Practice problems (don’t turn in):

1. [DPV] 5.11 union-find data structure implementation – **do without path compression**.  
   Note, we may not cover in class until Tu. 10/29.

2. Min-heap (aka priority queues):
   Starting from an empty Min-heap, perform the following sequence of operations, and draw the final Min-Heap data structure.
   
   (a) Insert a;7 (i.e., insert element a with key 7).
   (b) Insert b;4.
   (c) Insert c;9.
   (d) Insert d;12.
   (e) Insert e;10.
   (f) Insert f;3.
   (g) Decrease-key of e to 2.
Problem 1  [DPV] Problem 4.8 (Professor Lake’s shortest path alg. with negative weights)
Problem 2  [DPV] Problem 4.14 (shortest paths through \(v_0\))

A faster algorithm is worth more.

Hint: Use Dijkstra’s algorithm as a black-box. How many runs of Dijkstra’s algorithm do you need?

Algorithm: (Explain your algorithm in words)

Running time: (brief explanation required)

Correctness: (explain why your alg. solves the problem)
Problem 3 [DPV] Problem 5.23 part (b)

For $e \notin E'$ its weight is decreased. Update the MST.
(You can do part (a) as practice but don’t turn-in.)

Algorithm: (Explain your algorithm in words)

Running time: (brief explanation required)

Correctness: (explain why your alg. solves the problem)
Problem 4  [DPV] Problem 5.23 part (d)

For $e \in E'$ its weight is increased. Update the MST.
(You can do part (c) as practice but don’t turn-in.)

Algorithm: (Explain your algorithm in words)

Running time: (brief explanation required)

Correctness: (explain why your alg. solves the problem)