Reading


Exercises

1. **Algorithms and Properties**
   1.1 (w) CLRS 24.3-1 (also show the contents of the priority queue).
   1.2 (w) CLRS 24.2-1
   1.3 CLRS 24.3-2
   1.4 CLRS 24.3-4
   1.5 Let \( T \) be a shortest path tree from a node \( s \) in a graph \( G \). Assume we add a constant \( c \) to all edge weights in \( G \). Is \( T \) still a shortest path tree?

2. **Cable Routing** Exercise 3 from the 2012 exam set (respectively 02326 and 02105)

3. **Longest Paths in DAGs** Give an algorithm to find the longest path in a DAG.

4. **BSc Negative Edges** Explain where in the proof of Dijkstra’s algorithm we use that edge weights may not be negative.

5. **Node Weighted Dijkstra** Let \( G \) be a directed graph where all nodes are associated with a non-negative weight. The weight of a path in \( G \) is the sum of the weights of the nodes on the path. Give an algorithm to compute the shortest path between two nodes in \( G \).

6. **[*] Zombie Travel** In the post-apocalyptic zombie world you need to know the safest travel between two cities such that you hopefully avoid being eaten by the zombies. You are given a graph \( G \) where each node represents a city and each edge a road between two cities. Each edge \( e \) has a probability \( s(e) \), \( 0 \leq s(e) \leq 1 \) for surviving traveling on that edge without being eaten. The probabilities on each edge are independent and the probability of surviving the entire travel along a path \( P \) is the product of the probabilities of surviving on each edge of \( P \).

As an example look at the above graph. If you travel directly from node 2 to 4 you have 50% chance of surviving. If you instead travel via node 3 you have \( 0.7 \cdot 0.9 = 63\% \) chance of surviving. If you travel via 3 and 1 you only have \( 0.7 \cdot 0.1 \cdot 0.5 = 3.5\% \) chance of surviving. Give an algorithm that computes the safest way from a node \( s \) to another node \( t \).
7  **Loopy Trees**  A *loopy tree* is a weighted directed graph constructed from a binary tree by adding an edge from each leaf to the root. All edges have non-negative weights.

7.1 How long time does Dijkstra’s algorithm use to compute the shortest path from a node $s$ in a loopy tree?

7.2 [∗] Give a faster algorithm.