CSC373 Worksheet 1

June 6, 2020

Source: link

1. **CLRS 4.2-2:** Write Pseudocode for Strassen's algorithm

2. **CLRS 4.2-4:** What is the largest k such that if you can multiply 3×3 matrics using k multiplications (not assuming commutativity of multiplciation), then you can multiply $n \times n$ matrices in time $o(n^{\lg 7})$? What would the runing time of this algorithm be?

3. **CLRS 4.2-5:** V.Pan has discovered a way of multiplying 68×68 matrices, using 132,464 multiplications, a way of multiplying $70 \times 70 matrices$ using $143, 640$ multiplications, and a way of multiplying 72×72 matricsusing $155, 424$ multiplicaiotns. Which method yields the best asymptotic running time when used in a divide-and-conquer matric-multiplication algorithm? How does it compare to Strassen's algorithm?

4. **CLRS 4.2-7:** Show how to multiply the complex numbers $a + bi$ and $c + di$ using only three multiplications of real numbers. The algorithm should take a, b, c and d as input and product the real component $ac - bd$ and the imaginary component $ad + bc$ separately.

5. **CLRS 4-1:** Give asymptotic upper and lower bounds for $T(n)$ in each of the following recurrences. Assume that $T(n)$ is constant for $n \leq 2$. Make your bounds as tight as possible, and justify your answers

   a) $T(n) = 2T(n/2) + n^4$
   b) $T(n) = T(7n/10) + n$
   c) $T(n) = 16T(n/4) + n^2$
   d) $T(n) = 7T(n/3) + n^2$
   e) $T(n) = 7T(n/2) + n^2$
   f) $T(n) = 2T(n/4) + \sqrt{n}$
   g) $T(n) = 2T(n - 2) + n^2$

6. **CLRS 33.4-2:** Show that it actually suffices to check only the points in the 5 array positions following each point in the array Y'

7. **CLRS 33.4-4:** Give two points p_1 and p_2 in the plane, the L_∞-distance between them is given by $max(|x_1 - x_2|, |y_1 - y_2|)$. Modify the closest-pair algorithm to use the L_∞-distance.

8. **CLRS 33.4-6:** Auggest a change to the closest-pair algorithm that avoids presorting the Y array but leaves the running time as $O(n \lg n)$. (*Hint:* Merge sorted arrays Y_L and Y_R to form the sorted array Y.