CSCI 503B:
HOMEWORK 2

Each question has equal weight.

1. Consider the following variant of the activity selection problem: rather than scheduling the highest number of activities, we would like to schedule them (without overlaps) so that we maximize the time that the room is occupied. For instance, a single activity that lasts from the opening time to the closing (if such an activity exists) would be an optimal solution. My algorithm is to pick the longest activity that does not conflict with any others, then pick the next, etc. Once I'm done picking those, then I sort them and rent out the room starting with the earliest. For instance, if I have a room from 1pm to 11pm and the activites are: [1,4),[2,3), and [3, 9), I will first pick [3,9) since it is longest, then [2,3) since it's the only one that doesn't overlap with anything picked previously. Then the activities, in order, will be [2,3) and [3,9).

   Will this always give me the optimal solution for any activity set? Prove.

2. Prove the following. Let $a_1, a_2, \ldots, a_k$ be the set of activities that we picked using the greedy activity selection that we discussed in class (sorted according to their times), in the original problem. Let $b_1, b_2, \ldots, b_k$ be any other optimal solution to the same problem (sorted similarly). Given any a_i from the first group, consider the corresponding b_i from the second. Prove by induction that the finishing time of a_i is no later than that of b_i. I asked this question in class, if you remember, and mentioned that it's fair game for a homework or midterm. There you go...

3. Draw the Huffman tree for an alphabet consisting of the letters a, b, c, d, e, f, g, h, with the following frequencies respectively:

   $$0.1,\ 0.2,\ 0.4,\ 0.02,\ 0.04,\ 0.03,\ 0.04,\ 0.17$$

   What is the length of the code for a document with n symbols? What is the length per symbol? How much of an improvement is this compared to fixed length encoding?

4. These two questions are harder (and have less weight, since each one is 12.5 points). Exercise those brains... 16-1.2 from the book. 16-3.8 also. For the second problem, note that you need to argue you will be building a complete binary tree. You can do an induction on the levels of this tree. You start by combining two of the leaves. Then which two nodes will you combine? What can you say about the frequencies of the intermediate nodes that you are creating?