Cell phones, pagers, etc. should all be off! If such a device goes off, you might be officially finished with your test. No tools (calculators, computers, sliderules, screwdrivers, friends, books, or notes) should be used for the test, other than a pen/pencil, your wits and knowledge. You should try to answer all questions for the rote exams. Good luck.

You may use the Master Theorem. It follows, in a somewhat over-simplified format. (Restrictions for case 3 have been dropped, they won’t be needed for the first problem of this test.)

For $T[n] = aT[n/b] + f(n)$, then

1. If $f(n) = O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, then $T[n] = \Theta(n^{\log_b a})$.

2. If $f(n) = \Theta(n^{\log_b a})$, then $T[n] = \Theta(f(n) \lg n)$.

3. If $f(n) = \Omega(n^{\log_b a + \epsilon})$ for some constant $\epsilon > 0$, then $T[n] = \Theta(f(n))$.

1. (10 points) Prove $\Theta(\cdot)$ notation for the following recurrence relation: SOME RECURRENCE HERE. THE MASTER THEOREM WILL APPLY.

2. (10 points) For SOME RECURRENCE RELATION HERE, prove SOME RELATION HERE. THE SUBSTITUTION METHOD WOULD BE A GOOD CHOICE. THE MASTER THEOREM? NOT SO MUCH.

3. (10 points) Consider the following loop. You may assume that $n > 0$. I WILL GIVE YOU A LOOP HERE.

I WILL GIVE YOU A LOOP INVARIANT HERE.

Prove the loop invariant holds (initialization, and maintenance). What does it prove about the loop when the loop exits (termination)?

4. (10 points) Some question dealing with 23 trees, and/or top-down 234-trees, and/or bottom up 234 trees. Know the difference. Know how insertion works for all.

5. (10 points) Given a list of numbers, list all pairwise comparisons made, in order, for some sort(s) chosen from (bottom-up mergesort, top-down mergesort, quick-sort, quick-select), where those last two would have pivots specified.

6. (10 points) SOME GENERIC RUNTIME QUESTION HERE: know the runtimes of all basic algorithms we have seen, and be able to do algebra manipulations so that, given a runtime on one sized set of inputs, you can calculate an expected runtime on a different sized set, perhaps on a different speed machine.

7. (10 points) Some kind of heap question. Know how to insert/delete from a heap, and what heapsort looks like right in the middle of running it. For instance: you are in the middle of running heapsort, and are in the midst of deleting values from the heap. The array of values is given below. Show the array after 2 more values have been deleted. List all comparisons made between elements.

8. (10 points) You are running A LINEAR OR QUADRATIC SORT NAMED HERE, WITH DETAILS OF WHAT HAPPENS DURING ITS RUN. Given those details, what is its expected runtime?