*CSE 551 Foundations of Algorithms
Mid Term Make-up Test, Fall 2017
Closed Books, Closed Notes
Time: 1 hour
Answer any three questions
Each question carries 25 pts.

Problem 1: Suppose that you are choosing between the following three algorithms:

(i) Algorithm A solves the problem by dividing it into five subproblems of half the size, recursively solving each subproblem, and then combining the solutions in linear time.

(ii) Algorithm B solves the problem of size n by recurisively solving two subproblems of size $n-1$ and then combining the solutions in constant time.

(iii) Algorithm C solves the problem of size n by dividing them into nine subproblems of size $n/3$, recurisively solving each subproblem, and then combining the solutions in $O(n^2)$ time.

What are the running times of each of the algorithms (in big-O notation), and which would you choose?

Problem 2: Suppose that the dimensions of the matrices A, B, C and D are $20\times 2, 2\times 15, 15\times 40$ and 40×4, respectively. Find the fewest number of multiplications that will be necessary to compute the final matrix and the order in which the matrix chain be multiplied so that it will require the fewest number of multiplications. **Show all your work**.

Problem 3: If k is a nonnegative constant, then the solution to the recurrence

$$T(n) = \begin{cases} k, & n = 1 \\ 3T(n/2) + kn, & n > 1 \end{cases}$$

for n a power of 2 is $T(n) = 3kn^{log_2 3} - 2kn$

Prove this statement.

Problem 4: While discussing the Longest Common Subsequence (LCS) problem, we noted that there may be more than one LCS between two given sequences X and Y. The algorithm discussed in class produces only one (possibly among many) LCS. Develop an algorithm to find *all* Longest Common Subsequences of two given sequences X and Y.