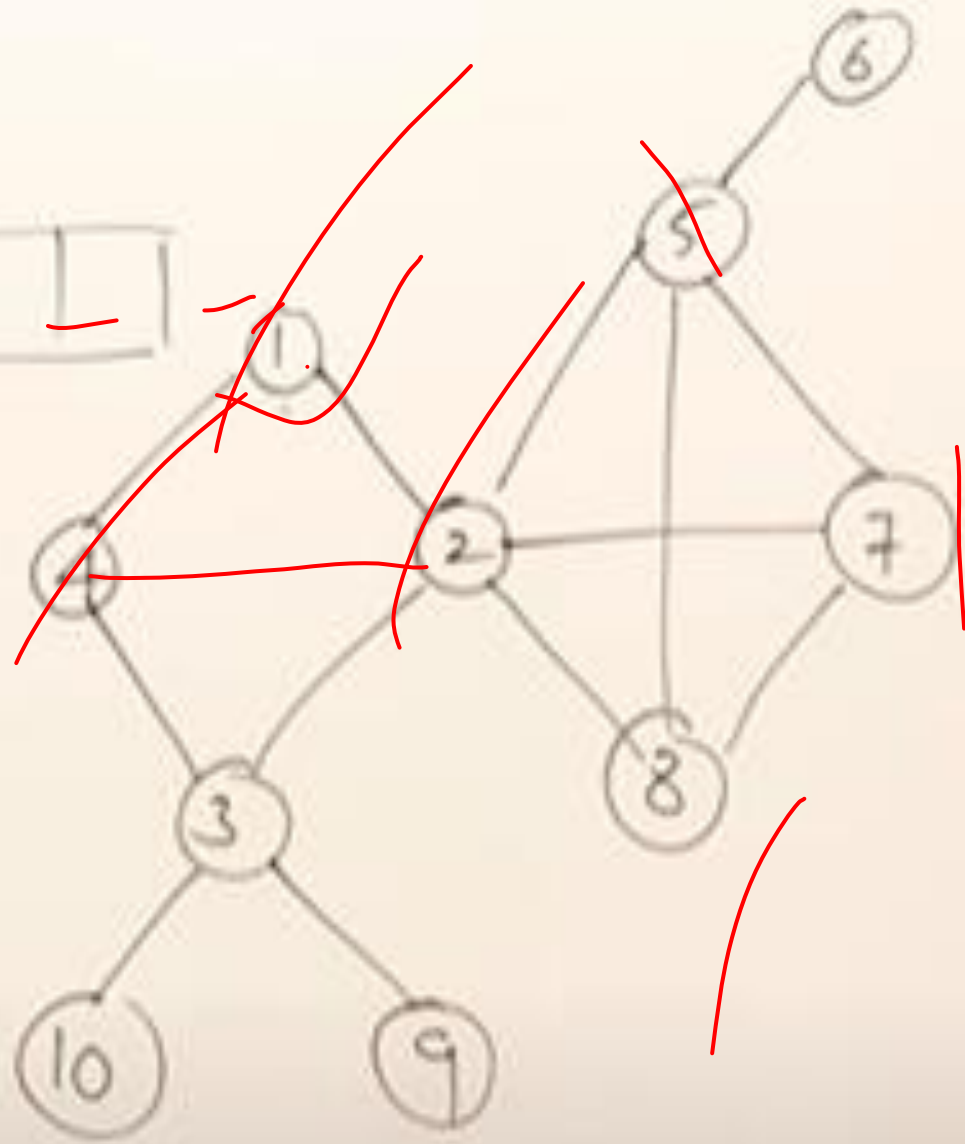
DFS: 1, 2, 3, 6, 7, 4, 5

# BFS & DFS

BFS : 1

