Algorithm PA3

林桓鈺 B10502013

The cycle breaking problem aims to transform a graph with cycles into an acyclic graph with minimal total edge weight removal. For unweighted or weighted undirected graphs I use different method to solve the problem. Following is the introduction.

Undirected Graph

- I use Kruskal's Algorithm to solve the problem. First, I sort the edges in non-increasing order instead of non-decreasing. When we run the algorithm, we remove the edges with the lightest weight, which guarantees that the graph does not contain any cycle. I save the removed edges in a vector `A`. After finishing the algorithm, we push the other edges that were not removed into `A` and return it. I save the number of removed edges in `remove_edge_number` and the total weight removed in `remove_edge_weight`.
- For the undirected graph, `A` is the solution we need. I write the `remove_edge_weight` and the first `remove_edge_number` elements into the output file.

Directed Graph

- I use the same method for undirected graphs, first running Kruskal's Algorithm. However, since the algorithm cannot determine the directionality between nodes, additional verification is needed. I use DFS to check if the edges obtained from Kruskal's Algorithm will form cycles under directed conditions. It's important to note that if the weight of an edge is negative, it does not need to be checked for cycles and can be directly removed. Removing a negative weight edge can reduce the total removed weight, which is beneficial for optimization.

- Before running DFS, I create an adjacency list that does not include the edges removed by Kruskal's Algorithm. When checking if an edge will form a cycle, I push it to the adjacency list and do DFS. If a back edge is found during the DFS process, it indicates a cycle, and the edge is removed from the adjacency list and stored in `temp`. If no back edge is found, indicating that the edge does not form a cycle, it is not added to `temp`. Finally, the value in `temp` will be our required solution.

Conclusion

- In this PA3 assignment, I spent an afternoon completing the undirected graph portion. However, handling the directed graph took much longer. I spent a lot of time researching and discussing various algorithmic approaches with my classmates. Ultimately, I managed to complete the assignment during the very hectic final exam week. Thank you to the professor and TAs for all your hard work this semester. Taking this course has been very beneficial for me.