

**CMPT307**  
**Summer 2020**  
**Assignment 2**  
**Due Wed June 24 at 23:59**  
**3 problems, 40 points.**

1. Improve the Longest Common Subsequence (LCS) algorithm (10 points):

- (a) Show how to compute the length of an LCS using only  $2 \min(m, n)$  entries in the  $c$  table plus  $O(1)$  additional space. Express in pseudocode. Then analyze the memory space usage of your algorithm.
- (b) Then show how to do the same thing, but using  $\min(m, n)$  entries plus  $O(1)$  additional space. Again, express in pseudocode, and analyze the memory space usage of your algorithm.

2. Refer to the power of 2 problem (Lecture 12, slides p21) (10 points).

- (a) Redo the problem using the accounting method.
- (b) Redo the problem using the potential method.

3. Coin changing (20 points):

Consider the problem of making change for  $n$  cents using the fewest number of coins. Assume that each coin's value is an integer.

- (a) Describe a greedy algorithm to make change consisting of quarters, dimes, nickels, and pennies. Prove that your algorithm yields an optimal solution.
- (b) Suppose that the available coins are in the denominations that are powers of  $c$ , i.e., the denominations are  $c^0, c^1, \dots, c^k$  for some integers  $c > 1$  and  $k \geq 1$ . Show that the greedy algorithm always yields an optimal solution.
- (c) Give a set of coin denominations for which the greedy algorithm does not yield an optimal solution. Your set should include a penny so that there is a solution for every value of  $n$ .
- (d) Give an  $O(nk)$ -time algorithm that makes change for any set of  $k$  different coin denominations, assuming that one of the coins is a penny.