

Programming Assignment #3 report

B10901085 李昀儒

1. Algorithm

i. Undirected case:

The goal of undirected cases is to find the maximum spanning tree with $|V|-1$ edges, and these are the edges that will not be deleted. So we use Kruskal's algorithm with disjoint set to find the solution with time complexity $O(|E|\alpha|V|)$ (with union by rank and path compression implemented)

ii. Directed case:

There are 3 phases in the algorithm:

- Treat the graph as an undirected one, then find its maximum spanning tree with Kruskal's algorithm and mark these edges.
- For the unmarked edges, if the weight of the edge ≤ 0 , delete it.
- For the rest of edges that are neither marked nor deleted, check these edges decreasingly that if marking the edge will create a cycle among the marked edges, if it doesn't, mark it.

The time complexity is $O(|E|(|V|+|E|))$ since it performs DFS to check if the edge can be marked in phase c.

2. Data structures used

- Use stack for DFS and disjoint set to implement Kruskal's algorithm.
- Define two struct, edge and vertex, to store the information needed.