CS 361, HW7

Prof. Jared Saia, University of New Mexico

Ungraded

1. Consider the recurrence $T(n) = 3T(n/3) + n$
   (a) Use the recurrence tree method to get a tight upper bound (i.e. big-O) on the solution to this recurrence
   (b) Now use annihilators (and change of variables) to get a tight upperbound on the solution to this recurrence.
   (c) Now solve using the Master Theorem (all three bounds should match)
2. Consider the recurrence $T(n) = 3T(n-1) - 2T(n-2) + 4$, $T(0) = 0$, $T(1) = 0$. Solve this recurrence *exactly* using annihilators. Don't forget to check your answer.
3. (From Cris Moore's old midterm) Consider the following recursive sorting algorithm:

   Wackysort:

   (a) Bubblesort the first half of the list
   (b) Wackysort the second half of the list
   (c) Merge the two sorted halfs together

   (a) Write down and solve in terms of tight big-O, the recurrence relation for the run time of Wackysort on a list of size n.
   (b) Assuming that the merge subroutine and Bubblesort are correct, prove by induction that Wackysort is correct.
4. Search Tree and Skip List questions (some taken from old finals by Cris)

(a) Recall that a node in a skip list at level i has i copies (one at each level less than or equal to i). Imagine the following new scheme for choosing the level of a node in a skip list. Let X be the random variable giving the level chosen for a new node. Then $P(X \geq i) = (1/3)^{i-1}$ for i from 1 to infinity. What is the expected height of each node? (Recall that $E(X) = \sum_i P(X \geq i)$). Now what is the expected *total* height of all n nodes in such a skip list? (Note: Think about how you would solve this problem if there was some cutoff for i e.g. $P(X \geq i) = 0$ for $i \geq k$)

(b) Suppose we have a balanced trinary tree where every internal node has 3 children instead of 2 and where all the leaves are the same distance from the root. When the number of nodes n is large, what is the fraction of nodes that are leaf nodes?

(c) Recall that an AVL tree is one where every node has a height imbalance on its children of +1,0, or −1. How many leaf nodes does the AVL tree of depth 7 with the smallest number of nodes have? Try to avoid actually drawing the tree.

5. The following algorithm takes an input array A and assigns each array location the value of the maximum element in the original array.

```
ChangeToMax(A,n){
  for (i=0;i<n;i++){
    if(A[i+1]<A[i]){
      A[i+1] = A[i];
    }
    else{
      //change each elem in A[0..i] to A[i+1]
      for(j=0;j<=i;j++){
        A[j] = A[i+1];
      }
    }
  }
}
```

- State a loop invariant for the outer for loop. You can assume that the inner loop does what the comment says it does
- Establish initialization, maintenance and termination for your loop invariant.