Spring 2019 CS372 Assignment #8.

Due: at the beginning of the lecture on Thursday, May 2.
You may do this assignment in groups of 2 or individually.

1. Recall the longest common subsequence (LCS) problem discussed in class. Recall that c[i,j] is the length of LCS of X_i and Y_j, where X_i is the prefix of X of length i and Y_j is the prefix of Y of length j. LCS recursive solution is given by the following

$$c[i\ ,j] = \begin{cases} c[i-1, j-1]+1, & if\ x[i]=y[j] \\ \max(c[i, j-1], c[i-1, j]), & if\ x[i] \neq y[j] \\ 0, & if\ i=0\ or\ j=0 \end{cases}$$

Find the length of LCS of “AABCBABA” and “BCBCAAB”, as well as an actual longest common subsequence. Show your work - draw a table and fill it in.

2. Use dynamic programming algorithm to find edit distance between strings "STUDIES" and "SUCCESS". Show your work - draw a table and fill it in. (The algorithm is described in Section 6.3)

3. Exercise 6.1. Hint: Subproblems are D(i) ($0 \leq i \leq n$) where D(i) is the largest sum of a (possibly empty) contiguous subsequence ending exactly at position i. You need to write a recursion which can be used to compute D(i), explain your algorithm and show that its running time is linear.

4. Exercise 6.7. Hint: Subproblems are L(i,j) ($1 \leq i \leq j \leq n$) where L(i,j) is the length of the longest palindromic subsequence of string x[i,…,j]. You need to write a recursion which can be used to solve the subproblems, explain your algorithm and show that its running time is $O(n^2)$.

5. Let X, Y, Z be 3 strings. Design a dynamic programming algorithm that would find the length of the longest common subsequence of X, Y, and Z.
(a) Define a suitable subproblem.
(b) Give a recursive solution to the subproblem.
(c) Give a pseudocode for a dynamic programming algorithm that solves the problem.
(d) Analyze the running time of your algorithm.