CS 477/677 Analysis of Algorithms

Homework 7

Due April 30, 2024

1. [50 points] (U&G-required)

A laundry store cleans laundry from n customers C_1, C_2, ... C_n in a particular day. The clothes from each customer must be first washed and then dried on the store's machines. However, for the current day all except one of the washing machines are scheduled for maintenance, thus there is only one washing machine available for use. After washing, there are sufficient drying machines to dry the clothes from all customers in parallel, and they are all operational. For customer C_i, the laundry needs w_i minutes to be washed and d_i minutes to be dried. The store needs to come up with a *schedule* in which the customers' clothes are to be washed in the single machine. The *cleaning time* of the schedule is the **earliest time by when all the clothes have been washed and dried**. Give a polynomial-time algorithm that finds a schedule with a **minimum *cleaning time***.

Note: to prove that your greedy strategy yields the optimal solution, you have to prove that the problem has the *greedy-choice property*.

2. [50 points] (U&G-required)

You are running a house cleaning company that services an entire building of n apartments. Your goal is to keep the apartment owners happy, so that they can continue using your services. On a cleaning day, the company needs to schedule the order in which the apartments will be cleaned. Each apartment i will take time t_i time to clean. The *finishing time* of cleaning apartment i is denoted with F_i. For instance, if apartment i is the first one cleaned, then $F_i = t_i$, and if it is cleaned second, after some apartment j, then $F_i = t_j + t_i$. The owners of the apartments each have a loyalty ranking r_i, indicating their importance to your business. The happiness of the apartment owners is influenced by the time F_i by which their apartment gets cleaned. Thus, your company aims to schedule the apartments for cleaning in such a way as to **minimize the weighted sum of the finishing times**, $\sum_{i=1}^{n} r_i F_i$. Design an efficient algorithm that, given a list of n apartments, with cleaning times t_i and loyalty ranking r_i, schedules the apartments for cleaning such that $\sum_{i=1}^{n} r_i F_i$ is minimized.

Example. Suppose there are two apartments: the first takes time t_1 = 20 minutes and has rank r_1 = 8, while the second apartment takes time t_2 = 50 minutes and has rank r_2 = 4. Then cleaning apartment 1 first would yield a weighted finishing time of 20*8 + (20+50)*4 = 160 + 280 = 540. while cleaning the second apartment first would yield the larger weighted finishing time of 50*4 + (50+20)*8 = 200 + 560 = 760.

Note: to prove that your greedy strategy yields the optimal solution, you have to prove that the problem has the *greedy-choice property*.

4. [20 points] (G-required)

Consider the coin changing problem discussed in class, with the following coin denominations: 1, 8, 12. Indicate if the greedy algorithm from class is optimal for these denominations. If the algorithm is optimal, give a brief justification, if it is not optimal give a counterexample.

5. [20 points] (Extra credit)

Use Huffman's algorithm to construct an optimal binary prefix code for the letters in the following table:

Letter	A	B	I	M	S	X	Z
Frequency	14	9	20	12	11	7	4