CS 146 (Taylor) **Fall 2023**
Data Structures & Algorithms **Rote Exam Part 1 Template (1 page)**

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.** 60 points total. Good luck.

1. (8 points) For SOME RECURRENCE RELATION HERE, prove SOME RELATION HERE.

   I highly recommend using the substitution method here, in which case you can use the following template. (If you choose to not use the template, you can just write your answer in the given blank space below...but I recommend that you use the template.)

   First, we inductively assume that for $x < n$, _____ holds. Using that inductive hypothesis and our given recurrence relation, we know that _____, and we want to prove that _____.

   The relation we are trying to prove holds for values $C =$_____ and $n_0 =$_____, as is shown by the math below. (Show your math below.) (I will also give this template on the exam.)

2. (8 points) Consider the following loop. You may assume that $n > 0$. I WILL GIVE YOU A LOOP HERE.

   The loop invariant is: 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)?

3. (8 points) You are given the following 234-tree, but do not know if it is a top-down or bottom-up tree. Pick a single number to insert which will help distinguish one possibility from the other. Redraw the tree, both ways, after inserting that value.

4. (4 points) List the comparisons made, in order, for bottom-up mergesort on the following list of numbers. LIST GOES HERE.

5. (4 points) List the comparisons made, in order, for top-down mergesort on the following list of numbers. LIST GOES HERE.

6. (4 points) List the comparisons made, in order, for quick sort on the following list of numbers. Use the last possible value in each list as the pivot. You can (should) use the abstract partition that just slides small values to the left and large values to the right of the pivot, in a stable way. LIST GOES HERE.

7. (8 points) 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. (5 points) I will give you a weighted graph, specified in adjacency matrix format (where blanks mean that there is no edge). In some (specified) format, you will need to describe how Boruvka's Minimum Weight Spanning Tree Algorithm works on that data.

9. (6 points) I will give you some kind of dynamic program similar to the memoized Fibonacci program that we saw. You will have to specify a single order which shows (a) when calls are made to the program, (b) when calls return from the program, and (c) when items get filled into the table. (A single order should show all three things. You will not have to show what values actually get filled in.) The assumption is that evaluation happens left to right: if in the Fibonacci program, you have a recursive call like "return f(n-1) + f(n-2)", that f(n-1) will be called on that line before f(n-2). However, anything that f(n-1) calls will return an answer before f(n) does.

10. (5 points) Consider the following graph:

    GRAPH GOES HERE

    For this graph, I will ask questions about different NP related problems, such as the smallest vertex cover, the largest independent set, the largest clique, and/or the smallest dominating set.