

CS610 (2017) Homework 2, Due 12th November

October 31, 2017

Write down the names of your collaborators. Please follow the honor code as outlined on the class webpage. Please submit a single pdf that contains typed up/scanned answers. Using Latex is strongly advised. For each problem, besides writing down the algorithm, please write down 1) a clear analysis of complexity 2) a clear claim of correctness and 3) a proof of the claim.

1. KT chapter 6, problem 1.
2. KT chapter 6, problem 2.
3. KT chapter 6, problem 3.
4. KT chapter 6, problem 5.
5. Let us define a multiplication operation on three symbols a, b, c according to the following table; thus $ab = b$, $ba = c$ and so on. Note that the multiplication operation defined below is neither associative nor commutative.

	a	b	c
a	b	b	a
b	c	b	a
c	a	c	c

Find an efficient algorithm that examines a string of these symbols, say $bbbac$ and decides whether or not it is possible to parenthesize in a way such that the value of the resulting expression equals a . For example, on input $bbbac$ your algorithm should return *yes* since $((b(bb))(ba))c = a$. But it should return no on input bbb .

6. Given an unlimited supply of coins of denominations x_1, x_2, \dots, x_n , we wish to make change for a value v ; that is, we wish to find a set of coins whose total value is v . This might not be possible. For instance, if the denominations are 5 and 10, we can make change for 15 but not for 12. Give an $O(nv)$ dynamic programming algorithm for the following problem:
Input: $x_1, \dots, x_n; v$.
Output: Is it possible to make change for v using coins of denominations x_1, \dots, x_n ?