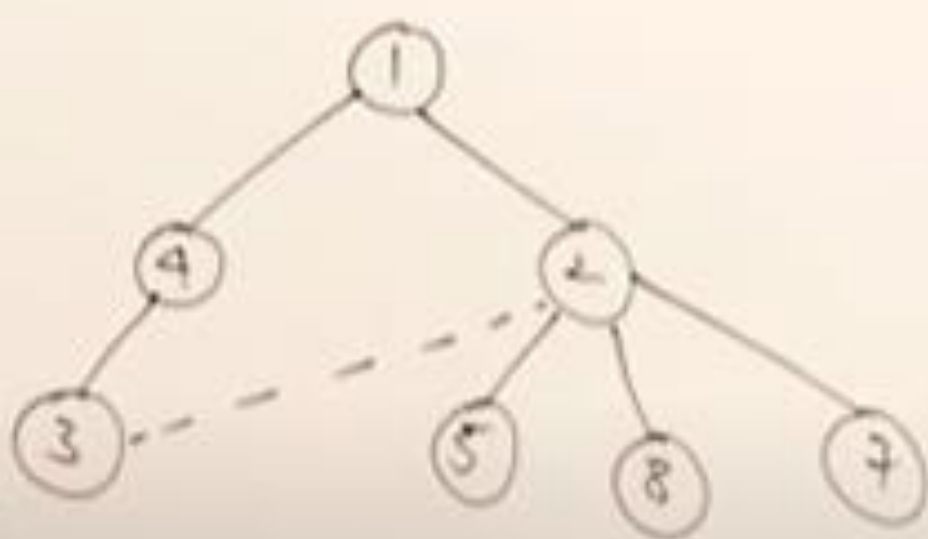
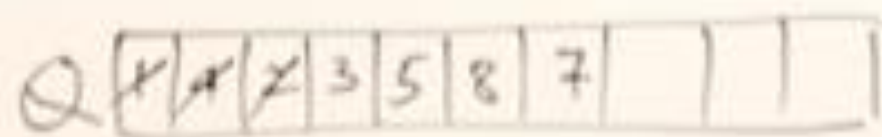


