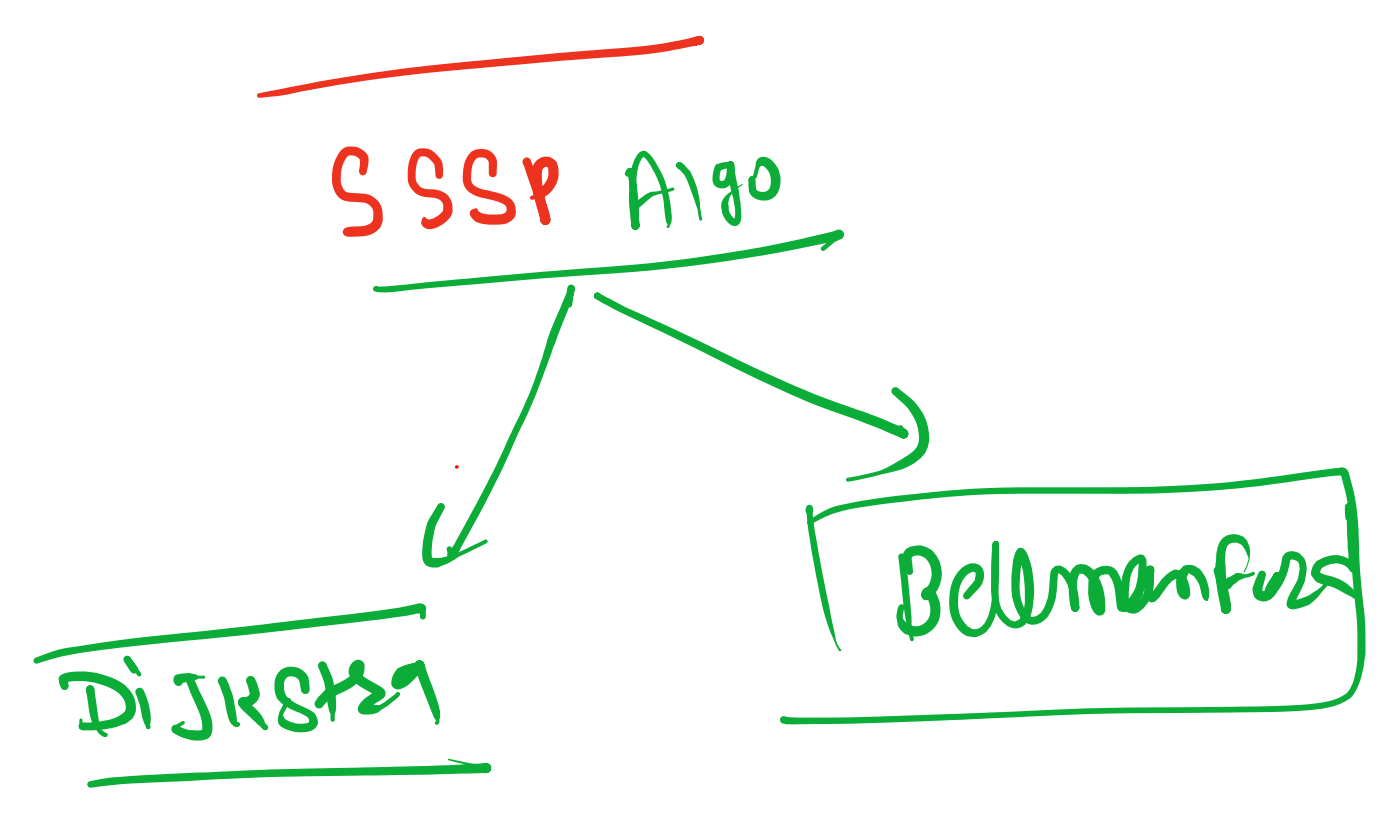
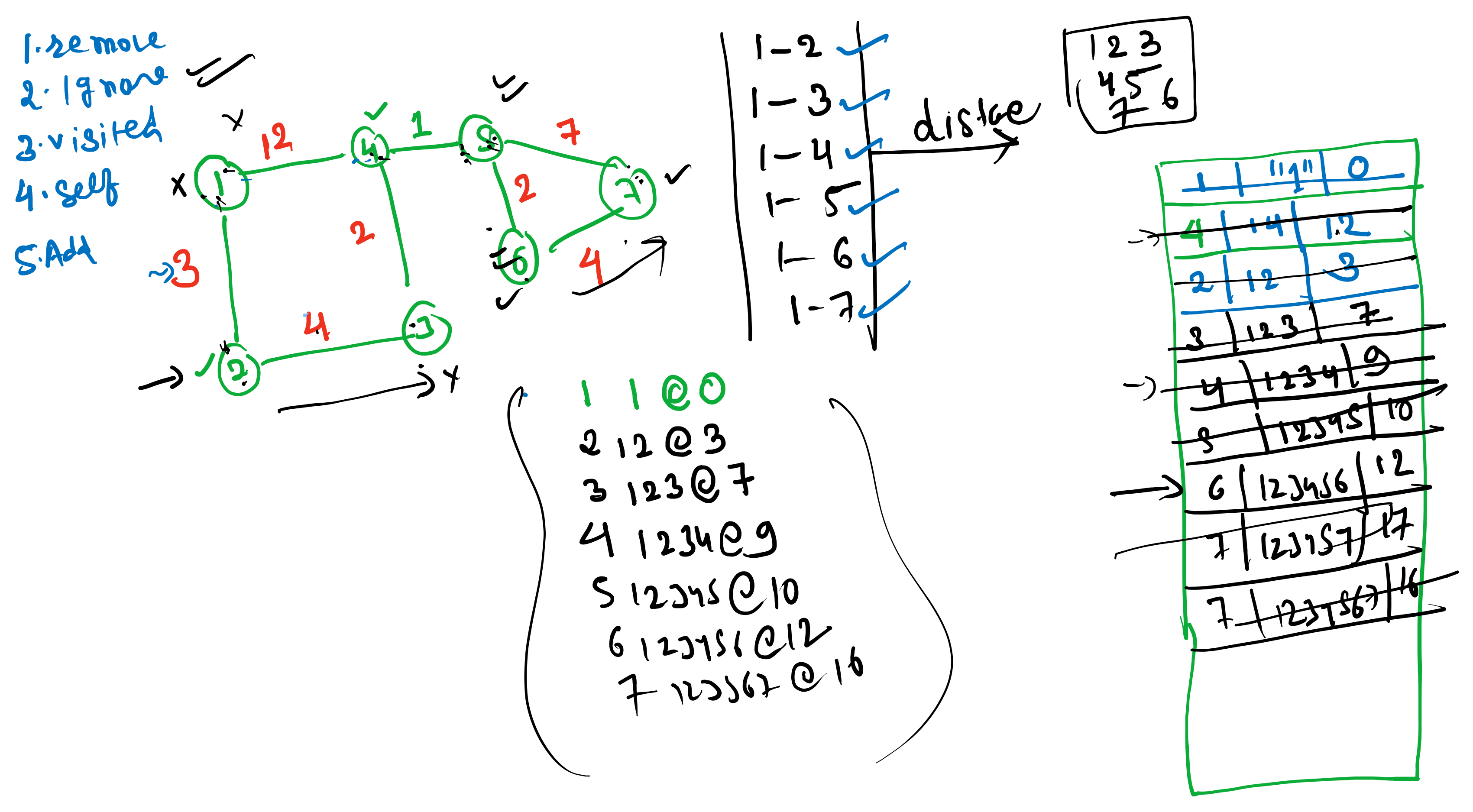
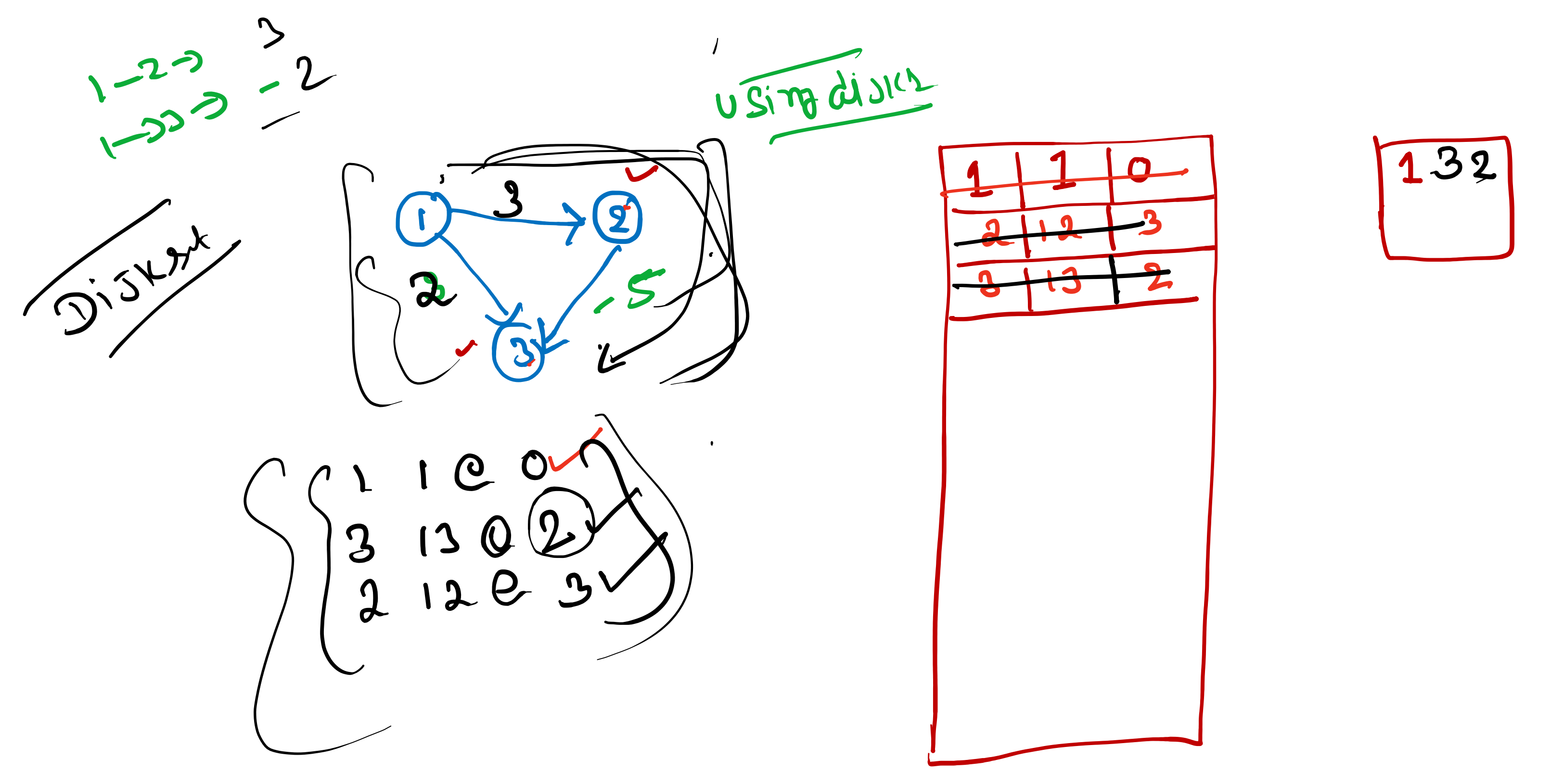
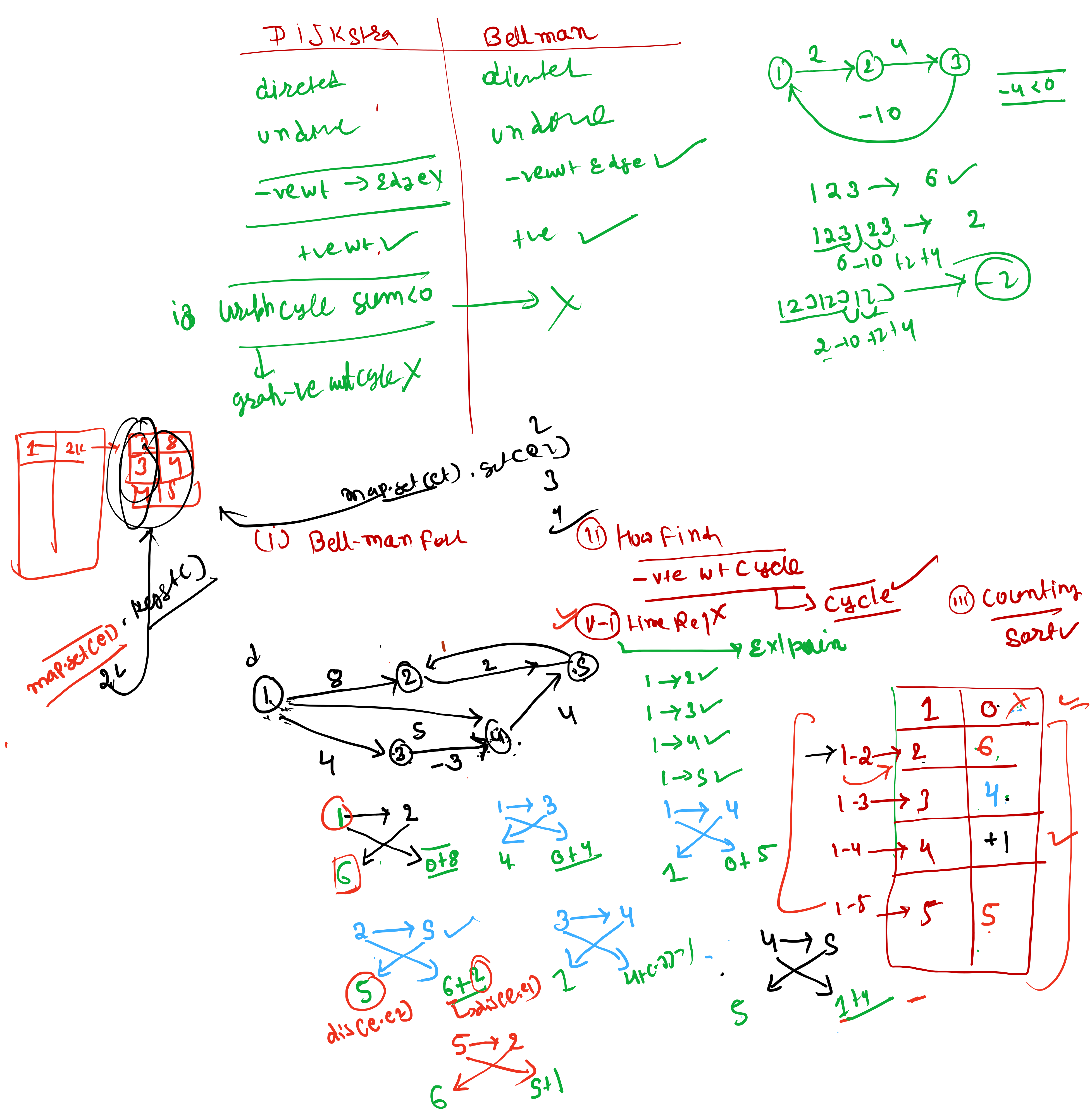
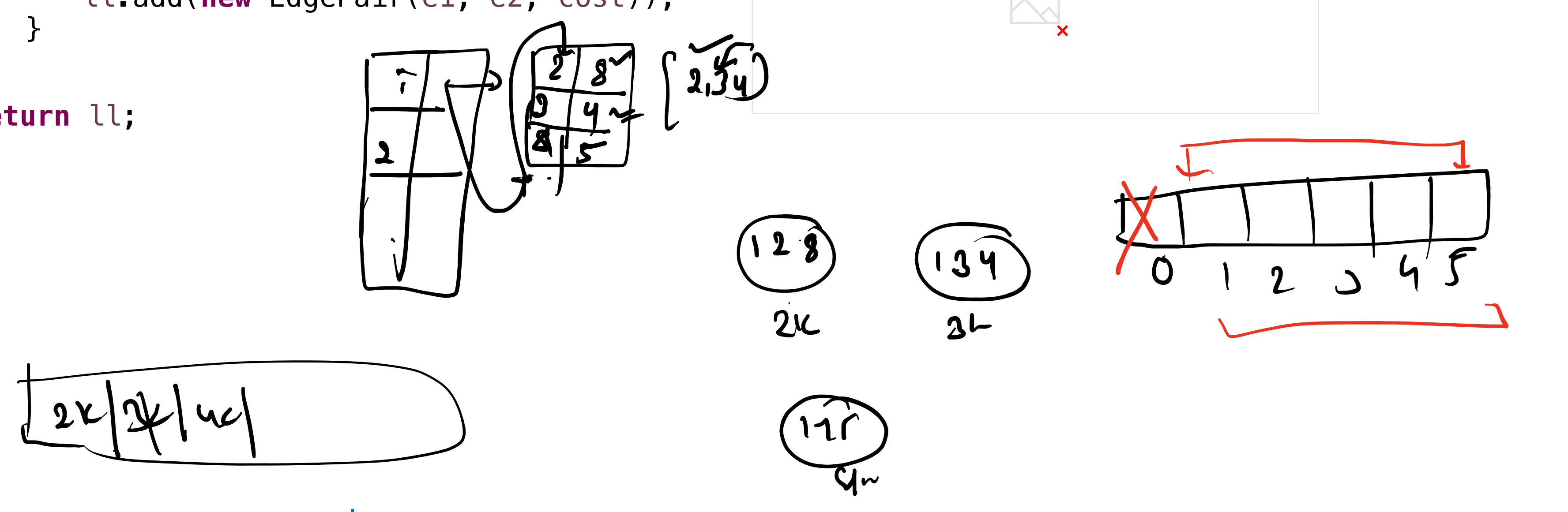


