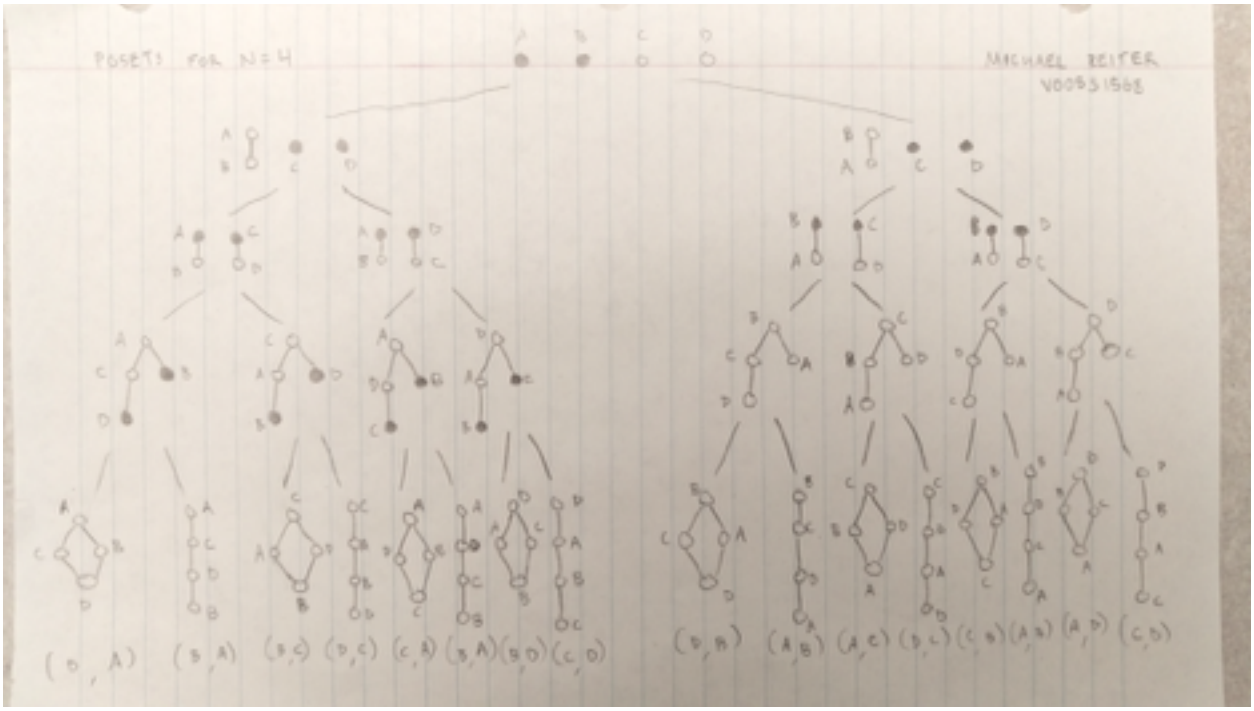
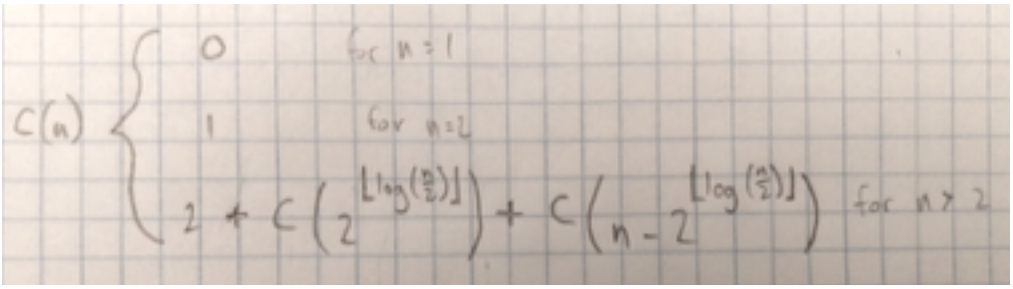
Michael Reiter
V00831568

CSC 226 Assignment 2 Written



The smallest value of n for which mmA and mmB use a different number of comparisons is 6.



The minimum number of inversions of a permutation of 1, 2, … n is 0 (provided its already in order).
The maximum number of inversions of a permutation of 1, 2, … n is n choose 2 = n(n-1) / 2.

To compute inversions in a permutation, you can create a Red Black tree and keep track of a sum of inversions. Each item in the permutation is added to the tree. Whenever a node is attached to the left subtree, after doing any necessary rotations, we add the amount of nodes to the right of it to the sum (i.e. size of right subtree+1). After the tree is complete, the sum will equal the number of inversions. Since we are adding n items to the tree, we are calling put() n times. put() runs in O(logn) time since the Red Black tree is a balanced binary tree. The modification to put to keep track of the inversions incurs no extra run time asymptotically, so the total run time is O(nlogn). The code for the modification to put would look something like this:

```
if (cmp < 0) { h.left = put(h.left,  key, val); inversions +=
size(h.right)+1; }
```