EXERCISES TO ACCOMPANY GRAPH ALGORITHMS: TREES, PATHS, AND CENTERS

R. TRAYLOR

§1. Maximum Branching

Problem 1 (*Optimization Algorithms for Networks and Graphs*, Minieka, 1978)**.** Suppose we need to construct a pipeline system to connect seven refineries $R1 \ldots R7$ with the port facility P receiving crude oil. The cost of building the pipeline between any two points is \$1000/mile plus a \$4000 setup cost for each segment. The distances between all pairs of points is given in the table below, which is upper triangular due to symmetry:

	P	R1	R2	R3	R4	R5	R6	R7
P	0	5	6	8	2	6	9	10
R1		0	4	10	5	8	6	10
R2			0	11	8	4	9	10
R3				0	10	3	6	7
R4					0	2	5	9
R5						0	10	5
R6							0	8
R7								0

(a) Explain why this is a maximum branching problem.
(b) Could you use a spanning tree algorithm on this? Why or why not? If you can use a spanning tree algorithm, what kind of spanning tree are you seeking?
(c) *Challenge:* Find the least cost pipeline system.

Problem 2. Can you think of a networking problem (that was not listed in the presentation) that can be reframed as seeking a maximum or minimum branching (or spanning arborescence)?

§2. Path Problems

Problem 3. Give a networking example where one would seek a path with the minimum number of arcs, regardless of weight on the arcs.

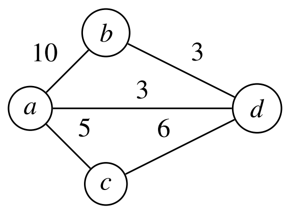

Figure 1. G

Problem 4. Find the shortest $b-c$ path, by hand, using Dijkstra's algorithm. Show all steps. (You should be able to do this by inspection, but work the algorithm out by hand to be sure you understand it.)

Date: October 8, 2020.

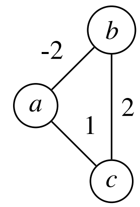

FIGURE 2. G

Problem 5. Implement Dijkstra's algorithm on the graph above by hand. Why does Dijkstra's algorithm fail when negative weights are allowed?

Problem 6. Suppose after running Dijkstra's algorithm on the graph of Problem 4, you discover that there was actually a missing (b,c) arc with weight 8. Is it possible to salvage these results, or must Dijkstra's algorithm be re-run? In general, if a new arc is discovered after Dijkstra's algorithm is completed, is it necessary for the algorithm to be re-run? Why or why not?

Problem 7. Use Floyd's algorithm to find the shortest path between any two pairs of vertices for the graph in Problem 4.

Problem 8 (*Optimization Algorithms for Networks and Graphs*, Minieka, 1978)**.** Here's an interesting example of a path problem that isn't obviously a path problem.

A hotel manager must make reservations for the executive suite for the coming month. He has received a variety of requests for various combinations of arrival and departure days. Each reservation would earn a different amount of revenue for the hotel since the rates are different for corporate, member, or regular clients.

(a) How can Dijkstra's algorithm be used to find the best way to schedule the bridal suite with maximum hotel profit?
Hint: You could represent each reservation request by an arc that joins the arrival date to the departure date. It is possible to prove that the resulting digraph will contain no circuits.
(b) Create a small dataset and try this out.
(c) Can you think of another "oddball" situation that can be reframed as a shortest-path problem?

Problem 9. Given the following shortest paths, reconstruct the network. The graph is undirected.

Path	Cost
AB	10
AC via ABC	12
AD	8
AE	5
BD via BCD	5
BE	11
CE via CAE	7
DE via DAE	7

Problem 10. For the graph of problem 3, find
(a) Center
(b) Median
(c) General Center
(d) General Median

Problem 11. Try to find examples in networking where you would be seeking
(a) Center
(b) Median
(c) General Center
(d) General Median

Problem 12. Draw your own 5-node network, assign weights, and find the locations as in problem 10. How do your results change when you direct the graph?