DSC 40B - Discussion 01

Problem 1.

Modify the `DisjointSetForest` data structure so that it keeps track of the maximum and minimum keys within each disjoint set without adding to the time complexity of any of the operations.

Problem 2.

The *fractional* knapsack problem is as follows. You have a bag that can hold B liters. In front of you are n piles of gold dust, silver dust, etc. The ith pile contains s_i liters of dust, and the dust in the pile is worth w_i dollars in total. You may choose any amount of dust from any pile to put in your bag (up to s_i). Your goal is to maximize the value of the dust in your bag.

Describe a greedy algorithm for solving this problem. Is it guaranteed to find the optimal answer?

Solution: Start by calculating the *price per liter* for each type of dust. Then, starting with the most expensive dust in dollars per liter, place as much as you can into your bag until 1) your bag is full or 2) the pile is gone. Then move to the next most valuable pile by price per liter and repeat.

This is optimal.