Lec-45

01/09/24 1:06 PM

Topological DFS

```
public void TopologicalSort() {
    Queue<Integer> q = new LinkedList<>();
    int[] in = indegree();
    for (int i = 0; i < in.length; i++) {
        if (in[i] == 0) {
            q.add(i);
        }
    }
    while (!q.isEmpty()) {
        int rv = q.poll();
        System.out.print(rv + " ");
        for (int nbrs : map.get(rv).keySet()) {
            in[nbrs]--;
            if (in[nbrs] == 0) {
                q.add(nbrs);
            }
        }
    }
    System.out.println();
}
```

```
public void Bellmanford() {
    int v = map.size();

    int[] dis = new int[v + 1];
    for (int i = 2; i < dis.length; i++) {
        dis[i] = 9799899;
    }
    List<EdgePair> ll = getallEdges();
    for (int i = 1; i < v; i++) {
        for (EdgePair e : ll) {
            if (dis[e.e2] > dis[e.e1] + e.cost) {
                dis[e.e2] = dis[e.e1] + e.cost;
            }
        }
    }
}
```