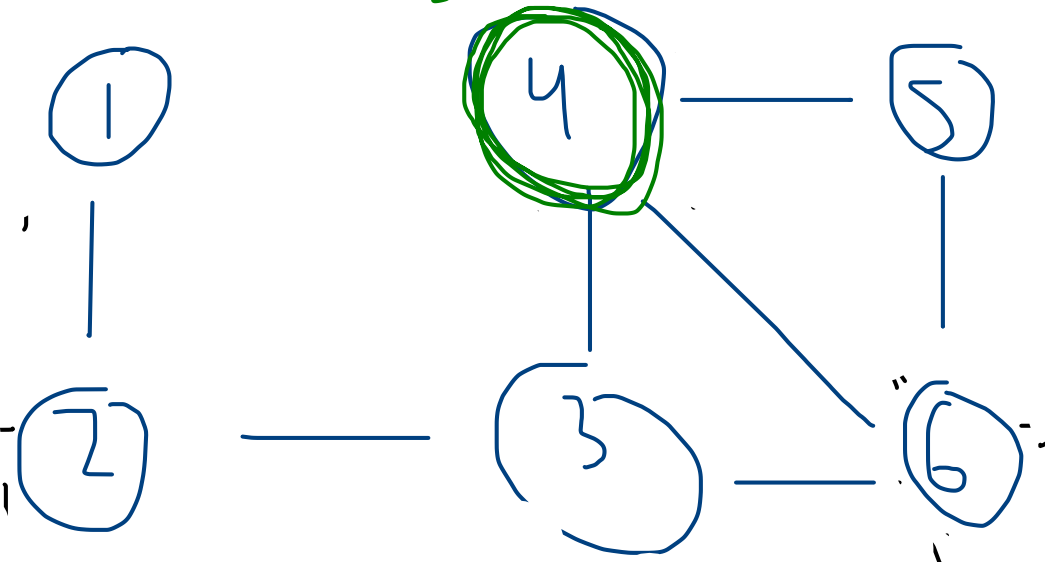


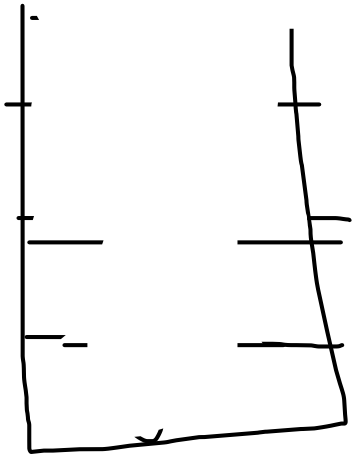
Depth First Search

sol



4 → 5 → 6 → 5

4 → 3 → 2 → 1

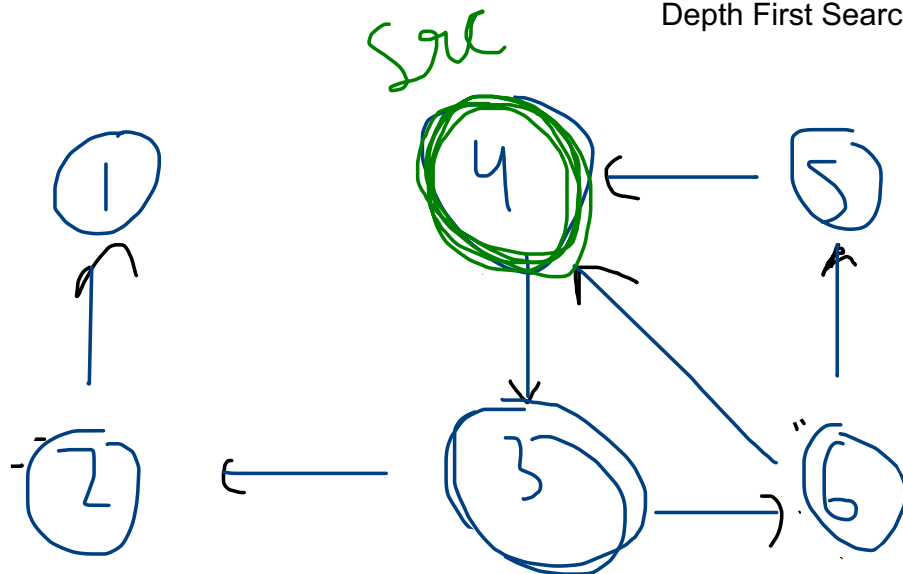
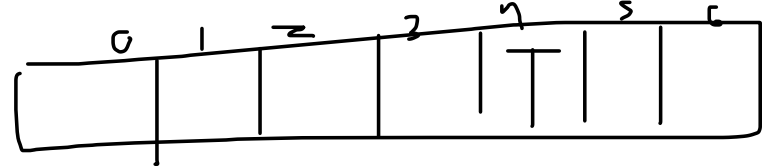


