ECS122A Homework Assignment #4 **Due: 4:00pm, February 8, 2016**

1. Suppose that we have a set of activities to schedule among a large number of lecture halls. We wish to schedule all the activities using as few lecture halls as possible. Given an efficient greedy algorithm to determine which activity should use which lecture hall. Provide the required time in the worst case, and justify the efficiency of your algorithm.

2. We are given n jobs $j_1, j_2, \ldots, j_n$, with known running time $t_1, t_2, \ldots, t_n$, respectively. We have a single processor. We want to schedule these jobs so as to minimize the avarage completion time. Give a greedy algorithm for the job scheduling problem, and verify the optimality. (*We assume nonpreemptive scheduling, i.e., once a job is started, it must run to completion.*)

3. Suppose the characters a, b, c, d, e occur with frequencies 1/2, 1/4, 1/8, 1/16, 1/16, respectively.

   (a) What is the Huffman codes of the alphabet?

   (b) If this coding is applied to a file consisting 1,000,000 characters of a, b, c, d, e with the given frequencies, what is the length of the encoded file in bits?

4. (a) What is a Huffman code for the following set of frequencies, based on the first 8 Fibonacci number?

   `a:1, b:1, c:2, d:3, e:5, f:8, g:13, h:21`

   (b) Generalize your answer to find the optimal code when the frequencies are the first n Fibonacci numbers?

5. We use Huffman's algorithm to obtain an encoding of allphabet $\{a, b, c\}$ with frequencies f_a, f_b and f_c. In each of the following cases, either give an example of frequencies $\{f_a, f_b, f_c\}$ that would yield the specified code, or explain why the code cannot possibly be obtained (no matter what the frequencies are)

   (a) Code: $\{0, 10, 11\}$ (b) Code: $\{0, 1, 00\}$ (c) Code: $\{10, 01, 00\}$

6. The 0-1 knapsack problem. Given six items $\{(v_i, w_i)\}$ for $i = 1, 2, \ldots, 6$ as follows:

i	v_i	w_i
1	40	100
2	35	50
3	18	45
4	4	20
5	10	10
6	2	5

   and the total weight $W = 100$, where v_i and w_i are the value and weight of item i, respectively.

   Find the greedy solutions by using following strategies:

   (a) Greedy by value, i.e., at each step select from the remaining items the one with the highest value

   (b) Greedy by weight, i.e., at each step select from the remaining items the one with the least weight.

   (c) Greedy by value density, i.e., at each step select from the remaining items with the largest value per pound ratio v_i/w_i.

   Are these greedy solutions optimal? Comment your findings.