

You have to use the designated spaces for your answers. No extra pages will be provided.

**Problem 1: Benchmarking Randomized Quicksort (10 points)**

In this assignment, you will be implementing and analyzing the performance of the randomized quicksort algorithm you saw in class.

1. Implement the randomized quicksort algorithm in your favorite programming language. Use the pseudocode in the textbook as a guide.
2. Modify your quicksort routine so that in addition to sorting a list, it also returns the number of comparisons made. You can do this by adding only a few lines to your existing code.
3. Put the numbers from 1 to 100 into a list. Then shuffle this list once. Let this shuffled list be called  $L$ .
4. Run randomized quicksort on  $L$  100 times. Make sure you're running quicksort on the same list. Every run of quicksort should receive the exact same list as input.
5. Compile the number of comparisons made in the hundred runs of quicksort. Then take their average. Call this average  $X$ .
6. Let  $\bar{X}$  be the expected number of comparisons made when you run randomized quicksort on  $L$ . What is  $\bar{X}$ ? Do not use approximations. Explicitly compute the sum you saw in class.
7. Compare  $X$  with  $\bar{X}$ . Does the theoretical predictions match with the results of your experiments?
8. Prepare a report containing everything you did in this assignment. In particular, your report should address all of the points above.