CMSC 451 Quiz 2

This quiz is closed-book and closed-notes. You may use any algorithms or results given in class. The total point value is 50 points. Good luck!

Problem 1. (15 points)

(a) (3 points) Another greedy strategy for the interval scheduling problem is to sort the intervals in decreasing order by *start times*. Repeatedly take the interval with the *latest start time*, that does not overlap with any previously scheduled intervals.

True or False: This strategy is optimal. (No explanation needed.)

(b) (3 points) Which of the following scheduling problems can be modeled as graph coloring? (Select one.)

(i) Interval scheduling (unweighted)
(ii) Interval partitioning
(iii) Scheduling to minimize lateness

(Recall that *graph coloring* is the problem of assigning a minimum number of colors to the vertices of the graph so that no two adjacent vertices have the same color.)

(c) (6 points) Which of the following assertions hold for Gonzalez's algorithm? (Select all that apply)

(i) The distance between the closest pair of centers never increases
(ii) For any point, the distance to its closest center never increases
(iii) If there are two centers at distance x from each other, then every point is within distance x of its nearest center
(iv) If there are two centers at distance x from each other, then every point is within distance $x/2$ of its nearest center

(d) (3 points) Consider a set system (X, S), where the universe X has a elements, there are b sets in S, and there are c sets in the optimal set cover. Which of the following best describes the approximation ratio of the greedy set-cover algorithm? (Select one)

(i) $\ln a$ (ii) $\ln b$ (iii) $\ln c$ (iv) $e^{-1/c}$

Problem 2. (15 points) A pharmacist has W pills and n empty bottles. Let p_i denote the number of pills that can fit in bottle i, and let c_i denote the cost of purchasing bottle i. Given W, p_i's and c_i's, we wish to compute the cheapest subset of bottles into which to place all W pills. (You may assume that $\sum_i p_i \geq W$, so there is a feasible solution.)

Suppose that you only pay for the *fraction* of the bottle that is used. For example, if the ith bottle is half filled with pills, you pay only $c_i/2$. Our goal is to find a greedy algorithm to put all the pills into some of the bottles, to minimize the total bottle cost. (See Fig. 1 for an example with 3 bottles and 8 pills.)

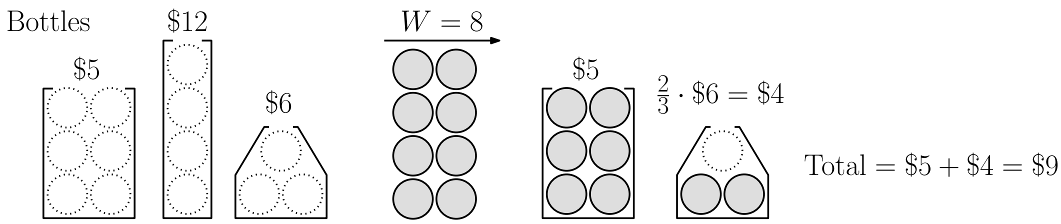

Figure 1: Filling bottles.

(a) (5 points) As a function of W, p_i, and c_i, what is the *greedy order* in which bottles should be filled for the optimal solution?

(b) (10 points) Provide a brief justification of optimality. (To save time, we don't need a complete proof. It suffices to show that the *first* bottle selected by your greedy algorithm must be in some optimal solution, even if you only put *a single pill* in this bottle.)

Problem 3. (10 points) Consider the weighted interval scheduling instance shown in Fig. 2. Show the result of executing the algorithm given in class for this problem. You need only show the *final contents* of the $W[0..n]$ array.

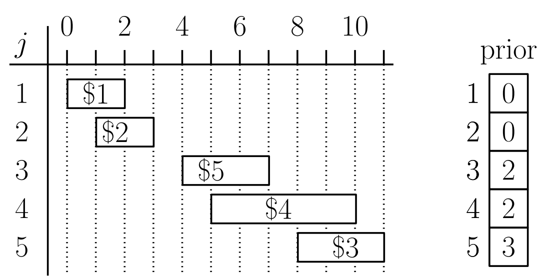

Figure 2: Weighted interval scheduling example.

Problem 4. (10 points) In this problem we consider a variant of the weighted interval scheduling (WIS) problem, where we *penalize* the solution whenever there are a gaps between scheduled intervals. As before, we are given a set of n intervals, where interval i is $[s_i, f_i]$ and yields value v_i. The day starts with the earliest start time and ends with the latest finish time. Whenever we leave a gap of length t in our solution, we subtract a penalty of t from our earnings.

For example, in Fig. 3, we show two solutions. One has value of $\$7 + \$8 = \$15$, but is incurs a gap penalty of $-\$(2 + 2 + 4) = \$8$, for a net of \$7. The other has a value of $\$6 + \$5$, but has a penalty of only $-\$3$, for a total of \$8, which is better.

Give a recursive DP formulation for solving this problem. (We just want the recursive formula, not a full algorithm). Briefly explain your answer.

Hint: Let $W[i]$ be the optimal value (including penalties) for the first i intervals. You may assume that requests are already sorted by finish times, and you are given the prior(j) values. If you are completely stuck, you can just repeat the standard WIS solution for 30% credit.

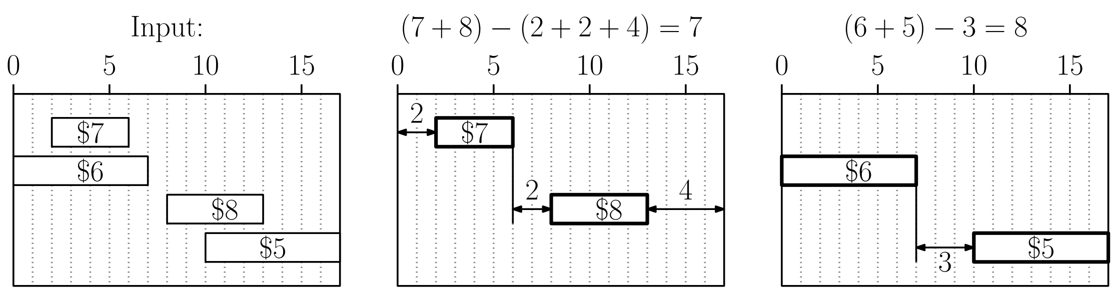

Figure 3: Weighted interval scheduling with penalties.