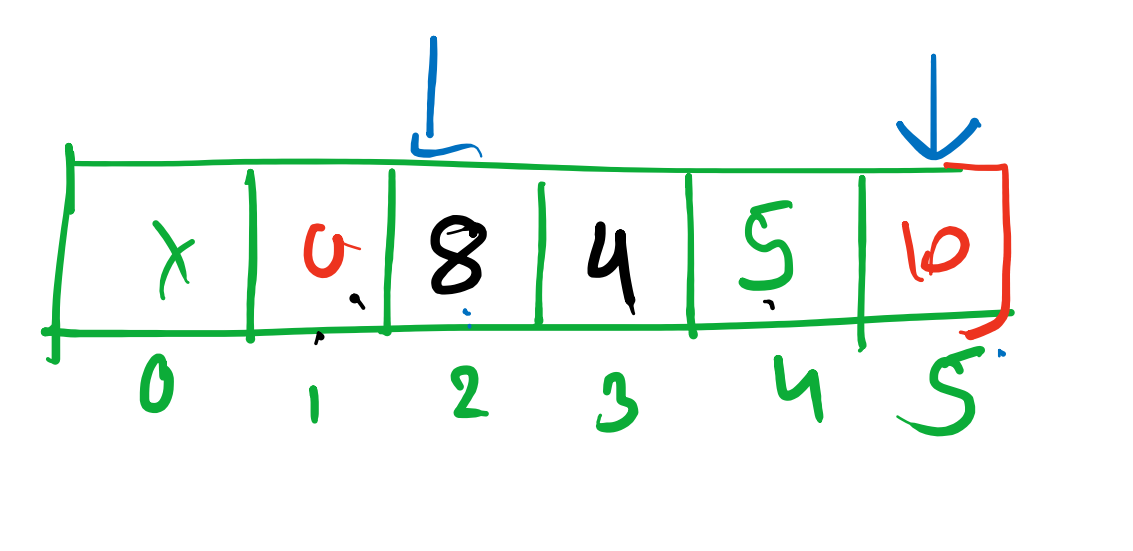




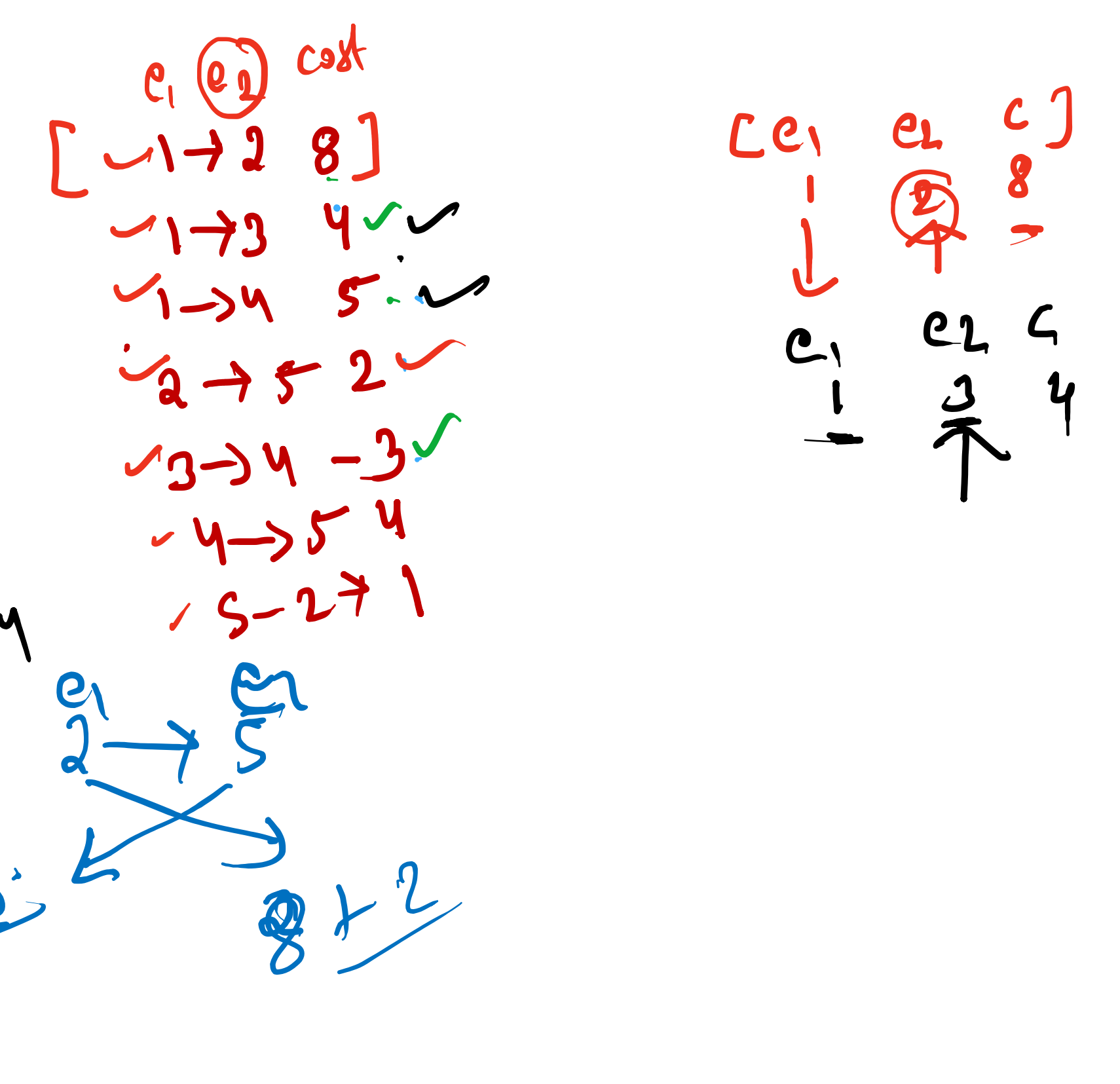


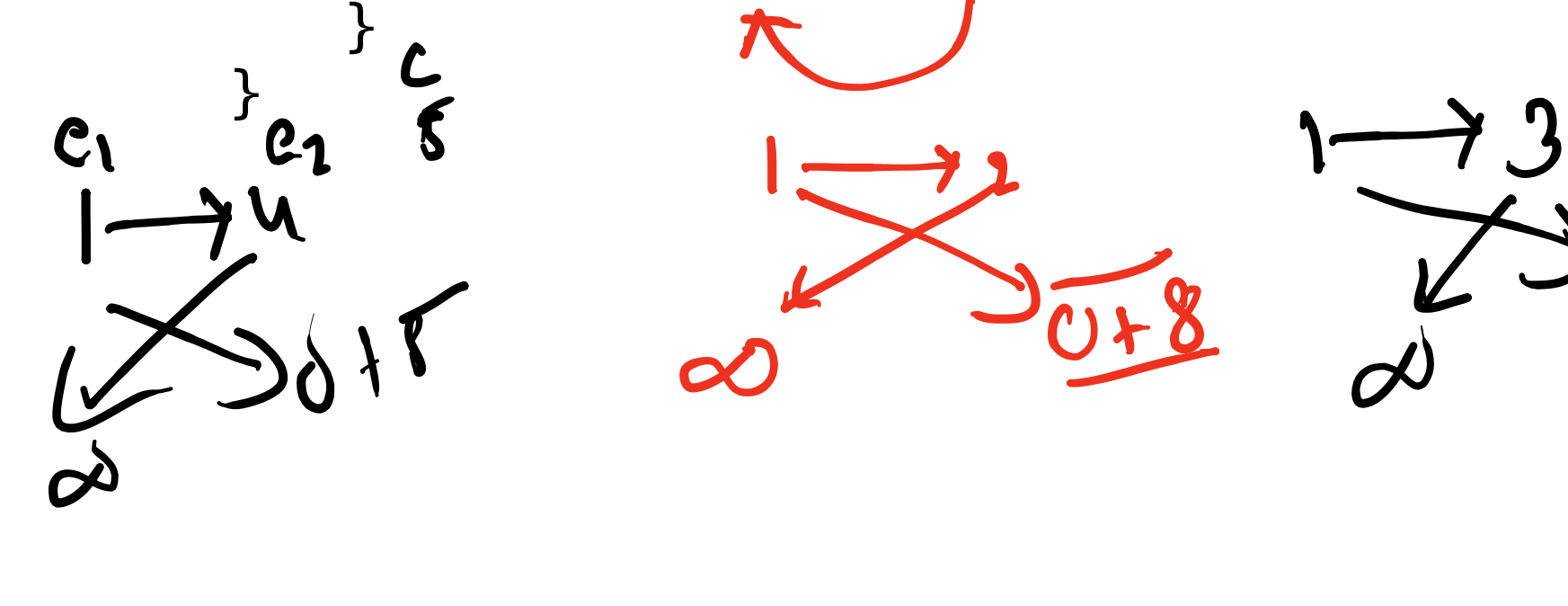
```java
public List<EdgePair> getAllEdge() {
        List<EdgePair> ll = new ArrayList<>();