4 → 5 → 6 → 3 → 2 → 1

4 → 6 → 5

4 → 6 → 5 → 2 → 1

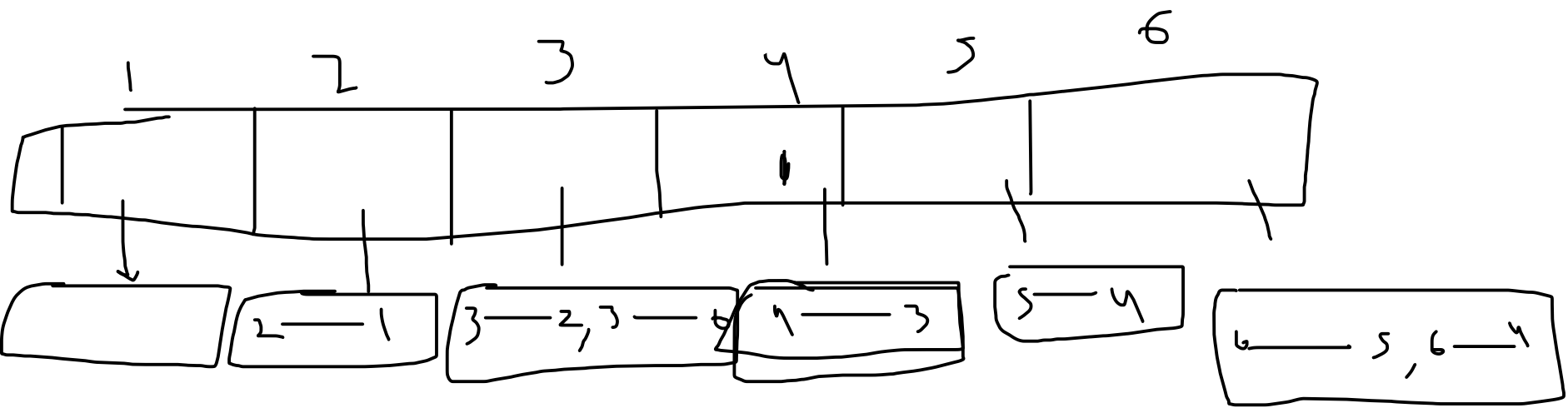
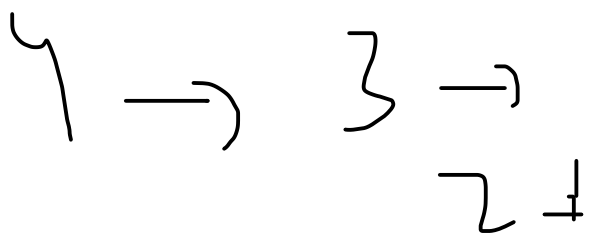
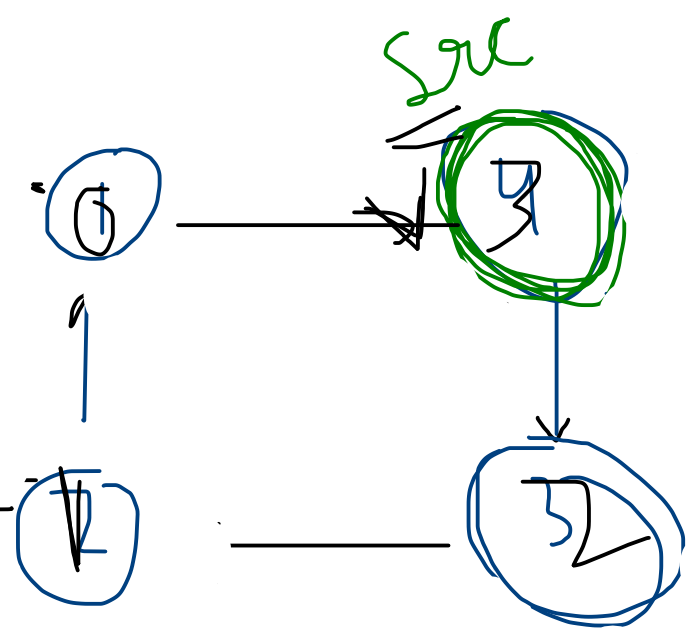
Depth First Search

