CS300 Homework #4

TA in charge: Jaeyoon Kim

jaeyoon1603@gmail.com

1. (20 points) Find a dynamic programming algorithm with a time complexity of O(nk) that computes the binomial coefficient

$${}_nc_k = \binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

$$\binom{n}{0} = \binom{n}{n} = 1$$

when n and k are given. Suppose that n and k are an integer, $0 \le k \le n$, and addition operations take one unit of time.

2. (20 points) You can create a string consisting of A or B. However, your string can not contain consecutive A's. For example, strings of length 2 that you can create are AB, BA, BB. You wish to find an algorithm using dynamic programming to find the number of possible strings of length N. State the set of subproblems that you will use to solve this problem and the corresponding recurrence relation to compute the solution.

3. (25points) Give a polynomial time algorithm that takes three strings, A, B and C, as input, and returns the longest sequence S that is a subsequence of A, B, and C.

4. (Bonus 20points) Give a algorithm that takes a positive integer n as input, and computes the number of possible orderings of n objects under the relations < and =. For example, if n = 3 the 13 possible orderings are as follows: a = b = c, a = b < c, a < b = c, a < b < c, a < c < b, a = c < b, b < a = c, b < a < c, b < c < a, b = c < a, c < a = b, c < a < b, and c < b < a. Your algorithm should run in time polynomial in n.

5. (35points) Giving change You want to program the algorithm for giving the change of a vending machine so that it gives the least number of coins. The machine contains several types of coins of value $\{v_1, \ ..., \ v_n\}$. The amount to be given back is denoted by S. We suppose that the available coins allow to reach exactly the value S. First, we consider that the coins are in unlimited amount.

(a)(20points) Indicate how to solve the problem in general using dynamic programming. For that, you must describe the optimal sub-problems to be calculated, the basic cases and the recurrence relation between the optimal solutions of the sub-problems.

(b)(15points) Find the optimal solution with coins of value {1, 4, 5} and S = 12 using your dynamic programming.