        for (int e1 : map.keySet()) {// [1,2,3,4,5]
            for (int e2 : map.get(e1).keySet()) {
                int cost = map.get(e1).get(e2);
                ll.add(new EdgePair(e1, e2, cost));
            }

        }
        return ll;

    }
```



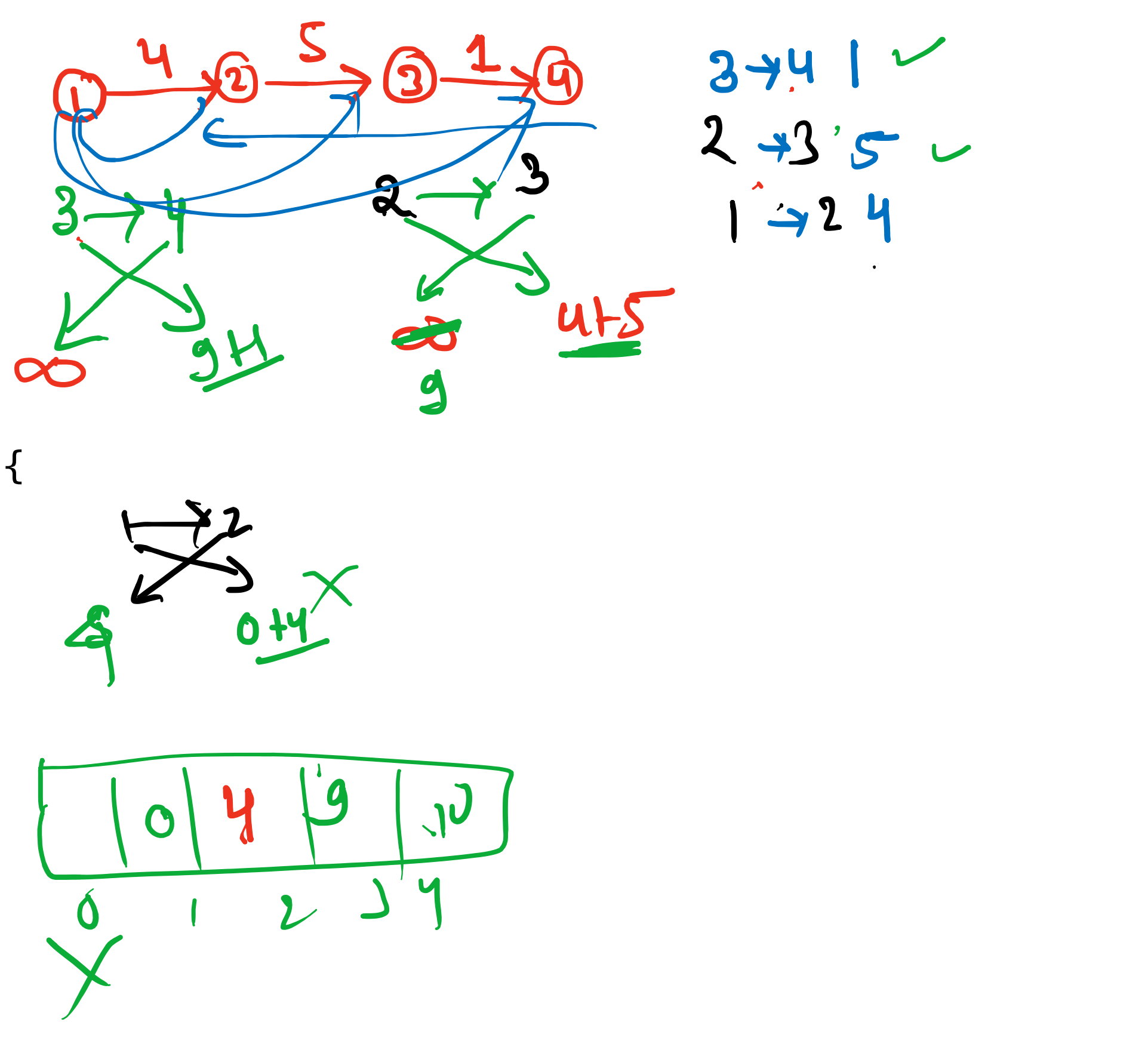


```java
for (EdgePair e : ll) {
        if (dis[e.e2] > dis[e.e1] + e.cost) {
