CSE 417

Algorithms & Computational Complexity

Assignment #6 (rev. a)

Due: Friday, 2/18/22

1. Solve the following recurrence using the "Master Recurrence" (approximately slide 56 in the D&C slides). Clearly specify the constants a, b, c, d, k. You may assume that n is a power of 3:

$$T(n) = \begin{cases} 0, & \text{if } n = 1 \\ 3T(n/3) + cn, & \text{otherwise.} \end{cases} \tag{1}$$

2. Solve the recurrence given in the previous problem using the recursion tree/tabular method shown in lecture (approximately slides 49-55). You may assume that n is a power of 3. Try to give an exact solution, but a good big-O estimate is adequate if you can't.

3. Let A and B be two arrays of length n containing numeric values, each array in sorted order. Give an $O(\log n)$ time divide-and-conquer algorithm to find the *median* of the set of $2n$ items in $A \cup B$, where the median is defined to be the n-th smallest item in that union. For simplicity, you may assume that all values (in each array and between arrays) are distinct from each other, and you may assume that $n = 2^k + 1$ for some integer $k \geq 0$.

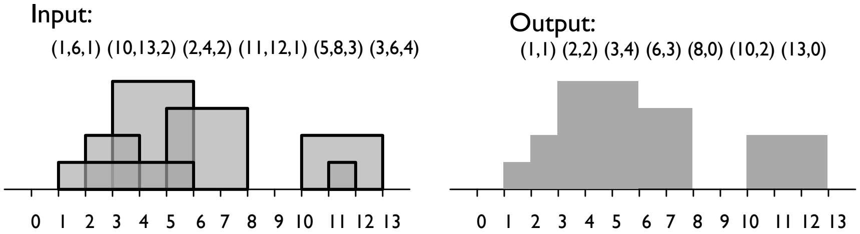
   Give a brief English language description of your (recursive!) algorithm. Also write a clear pseudocode version of it. As outlined in the "Code to Recurrence" section of the D&C slides: Select the operations you want to count, and very carefully state exactly what they are. Annotate your pseudocode to highlighting the base case(s), recursive calls, and counted operations. Then write the corresponding recurrence relation. Use the "master recurrence" to solve it (approximately slide 56 in the D&C slides); clearly indicate what the constants a, b, c, d and k given in the master recurrence are for this problem. The emphasis this time is on analysis, but do include a brief correctness argument.

4. Give a fast divide and conquer algorithm for the following *silhouette problem*:

   Input: an unordered list of triples (x_l, x_r, h), where $0 < x_l < x_r$ and $0 < h$.

   Problem: Each triple describes a rectangle of height h whose base is on the x axis, and whose left and right sides are at positions x_l and x_r, respectively. Imagine that each rectangle is opaque, and that the whole collection is illuminated from behind. The problem is to calculate the silhouette of the set of rectangles — a description of the visible edges in the resulting scene. (This is a very simple case of what's called *hidden line elimination* in computer graphics.)

   Output: A list of pairs (x_i, y_i), ordered by increasing x_i, of the height y_i at each point x_i where the height of the silhouette changes. (See example below.)



   Unlike the example above, you may assume that all x coordinates are distinct and that n is a power of two. Your answer should address all points raised in the last paragraph of problem 3.

5. Multiply the 4-digit decimal numbers 2142 and 1233 using a Karatsuba-like divide-and-conquer algorithm similar to the one given in class for mutiplying binary numbers (i.e., with 3 multiplies of half-size integers per recursive level; base case is "table lookup" of 1-digit by 1-digit decimal multiplication). Show all steps. The usual "grade school" multiplication method requires $4 \times 4 = 16$ "base case" multiplies. How many did you need? The unnumbered slide following D&C slide 45 gives an example. If you choose to follow this model, don't try to show all of the red/green arrows—too much clutter—but showing a few may make the pattern clearer.

Revision History:

Rev a: Added pointer to Karatsuba example slide. — 2/13/19.