CS140: Algorithms

Fall 2000

Assignment 2

This assignment is due by 5PM on Friday, Sept. 8. You may collaborate with your classmates to complete this assignment but you must reference your collaborators. In addition, you must do your own write-up. This assignment is worth 100 points (85 points if handwritten).

1. Prove $\sum_{i=0}^{n} i2^i = (n-1)2^{n+1} + 2$ by induction on n.
2. Compute the running times of the following program.

   algorithm what(n)

   ```
   for i = 1 to ⌊√n⌋ do
      for j = ⌊√n⌋ downto 1 do
         for k = 1 to j do
            { O(1) time sequence of statements }
         end
      end
   end
   ```

   end algorithm
3. CrazySort is a sorting algorithm that sorts a list of n integers as follows. If the list has 1 element, the algorithm returns the list as sorted. If the list has 2 elements, the algorithm compares the elements and swaps them if necessary. If the list has 3 or more elements, the algorithm recursively sorts the first 2/3rds of the input list, then recursively sorts the second 2/3rds, and then recursively sorts the first 2/3rds again. (You might want to convince yourself that the algorithm is correct.) Use the work tree approach to analyze the running time of CrazySort on inputs of size 3^m, $m = 0, 1, \ldots$. In particular, please answer the following questions.
   (a) How many nodes are at level i, where the root is at level 0?
   (b) What is the instance size associated with a node at level i?
   (c) How much work is associated with a node at level i?
   (d) How much work is done at the nodes in level i?
   (e) How many levels are in the work tree?
   (f) What is the running time of CrazySort?

   Now prove that the running time you found holds for arbitrary size input.