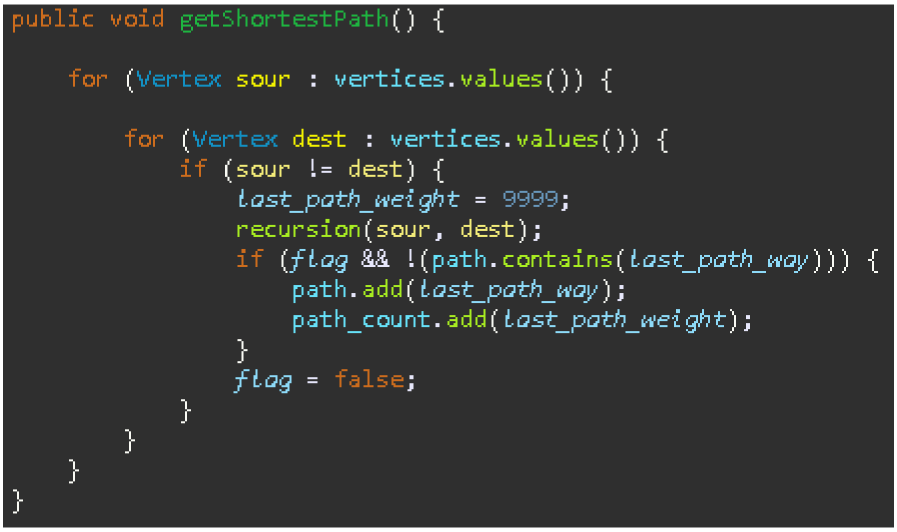
Report

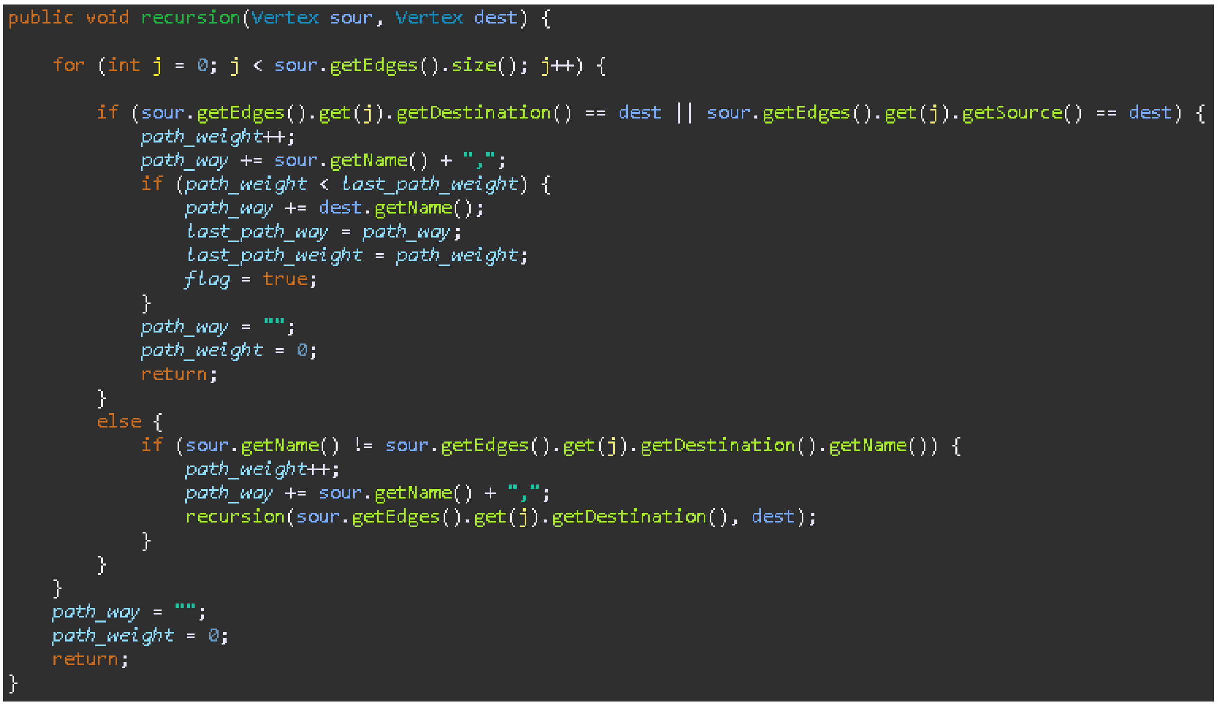
An algorithm that shortest path found on internet but could not implement to written code. Thus, the necessary algorithm has been written by Oğuz Kaan Şanlı. The algorithm had two loop which ‘for’ and included a recursion function. The algorithm executing slow due to the reasons.



```java
public void getShortestPath() {

    for (Vertex sour : vertices.values()) {

        for (Vertex dest : vertices.values()) {
            if (sour != dest) {
                last_path_weight = 9999;
                recursion(sour, dest);
                if (flag && !(path.contains(last_path_way))) {
                    path.add(last_path_way);
                    path_count.add(last_path_weight);
                }
                flag = false;
            }
        }
    }
}
```



```java
public void recursion(Vertex sour, Vertex dest) {

    for (int j = 0; j < sour.getEdges().size(); j++) {

        if (sour.getEdges().get(j).getDestination() == dest || sour.getEdges().get(j).getSource() == dest) {
            path_weight++;
            path_way += sour.getName() + ",";
            if (path_weight < last_path_weight) {
                path_way += dest.getName();
                last_path_way = path_way;
                last_path_weight = path_weight;
                flag = true;
            }
            path_way = "";
            path_weight = 0;
            return;
        }
        else {
            if (sour.getName() != sour.getEdges().get(j).getDestination().getName()) {
                path_weight++;
                path_way += sour.getName() + ",";
                recursion(sour.getEdges().get(j).getDestination(), dest);
            }
        }
    }
    path_way = "";
    path_weight = 0;
    return;
}
```