            dis[e.e2] = dis[e.e1] + e.cost;
```

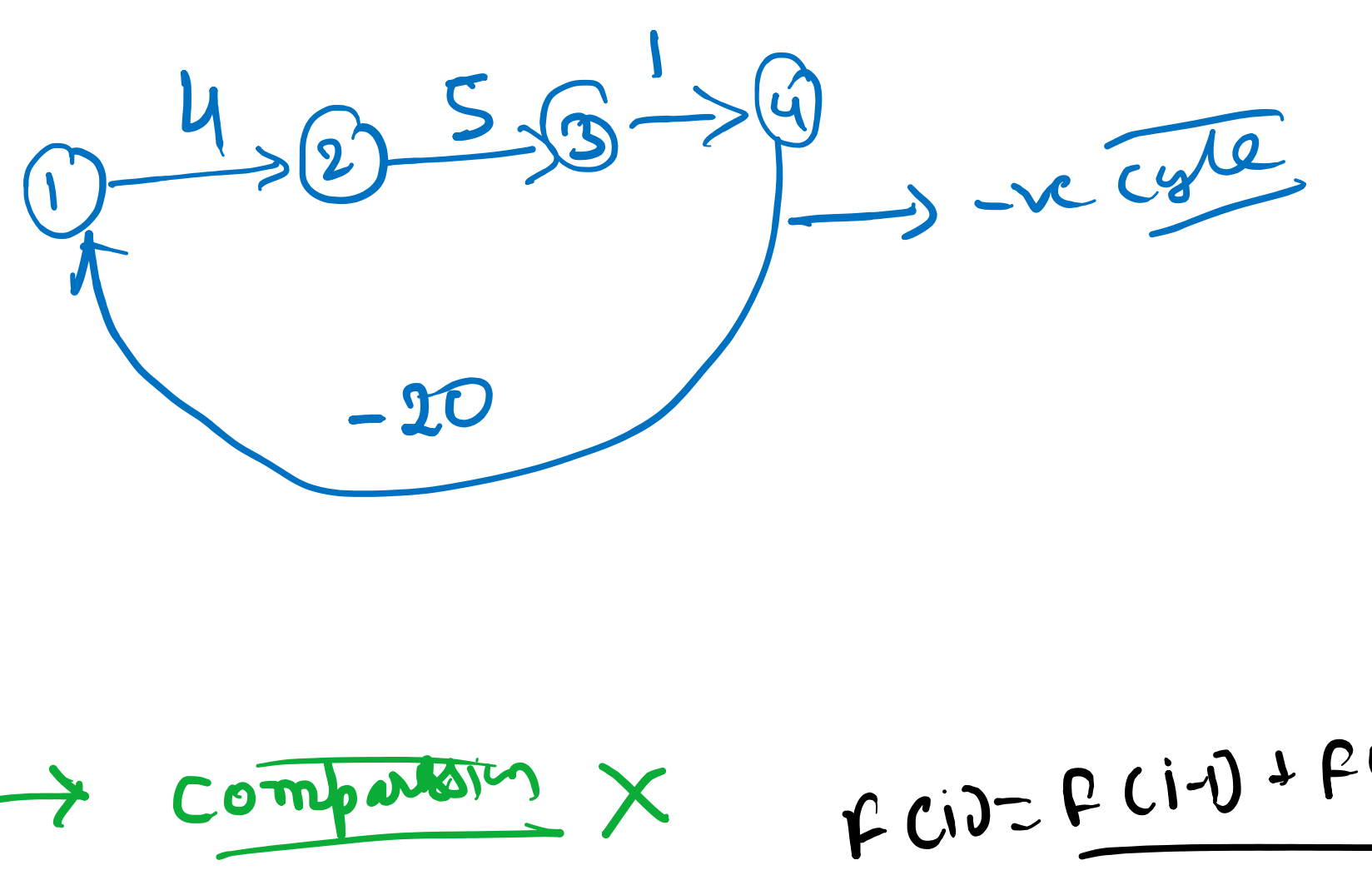




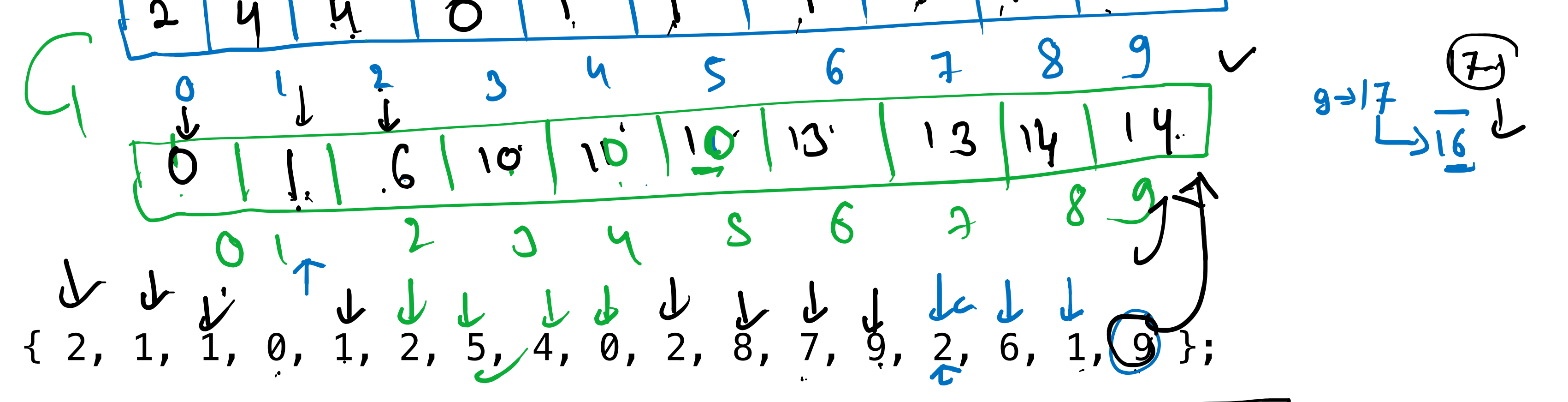
```java
public void BellManFord() {
        int v = map.size();
        int dis[] = new int[v + 1];
        List<EdgePair> ll = getAllEdge();
        for (int i = 2; i < dis.length; i++) {
            dis[i] = 87687679;
        }
        for (int i = 1; i <= v - 1; i++) {
            for (EdgePair e : ll) {
                if (dis[e.e2] > dis[e.e1] + e.cost) {
                    dis[e.e2] = dis[e.e1] + e.cost;
                }
            }
        }
        for (int i = 1; i < dis.length; i++) {
            System.out.print(dis[i] + " ");
        }

    }
```



-ve wt cycle



Counting Sort → Comparison X   F(i) = F(i-1) + F(i)



{ 2, 1, 1, 0, 1, 2, 5, 4, 0, 2, 8, 7, 9, 2, 6, 1, 9 };