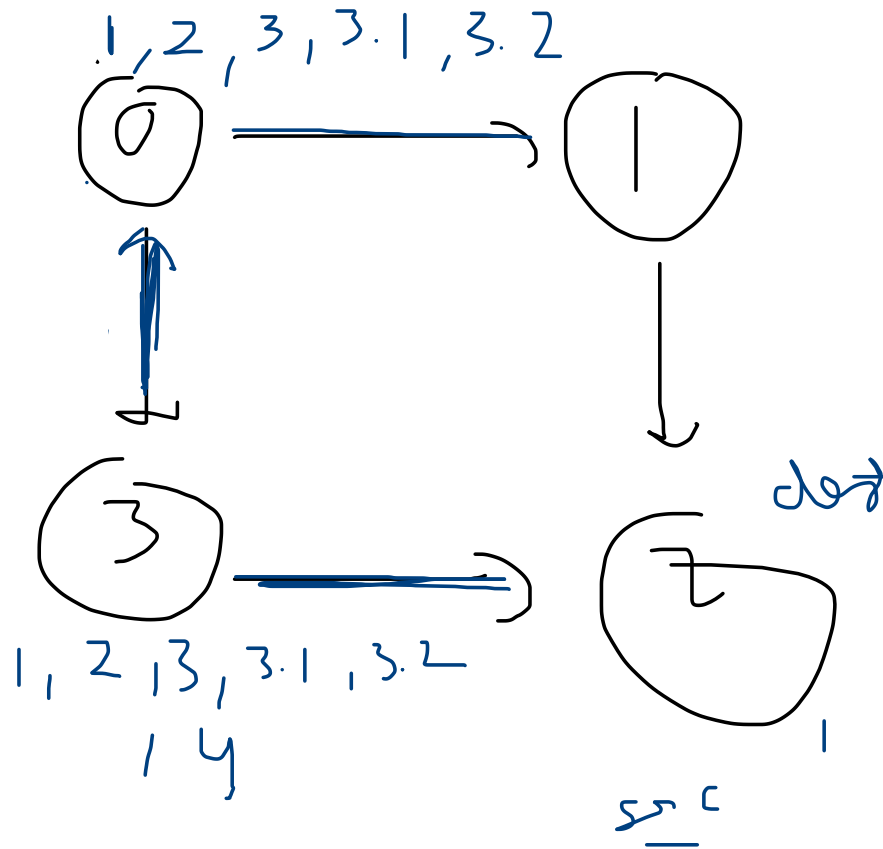


4 → 3 → 2 → 1

4 → 3 → 6 → 5

Depth First Search





```

if (srcVertex == dest){
    System.out.println(path);
    return;
}

visited[srcVertex] = true;

for (Edge e: graph[srcVertex]) {
    if (!visited[e.nbr]) {
        dfs(e.nbr, graph, visited, dest, path + e.nbr);
    }
}

visited[srcVertex] = false; // backtracking

```

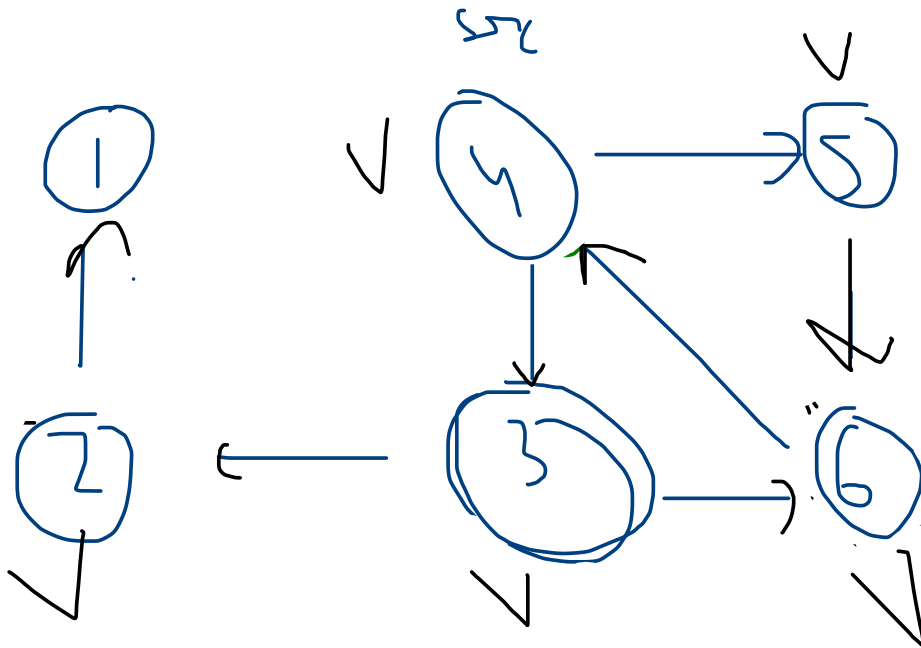
Yellow annotations on the code:

- 1 (next to `return;`)
- 2 (next to `visited[srcVertex] = true;`)
- 3 (next to `for (Edge e: graph[srcVertex]) {`)
- 1 (next to `if (!visited[e.nbr]) {`)
- 3.2 (next to `path + e.nbr`)
- 4 (next to `visited[srcVertex] = false;`)

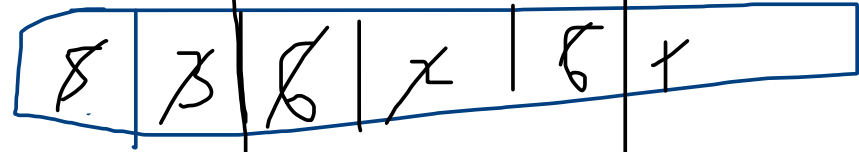
0	1	2	3
T	F	F	F

$P = 0 \ 3 \ 2$

BFS (Breadth-First Traversal) of Graph



Radially searching



Handwritten notes below the queue structure, including the numbers 6 and 7, and a vertical line.