①



# BFS & DFS

BFS: 1, 4, 2, 3, 5, 8, 7

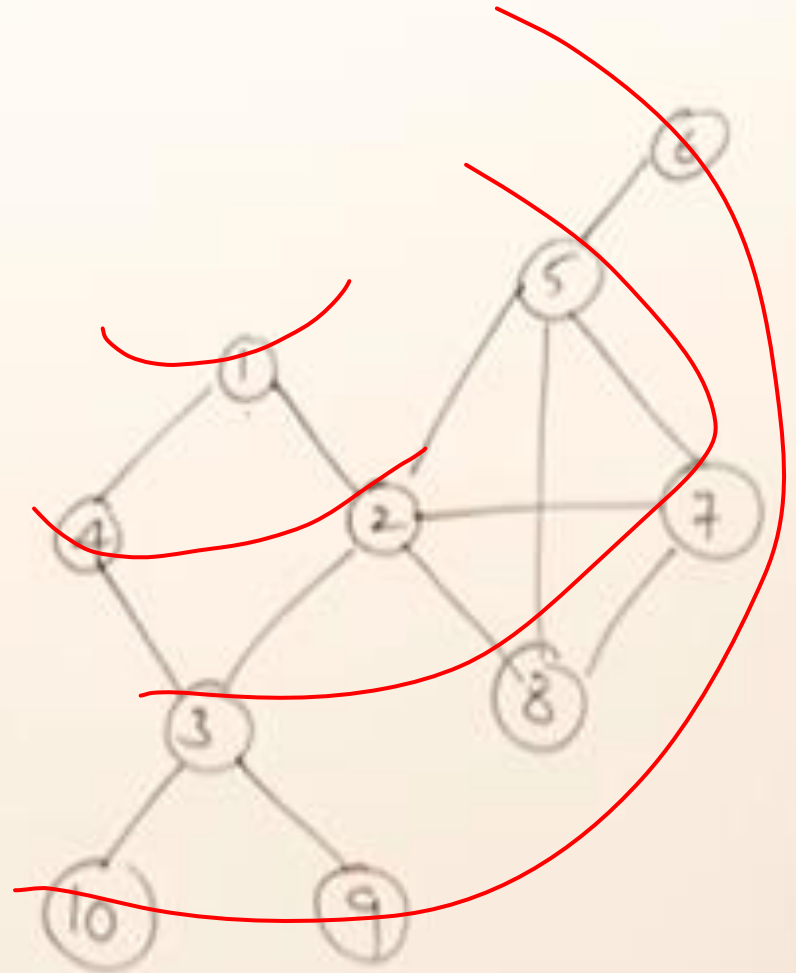
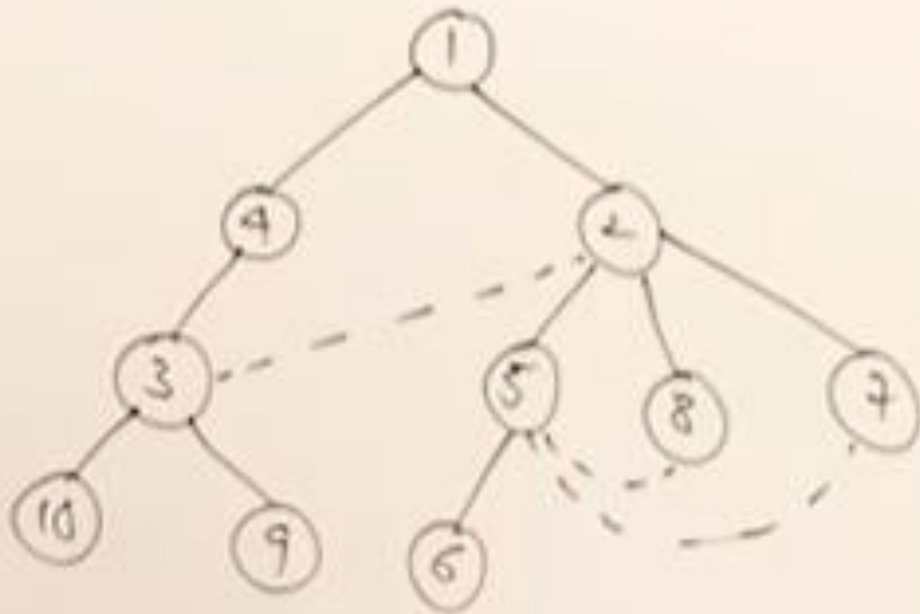


