CS41 Homework 7

This homework is due at 11:59PM on Sunday, October 28. This is a **6 point** homework.
Write your solution using LaTeX. Submit this homework using **github** as a **.tex** file. This is a **partnered homework**. You should primarily be discussing problems with your homework partner.

It's ok to discuss approaches at a high level with others. However, you should not reveal specific details of a solution, nor should you show your written solution to anyone else. The only exception to this rule is work you've done with a lab partner *while in lab*. In this case, note (in your **README** file) who you've worked with and what parts were solved during lab.

Note: There are two recurrence relations on this homework. Solve one of them using recursion trees. Solve the other using partial substitution.

1. Solve the following recurrence relation.

$$T(n) = 4T(n/2) + 3n^2,$$
$$T(1) = 1$$

2. **Integer Multiplication.** Recall the Karatsuba algorithm for integer multiplication for class, which multiplies two n-digit base-N numbers a, b by:

$$\begin{aligned} a \cdot b &= (a_L N^{n/2} + a_r)(b_L N^{n/2} + b_R) \\ &= a_L b_L N^n + (a_L b_R + a_R b_L) N^{n/2} + a_R b_R \\ &= A N^n + (C - A - B) N^{n/2} + B, \end{aligned}$$

where the three $(n/2)$-digit multiplications are:

- $A = a_L \cdot b_L$
- $B = a_R \cdot b_R$
- $C = (a_L + a_R)(b_L + b_R)$

Consider an algorithm for integer multiplication of two n-digit base-N numbers where each number is split into three parts, each with $n/3$ digits.

(a) Design such an algorithm, similar to the integer multiplication we did in class. Your algorithm should describe how to multiply two integers using only six multiplications (instead of the straightforward nine).

(b) Determine an asymptotic upper bound for the running time of your algorithm. (Write it as a recurrence, and then solve the recurrence.)

(c) Is this algorithm asymptotically faster than Karatsuba multiplication? That is, is it better to use the algorithm that breaks an integer into three parts, or two parts?

3. **(extra challenge)** Solve the following recurrence relation:

$$T(n) = \sum_{i=1}^{n-1} (T(i) + T(n-i)), \text{ for all } n > 1$$

$$T(1) = c$$

…where c is some constant. Show your work.