ALGORITHMS, FALL 2018, HOMEWORK 13

Due Friday, December 7 at 11:59pm.
Worth 3% of the final grade.

1. You are given a graph $G = (V, E)$ and its MST. Suppose that you wish to add a vertex v to G, along with some weighted edges from v to other vertices in G. Show how to update the MST. Analyze the time-complexity of your algorithm, and argue correctness. You may assume that all edge weights are distinct.

2. Let G be a graph for which all edge weights are greater than 1. For two vertices x, y in G, we want to find a path from x to y that minimizes the *product* of edge-weights. Show how to do this.

3. An infectious disease starts at a node x in a weighted graph. The time it takes to move along an edge is equal to the weight of the edge. Whenever it gets to an unprotected node, that node gets infected and the disease keeps moving in parallel along all incident edges. In the beginning, all nodes are unprotected. However, at some node y in the graph, an antidote is released, at the same time as the disease. It moves along the graph just as the disease does, and if it reaches an uninfected node, it protects that node forever.
   At any node that is reached simultaneously, the disease and the antidote neutralize (forever), and neither gets to keep traveling along incident edges. However, the node is left partially damaged.
   Show how to determine whether more nodes will be (fully) infected or protected. How fast is your algorithm?