CS 577: Introduction to Algorithms — Homework 4

Out: 02/17/16 — **Due: 02/24/16**

Ground Rules

- See HW1.
- Both problems will be graded.

Problems

1. A primary task for a typesetter is to break up text into lines so that each line is of roughly similar length. Suppose that the typesetter is given a paragraph with n words, with the ith word containing k_i characters. For simplicity we will assume that there are no spaces or punctuation marks between words. The typesetter must satisfy the following constraints. Each word must fit fully on a line and cannot be broken across two lines. Each line can contain up to L characters, but no more. A line containing L' characters with $L' < L$ faces a penalty equal to $L - L'$. All words are shorter than L characters, so that it is possible to simultaneously satisfy these constraints.

   Design an algorithm for the typesetter that minimizes the sum over all lines of the penalty—L minus the number of characters on that line. Your algorithm should run in polynomial time in n and L. Prove the correctness of your algorithm and analyze its running time.

2. Suppose that on a busy day of the semester you get homework for each of the n courses that you are taking. Homework for the i^{th} course is due within t_i days, and you earn p_i points for submitting the homework on time. Unfortunately it takes you one full day to do each homework. So, for example, if you have 4 homeworks due within 2, 3, 1, and 2 days, respectively, you cannot submit all four on time, but you can submit the first three on time by doing the third one on day 1, the first one on day 2, and the second one on day 3. Your objective is to maximize the total number of points you can earn by submitting a subset of the homeworks on time.

   (a) Let S be a subset of all homeworks. Call S "reasonable" if there is an order in which you can do the homeworks in S so that each one can be submitted on time. Prove that S is reasonable if and only if for all integers $t \leq n$, the number of homeworks in S due within t days or less is no more than t.

   (b) Design a polynomial time algorithm to determine which homework to do on which day so as to maximize the number of points you can earn by submitting a subset of the homework on time. Prove the correctness of your algorithm and analyze its running time.

   Hint: Suppose, first, that for a fixed reasonable set of homeworks, you want to finish all homeworks in this set before their respective deadlines. In what order should you do these homeworks? Next think about a greedy strategy for selecting a reasonable set. Use an exchange argument to prove the correctness of your algorithm.