CS 4349.501 Homework 2

Due Wednesday September 13th, in class

September 6, 2017

Please answer each of the following questions. Each student must write their solutions in their own words and submit their solutions on paper at the beginning of class. ***Include your name and/or Net ID at the top of each page.***

1. Using Θ-notation, provide asymptotically tight bounds in terms of n for the solution to each of the following recurrences. Assume each recurrence has a non-trivial base case of $T(n) = \Theta(1)$ for all $n \leq n_0$ where n_0 is a suitably large constant. For example, if asked to solve $T(n) = 2T(n/2) + n$, then your answer should be $\Theta(n \log n)$. **Give a brief explanation for each solution.** Giving only the upper or lower bound (using big-oh or big-Ω notation) may be worth partial credit.

   (a) $T(n) = 8T(n/2) + n^3$

   (b) $T(n) = 5T(n/3) + n$

   (c) $T(n) = T(n-1) + n^2$

   (d) $T(n) = T(n/2) + 2T(n/5) + n$

   (e) $T(n) = T(\sqrt{n}) + T(\sqrt[3]{n}) + \lg n$

2. Suppose you are given a matrix (a 2-dimensional array) $A[1..m][1..n]$ of numbers. An element $A[i][j]$ is called *good* if each of its neighbors $A[i-1][j]$, $A[i+1][j]$, $A[i][j-1]$, and $A[i][j+1]$ are at most $A[i][j]$. For simplicity, assume elements beyond the boundaries of A are equal to $-\infty$. In other words, $A[0][\cdot] = A[n+1][\cdot] = A[\cdot][0] = A[\cdot][n+1] = -\infty$.

   (a) Suppose $m = 1$ so we only have the array $A[1][1..n]$. Design and analyze an algorithm to find a good element of A in $O(\log n)$ time.

   (b) Now suppose $m = n$. Design and analyze an algorithm to find a good element of A in $O(n \log n)$ time. **Hint:** Suppose you know the maximum element of a column $A[\cdot][j]$, but it is not itself good. What can you say about the maximum elements of neighboring columns?

   (c) **Extra credit:** Design and analyze an algorithm to find a good element of A in $O(n)$ time.