

Homework 3

Henry Z. Lo

Due: Monday, July 21 in class

This homework assignment will review some graph algorithms that we've studied. For these problems, write pseudo-code; your algorithms should contain enough detail to implement. Assume that the following can be done in constant time:

- For a node n , find which nodes have edges going into it.
- For a node n , find which nodes n has edges going to.
- Getting or checking if a specific edge exists.

When writing pseudocode, do not write anything high-level like "topological sort" or "depth-first traversal". I want detailed line-by-line pseudocode of how these procedures are actually done.

1. Recall that Dijkstra's algorithm for finding the shortest path between nodes can be done in $O(V + E \log V)$ time.
 - (a) In a directed acyclic graph, we can find the shortest path even faster. Describe this faster algorithm.
 - (b) How fast is it? Explain line by line.
2. Suppose you want to find the shortest path from the corner of one city block to another. Each corner is connected to adjacent corners, with a distance of one city block width.
 - (a) Describe an algorithm to find the length of this shortest path.
 - (b) What heuristic can you use to speed up finding this path?
 - (c) Modify your algorithm to use this heuristic.
3. The delivery boy notices that there are V locations he commonly brings pizza to, and wants to calculate the total driving time between every pair of these locations. These locations form a graph, where edge weights represent driving time.
 - (a) Use transitive closure to calculate the least driving time from every location to every location.

- (b) This algorithm uses what (non-graph) technique that we have studied?
- 4. Introverts, Inc. wishes to make every employee feel as least awkward as possible. Every communication between employees i and j incurs an awkwardness cost c_{ij} to the receiving employee. When the CEO sends out a memo, it must reach every employee, but in a way that the total awkwardness felt by all employees is as low as possible.
 - (a) What type of structure should the network take to minimize total awkwardness? Consider that not everyone feels comfortable talking to the CEO.
 - (b) Give the pseudocode for this algorithm.
 - (c) Employee a and employee b were dating, but have broken up, and now refuse to talk with each other. Give an algorithm which makes sure that every employee can still get their memos.
 - (d) How fast is this algorithm?