# BFS & DFS

BFS : 1, 4, 2, 3, 5, 8, 7, 10, 9, 6

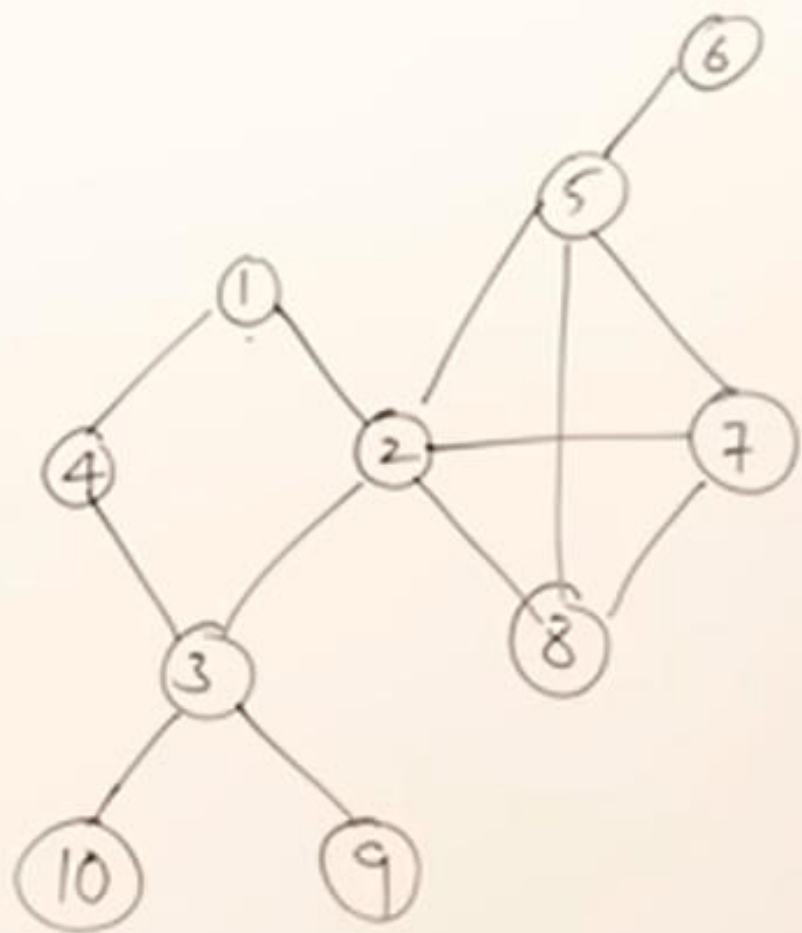
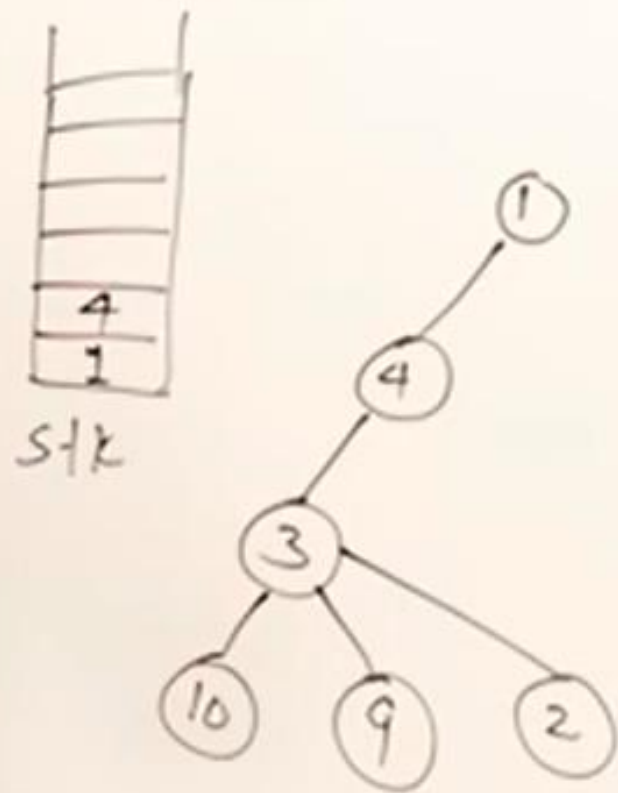
Q 

|              |              |              |              |              |   |   |    |   |   |
|--------------|--------------|--------------|--------------|--------------|---|---|----|---|---|
| <del>1</del> | <del>4</del> | <del>2</del> | <del>3</del> | <del>5</del> | 8 | 7 | 10 | 9 | 6 |
|--------------|--------------|--------------|--------------|--------------|---|---|----|---|---|



