CMSC 351: Practice Problems

Samir Khuller and Brian Postow

Due in class: Oct 29.

(1) Solve the following recurrence using the master theorem. (Express your answer using the Θ notation.)

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

Assume that n is a power of 3.

Also obtain the exact solution by the iteration method. Now that you have the exact solution, prove the bound by induction.

(2) Prove that $\sum_{i=1}^{n} i^k \in \Theta(n^{k+1})$.

(3) Prove or disprove:

(a) $2^{n+1} \in O(2^n)$.

(b) $2^{2n} \in O(2^n)$.

(4) Show that $\log(n!) \in \Theta(n \log n)$.

(5) Given the exact locations and shapes of several rectangular buildings in a city, draw the skyline of these buildings after eliminating hidden lines. You may assume that all the buildings are resting on a straight line. Building i is represented by a triple (L_i, H_i, R_i) where L_i and R_i are the left and right x coordinates of the building and H_i denotes the height of the building. A skyline is a list of coordinates and the heights connecting them are arranged in order from left to right. Heights are indicated in **boldface**. $(1, \mathbf{11}, 5), (2, \mathbf{6}, 7)$ and $(3, \mathbf{13}, 9)$ could denote a possible input. The output is $(1, \mathbf{11}, 3, \mathbf{13}, 9)$.

(6) The "maximum independent subset" of an array is a subset of the values whose sum is as large as possible, but no two elements of the subset can be contiguous in the array.

(a) Give a linear time algorithm to find the sum of the maximum independent subset. Hint: Keep track for each element the maximum independent sum of elements before and including that element and *also* keep track for each element the maximum independent sum of elements before and *not* including that element.

(b) Give a linear time algorithm that not only finds the sum of the maximum independent subset, but also finds (the index of) the elements in the the maximum independent subset.