# Homework #4

Due Time: 2022/06/17 00:00

Contact TAs: algota@noj.tw

## Introduction and Rules

1. The judge system is located at https://noj.tw or https://v2.noj.tw, please submit your code by the deadline.
2. 0215 Slides. (About Homework)
3. Can I refer to resources from the Internet or other sources that are not from textbooks or lecture slides?
   - Yes, but you must include a reference source, such as a website or book title and page number, and attach it as a comment at the top of the code.
   - Although you can refer to external resources, please write your own code after the reference.
   - Remember to specify the references; otherwise we'll view it as cheating.
4. The Non-Programming part is no need to hand in. Just for practicing!

# Programming Part

Please go to Normal Online Judge to read the programming problem.

# Non-Programming Part

## 1. Turnip Delivery

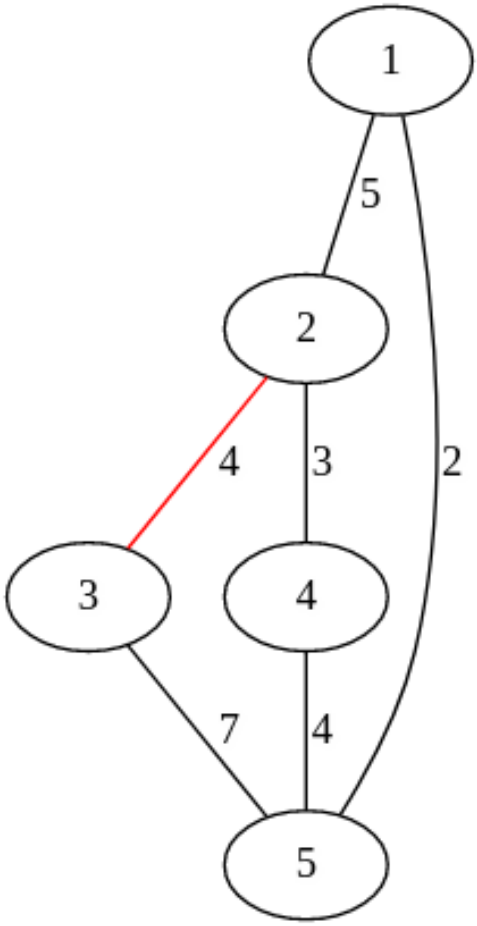
In Animal Crossing, there are many islands, and some of them can reach each other directly by airplane.

An airplane connects two different islands and has a weight limit of $w$. You want to take some turnips to another island, but the total weight of turnips must not be higher than $w$, or the airplane will crash.

What is the maximum weight of turnips you can deliver from island $A$ to island $B$ ?

It's guaranteed that all islands are connected, that is, you can reach any island from any other one by some paths.

For example, in *Figure* 1, the maximum weight of turnips we can deliver from 1 to 5 is 4.




There are several paths from 1 to 5:

1. $1 \rightarrow 5$: The maximum weight of turnips we can deliver through this path is 2.

2. $1 \to 2 \to 4 \to 5$: The maximum weight of turnips we can deliver through this path is 3.

3. $1 \to 2 \to 3 \to 5$: The maximum weight of turnips we can deliver through this path is 4.

So, the maximum weight of turnips you can deliver from 1 to 5 is 4.

**(1).**

Given a connected graph with $N$ nodes, how many edges are in the maximum spanning tree of this graph?

**(2).**

Please draw down the maximum spanning tree of the graph in *Figure* 1.

What is the relationship between MST and this problem? After finding out the MST, the path from $A$ to $B$ is unique. Moreover, the answer (the maximum weight of turnips) is the minimum weight limit airplane on the path from $A$ to $B$.

**(3).**

Try to explain the reason why the answer (the essential airplane) is always in MST. What properties of MST can help you to solve this problem?

## 2. Graph Quiz

**(1).**

With a graph $G = (V, E)$ without multiple edges, prove or disprove $O(|V| + |E|) \in O(|V|^2)$.

**(2).**

Find a graph with negative edges and apply Bellman-Ford and Dijkstra algorithm on it.

The answer graph must imply Dijkstra is not correct on a graph with negative edges.

In other words, you should calculate the correct and wrong answer. (The number of vertices in your graph should be no greater than 5.)

**(3).**

Prove or disprove after applying the Dijkstra algorithm on a graph formed by distinct weight edges, and we will get a minimum spanning tree got by Prim's algorithm.

And draw three example graphs, one is the original graph, another is the graph after applying Dijkstra, and the other is the minimum spanning tree got by Prim's algorithm.

The number of vertices in your graph should be no greater than 5. You also should specify the weight of every edge and the ID of every vertex.

## 3. Flow Quiz

FORD-FULKERSON-METHOD$(G, w, s)$
1 initialize flow $f$ to 0
2 **while** there exists an augmenting path $p$ in the residual network $G_f$
3 augment flow $f$ along $p$
4 **return** $f$

**(1).**

Please calculate the time complexity of pseudo code above when "finding augmenting path" is implemented by "DFS" and the graph is stored by adjacency List.

**(2).**

Please calculate the time complexity of pseudo code above when "finding augmenting path" is implemented by "BFS" and the graph is stored by adjacency List.

**(3).**

Follow the previous problem, please explain why causes the difference of time complexity between two flow algorithm.