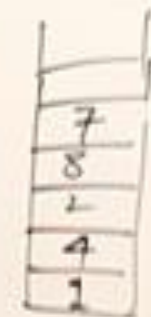
## BFS & DFS

DFS: 1, 4, 3, 10, 9, 2

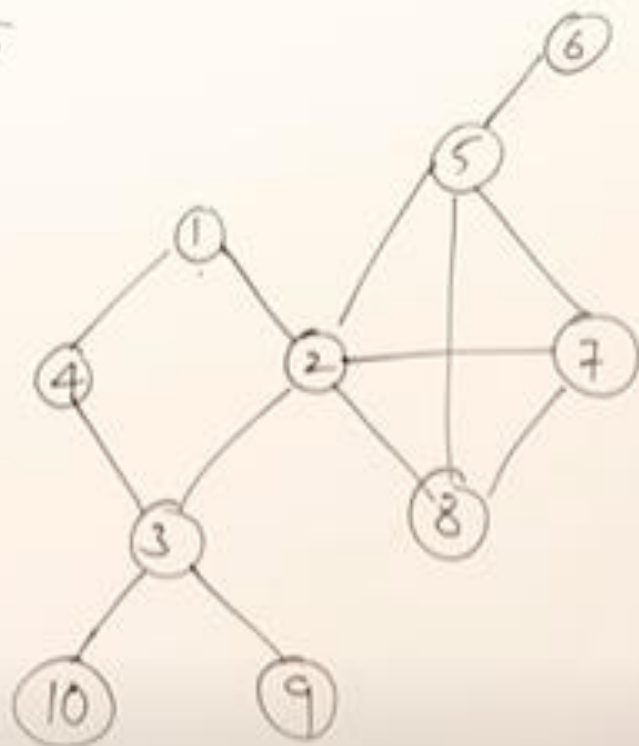
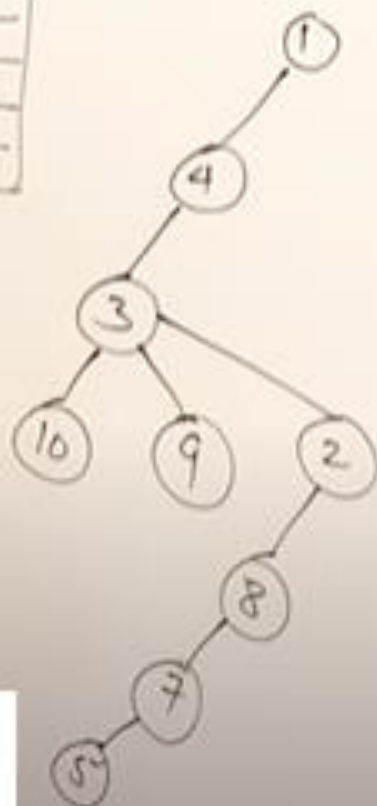


