CS2040S: Data Structures and Algorithms

Discussion Group Problems for Week 11

For: Oct. 31, 2019

hubs are connected components, so there are k components
if 2 cities are in a hub they are in the same component

Problem 1. Urban Planning Nightmares: Part II

Oh no! Mr Dnivog is back with even more urban planning issues for you to fix!

Problem 1.a. You have n cities. You want to collect the cities into "hubs" by linking them with highways. That is for any pair of cities in each hub, there are highways between the two cities. Given that you want to construct k hubs and given that the cost of building a highway between every two cities, determine the minimum cost to connect the entire city.

run kruskal then delete the most expensive edge k times or just run until you have k components - each time you run the number of components should decrease by 1. Cannot run Prim's algo directly because for Prim's you're choosing a 'random' source instead of the smallest

Problem 1.b. While creating a road network of minimum cost (in the form of spanning tree), you realise that the costs of all the roads round up to a very inauspicious number. Knowing that your supervisor is a very superstitious person, you decide to rectify this by finding an alternative spanning tree network that has the next best cost. However, you need to hurry as you need to propose this new network and its costs to the boss next morning! How can you do this as quickly as possible?

for every edge in current MST:
kick that edge out
look at all edges crossing cut
pick next best. If still inauspicious continue

Problem 1.c. After fixing the road network in the previous problem, you realise you have made a grave error! You forgot to add one more node in your network to finish your entire spanning tree network. Can you quickly determine the new spanning tree network with the new node added?

edges alr in MST = V - 1
edges that are new = V
total edges = 2V
Time: O(ElogE)

cut property - find smallest edge that connects alr formed MST and the node.
might end up changing MST when the new node is added. Consider the case where 1) u add the new node and there's a cycle
Solve: remove heaviest edge that causes the cycle by inspecting each edge - O(V^2(V))
2) you add the smallest edge connecting to one of the nodes but it might not make it an MST. Reconstruct the tree??? O(Vlog2V)

Problem 2. Cable Lines

Managing a computer network is hard. It is even harder when the network sometimes start overheating. You suspect that this is because of the existence of cycles within the network. You decide to monitor edges that are in cycles to see whether this hypothesis is correct.

Problem 2.a. Given a network, you want to select a set of links(edges) S such that for every cycle C in the graph there is at least one link $l \in C$ such that $l \in S$ as well. These are the edges that you will use to monitor the temperature of the links. However, putting temperature sensors are expensive. Given the cost c_l to install a temperature sensor at link l, determine the minimum cost for you to get a valid set S from your network.

cycle property: for every cycle in the graph, the most expensive edge in the cycle is not in the MST
Find maximum spanning tree, then cycle property is the opposite - least expensive edge in cycle is not in MST. Then choose from the set of least expensive edges. Either flip comparator or negate values

Problem 2.b. After you have measured the temperature variances of various links, you are able to determine which links have the propensity to heat up quickly and which will not be affected by high network traffic.

For every link l, let t_l represent the temperature the link when carrying network traffic. For any path P between two nodes u and v, you wish for the maximum temperature of all the links you pass through to be a minimum. We shall call this path the *coolest path*

MST

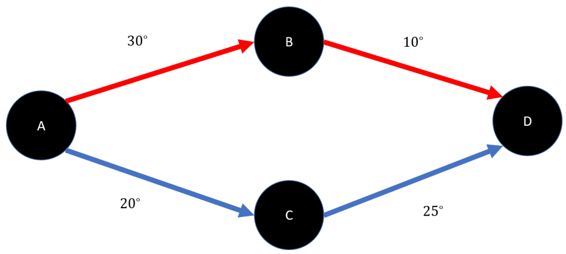

Figure 1: Example Graph

For example in the graph above, if we wish to send information from A to D, the path $(A \rightarrow B \rightarrow D)$ will be preferred, since the maximum temperature reached by any link in the path is $20°$ instead of the $30°$ it will reach on the path $(A \rightarrow C \rightarrow D)$.

Given the network and two points to send data from and too, find the *coolest path* between the two points.

Problem 3. Minimum Spanning Tree

Next, we will attempt the **Minimum Spanning Tree** problem on Kattis

`https://nus.kattis.com/problems/minspantree`