CS6033 Homework Assignment 6*

Due Oct 29th at 5:00 pm
No late assignments accepted

1. Solve these recurrence formulas using Θ notation:
   - $T(n) = 2T(n/3) + 1$
   - $T(n) = 5T(n/4) + n$
   - $T(n) = 7T(n/7) + n$
   - $T(n) = 9T(n/3) + n^2$
   - $T(n) = 8T(n/2) + n^3$
   - $T(n) = 7T(n/2) + \Theta(n^2)$
   - $T(n) = T(n/2) + \Theta(1)$
   - $T(n) = 5T(n/4) + \Theta(n^2)$

2. 7.4.1 from CLRS on page 184.

3. Question 7-1a on page 185 in the textbook.1

4. If the array contained many duplicate items, which would be a better partitioning algorithm: Hoare, or the one presented in class?

5. "A very common problem in computer graphics is to approximate a complex shape with a bounding box. For a set, S, of n points in 2-dimensional space, the idea is to find the smallest rectangle, R, with sides parallel to the coordinate axes that contains all the points in S. Once S is approximated by such a bounding box, we can often speed up lots of computations that involve S. For example, if R is completely obscured some object in the foreground, then we don't need to render any of S. Likewise, if we shoot a virtual ray and it completely misses R, then it is guaranteed to completely miss S. So doing comparisons with R instead of S can often save time. But this savings is wasted if we spend a lot of time constructing R; hence, it would be ideal to have a fast way of computing a bounding box, R, for a set, S, of n points in the plane. Note that the construction of R can be reduced to two instances of the problem of simultaneously finding the minimum and the maximum in a set of n numbers; namely, we need only do this for the x-coordinates in S and then for the y-coordinates in S. Therefore, design a *divide-and-conquer* algorithm for finding both the minimum and the maximum element of n numbers using no more than $3n/2$ comparisons."

*Many of these questions came from outside sources.

6. Professor Williams (having looked at the proof on page 1041) comes up with a scheme that allows the closest-pair algorithm to check only 5 points following each point in array Y'. The idea is always to place points on line l into set P_L. Then, there cannot be pairs of coincident points on line l with one point in P_L and one in P_R. Thus, at most 6 points can reside in the $\delta \times 2\delta$ rectangle. What is the flaw in the professor's scheme?

7. You are a visitor at a political convention with n delegates; each delegate is a member of exactly one political party. It is impossible to tell which political party any delegate belongs to; in particular, you will be summarily ejected from the convention if you ask. However, you can determine whether any pair of delegates belong to the same party by introducing them to each other. Members of the same political party always greet each other with smiles and friendly handshakes; members of different parties always greet each other with angry stares and insults.

   Suppose more than half of the delegates belong to the same political party. Describe a divide and conquer algorithm that identifies all members of this majority party[1]

   This is a slightly modified question from https://courses.engr.illinois.edu/cs374/sp2018/A/notes/01-recursion.pdf

8. Say you have an array for which the ith element is the predicted price of a given stock on day i. If you were only permitted to complete at most one transaction (i.e, buy one and sell one share of the stock), design an algorithm to find the maximum predicted profit. (Use divide and conquer to solve this question. hint: consider how we do merge sort in the text book page 31.)

9. Design an algorithm to find the closest pair of points in $O(n \log n)$ worst case running time. Justify your running time. You may use any algorithm presented in class.

10. (3 bonus points) Think of a good exam/homework question for the material covered in Lecture 6.

[1] For a small amount of extra credit - design a faster algorithm