Homework 5

All ungraded

1. For each part below, solve the following recurrences by giving tight Θ-notation bounds in terms of n for sufficiently large n, and briefly describe the steps. Assume that T(n) is a positive and non-decreasing function of n and represents the running time of an algorithm. In some cases, we shall need to invoke the Master Theorem with a generalization of case 2:

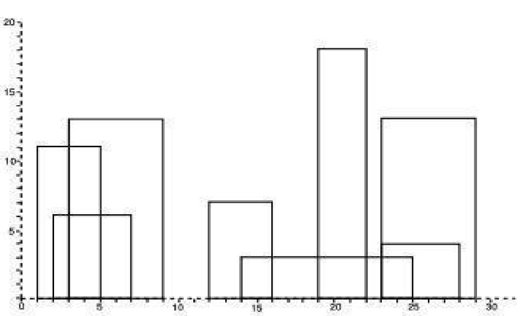
   If the recurrence $T(n) = aT(n/b) + f(n)$ is satisfied with $f(n) = \Theta(n^{\log_b a} \log^k n)$ for some k≥0, then $T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$.

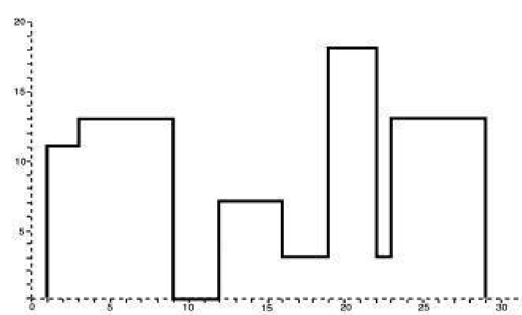
   a. $T(n) = 49T(n/7) + n^2 \log n$
   b. $T(n) = 4.001\, T(n/2) + n^2 \log^4 n$
   c. $T(n) = 100T(n/2) + n^{50}$
   d. $T(n) = 10T(n/2) + 2^n$
   e. $T(n) = 2T(\sqrt{n}) + \log n$

2. Assume that you have a blackbox that can multiply two integers. Describe an algorithm that when given an ***n***-bit positive integer a and an integer x, computes x^a with at most $O(n)$ calls to the blackbox. (a 'blackbox' simply means a 'function' whose implementation specifics are *unknown*)

3. Solve Kleinberg and Tardos, Chapter 5, Exercise 3.

4. A city's skyline is the outer contour of the silhouette formed by all the buildings in that city when viewed from a distance. A building B_i is represented as a triplet (L_i, H_i, R_i) where L_i and R_i denote the left and right x coordinates of the building, and H_i denotes the height of the building. Describe an O(n log n) algorithm for finding the skyline of n buildings.
   For example, the skyline of the buildings {(3, 13, 9),(1, 11, 5),(12, 7, 16),(14, 3, 25),(19, 18, 22),(2, 6, 7),(23, 13, 29),(23, 4, 28)} is {(1, 11),(3, 13), (9, 0), (12, 7), (16, 3), (19, 18), (22, 3), (23, 13), (29, 0)}. (Note that the x coordinates in a skyline are sorted)


buildings


skyline

5. Solve Kleinberg and Tardos, Chapter 5, Exercise 5.