CS6505 Computability and Algorithms

Homework 7. Due in class on Fri, Mar 12.

1. Assume you are solving a problem with input size n using divide-and-conquer. By performing n^2 operations, you can create 10 subproblems of size $n/3$, and with n^2 more operations, you can combine the answers to those subproblems into a solution to your original problem of size n.

   (a) Give a tight bound on the time complexity of the natural recursive algorithm.

   (b) Suppose you can reduce the number of subproblems to 9 with only n^2 more operations in the divide and merge steps. How does this improve your overall complexity? (Give the new bound).

   (c) Now suppose you can further reduce the number of subproblems to 8 with only n^2 more operations in the divide and merge steps. Give a new bound on the complexity.

2. When multiplying two 3×3 matrices with integer entries, the näive approach requires 27 multiplications. Can you multiply them with fewer multiplications? Full credit if you can prove it can be done with only 26. If you can prove it for $x < 26$ multiplications, you get $5 \cdot (26 - x)$ bonus points.

3. Suppose we have two $n \times n$ matrices, with $n = 2^k,\ k \in \mathbb{Z}$, that have the following recursive structure: When divided into four equal-size blocks, the two diagonal blocks are identical, and the two off-diagonal blocks are identical, i.e.,

   $$A = \begin{pmatrix} A_1 & A_2 \\ A_2 & A_1 \end{pmatrix}$$

   Each block also has this structure recursively all way down to the level of single entries, which are just integers. Give a method of multiplying two such matrices that takes advantage of this structure. Full credit for any algorithm that has complexity $O(n^{2.5})$ or less.