CS:3330 Homework 10, Spring 2018

Due in class on Thu, May 3

---

1. Problem 6.1. You can solve the problem using an approach very similar to what we used in class to solve the *Longest Increasing Subsequence* problem. Define $OPT(j)$ as the maximum sum of a contiguous subsequence in input $a_1, a_2, \ldots, a_j$. Define $OPT^+(j)$ as the maximum sum of a contiguous subsequence that ends with a_j in input $a_1, a_2, \ldots, a_j$. Write your answer in 3 parts.

   (a) First, write the recurrence relations that express $OPT(j)$ and $OPT^+(j)$ in terms of $OPT(i)$ and $OPT^+(i)$ for values of i smaller than j.

   (b) Then, fill out two arrays $OPT[0..n]$ and $OPTPlus[0..n]$ iteratively using the recurrences from (a).

   (c) Finally, use these arrays to compute a contiguous sequence with maximum sum. The pseudocode for this can be recursive or iterative.

2. Problem 6.2. Here are two hints. (a) The "Obvious Observation" you can use to get started is this: on the last day of the road trip you traveled from mile post a_i for some i, $1 \leq i < n$, to a_n. (b) Define $OPT(j)$ for $1 \leq j \leq n$ as the minimum total penalty you incur in traveling from mile post 0 to the hotel at mile post a_j. Express your answer in three parts, as in Problem 1.

3. Problem 6.4. Here are two hints. (a) The "Obvious Observation" you can use to get started is this: either the last valid word is $s[j..n]$ for some j, $1 \leq j \leq n$ or there is no valid word at the end of $s[1..n]$. (b) Define $OPT(j)$ for $1 \leq j \leq n$ to be True if the string $s[1..j]$ can be decomposed into a sequence of valid words; otherwise $OPT(j)$ is False.

4. Problem 6.7.