## BFS & DFS

DFS: 1, 4, 3, 10, 9, 2, 8, 7, 5



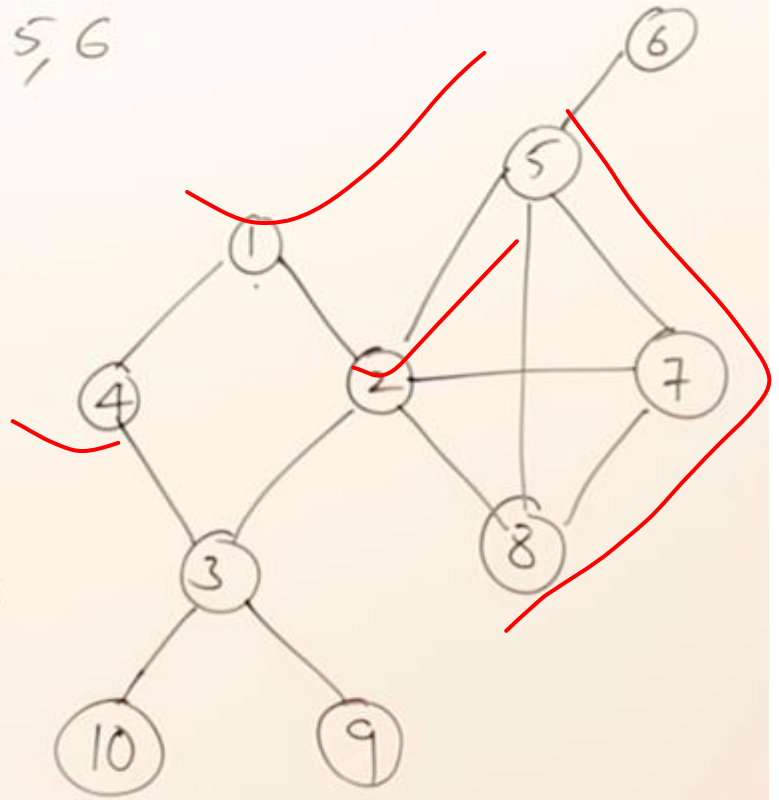
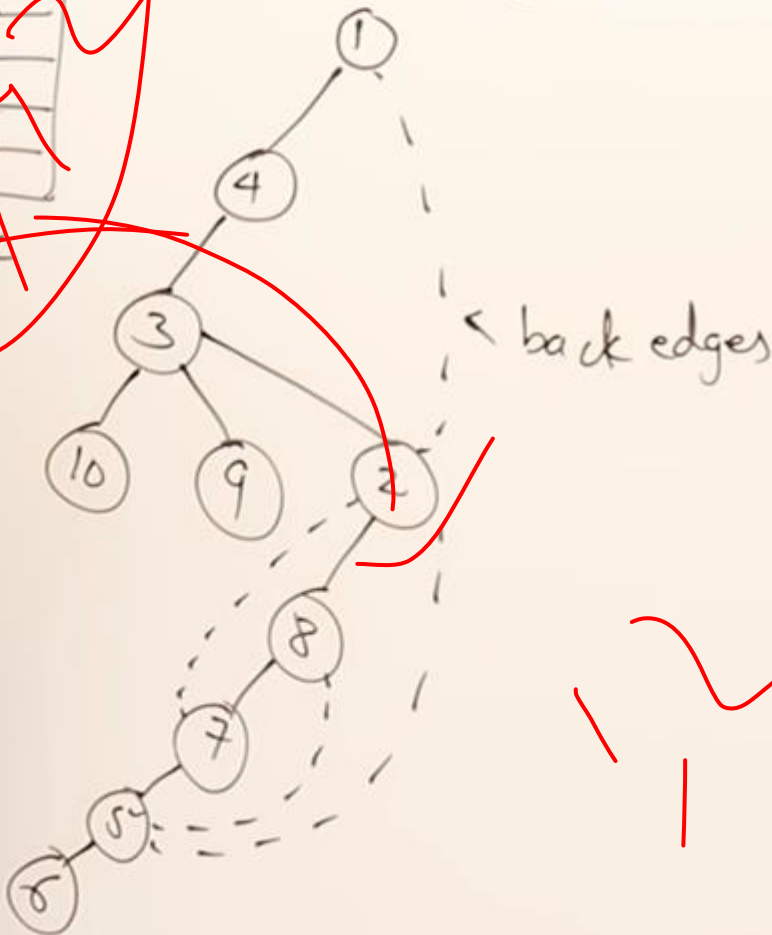
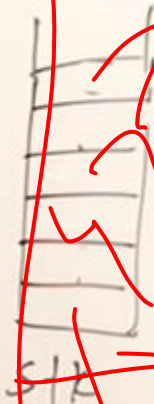
Stack

