Advanced Algorithm: Homework 2

Problem 1. 16.1-2.
Suppose that instead of always selecting the first activity to finish, we instead select the last activity to start that is compatible with all previously selected activities. Describe how this approach is a greedy algorithm, and prove that it yields an optional solution.

Problem 2.
Find a largest subset of mutually compatible activities among the following.

\([1,4), [3,6), [0,7), [3,10), [5,11), [6,10), [8,12), [2,14), [11,12]).\)

Problem 3.
What is an optimal Huffman code for the following set of frequencies?
- a:1, b:2, c:3, d:5, e:8, f:13

Problem 4. 16.3-4.
Prove that we can also express the total cost of a tree for a code as the sum, over all internal nodes, of the combined frequencies of the two children of the node.

Extra Credit Problem. 16.2-3.
Suppose that in 0-1 knapsack problem, the order of the items when sorted by increasing weight is the same as their order when sorted by decreasing value. Give an efficient algorithm to find an optimal solution to this variant of the knapsack problem, and argue that your algorithm is correct.