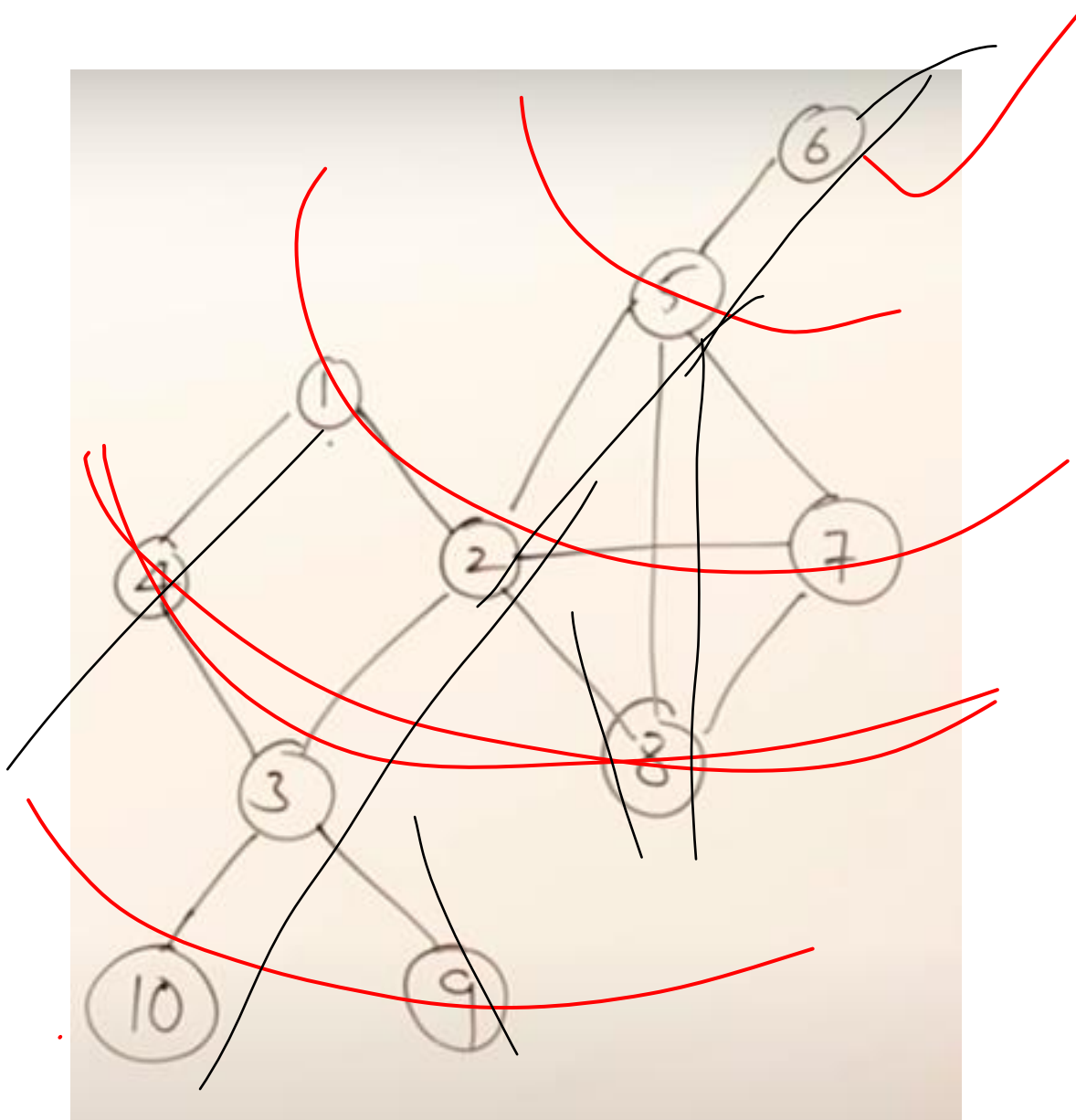


# DFS & DFS

DFS: 1, 4, 3, 10, 9, 2, 8, 7, 5, 6

DFS spanning Tree





## BFS & DFS

1) 1, 2, 8, 7, 5, 6, 3, 9, 10, 4,

2) 3, 4, 1, 2, 5, 6, 7, 8, 10, 9



# Important differences between BFS and DFS

| Sr. No. | Key                           | BFS                                                                                          | DFS                                                                                                                                                    |
|---------|-------------------------------|----------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1       | Definition                    | BFS, stands for Breadth First Search.                                                        | DFS, stands for Depth First Search.                                                                                                                    |
| 2       | Data structure                | BFS uses Queue to find the shortest path.                                                    | DFS uses Stack to find the shortest path.                                                                                                              |
| 3       | Source                        | BFS is better when target is closer to Source.                                               | DFS is better when target is far from source.                                                                                                          |
| 4       | Suitability for decision tree | As BFS considers all neighbour so it is not suitable for decision tree used in puzzle games. | DFS is more suitable for decision tree. As with one decision, we need to traverse further to augment the decision. If we reach the conclusion, we won. |
| 5       | Time Complexity               | Time Complexity of BFS = $O(V+E)$ where V is vertices and E is edges.                        | Time Complexity of DFS is also $O(V+E)$ where V is vertices and E is edges.                                                                            |

Thank you