You're expected to work on the problems before coming to the lab. Discussion session is not meant to be a one-way lecture. The TA will lead the discussion and correct your solutions if needed. For many problems, we will not release 'official' solutions. If you're better prepared for discussion, you will learn more. The TA is allowed to give some bonus points to students who actively engage in discussion and report them to the instructor. The bonus points earned will be factored in the final grade.

In the following, you can assume that all elements in the input are distinct.

1. (Basic) Using Figure 7.1 as a model, illustrate the operation of Partition on the array $A = \langle 13, 19, 9, 5, 12, 8, 7, 4, 21, 2\rangle$. Do the same thing when we have $A = \langle 13, 19, 9, 5, 12, 8, 7, 4, 21, 11\rangle$.
   Sol.
   Omitted.

2. (Basic) Explain the meaning of in-place sorting.

   Sol. The sorting algorithm uses at most $O(1)$ auxiliary space.

3. (Basic) Is Quick-Sort an in-place sorting algorithm? How about Merge-sort, Heap-sort, Insertion-sort?

   Sol. All are in-place sorting algorithms except (Standard) Merge sort.

4. (Basic) The running time of Randomized-Quicksort that picks the pivot uniformly at random is always $O(n \log n)$ for any input of size n. True of False?
   Sol. False. For some badly chosen pivots, the running time could be as high as $\Omega(n^2)$.

5. (Basic) The *expected* running time of Randomized-Quicksort that picks the pivot uniformly at random is $O(n \log n)$ for any input of size n. True of False?
   Sol. True.

6. (Intermediate) The running time of (Deterministic) Quicksort is $\Theta(n^2)$. True of False?
   Sol. True. For sure, it makes at most $O(n^2)$ comparisons for any input. Consider an already sorted array. The algorithm picks n, $n-1$, $n-2\ldots, 2$ as pivot in this order, where $n-1$, $n-2$, ..., 1 comparisons are made, respectively. So, it still makes $(n-1)+(n-2)+\ldots+1 = \Omega(n^2)$ comparisons. So, the running time is $\Omega(n^2)$. Thus, we can conclude that Quicksort's (worst-case) RT is $\Theta(n^2)$.

7. (Basic) Suppose the input is a random permutation of integers 1 through n. The average-case running time of (Deterministic) Quicksort is $\Theta(n^2)$. True of False?
   Sol. False. It is $\Theta(n \log n)$. Note that the expectation is taken over the random permutations.

8. (Intermediate) Solve $T(n) = T(0.9n) + T(0.1n) + \Theta(n)$. What is your answer? No need to give any explanations. Solve $T(n) = T((2/3)n) + T((1/3)n) + \Theta(n)$. What is your answer?
   Sol. $\Theta(n \log n)$ in both cases.

9. (Basic) In (Determinisitc) Quicksort, suppose the input is a sequence of integers $a_1, a_2, \ldots, a_n$. Let $X_{ij} = 1$ if a_i is compared to a_j in the execution; otherwise $X_{ij} = 0$. Consider another input sequence of integers, $b_1, b_2, \ldots, b_n$, where $b_1 = 2a_1, \ldots, b_n = 2a_n$. Let $Y_{ij} = 1$ if and only if b_i is compared to b_j. Is it true that $X_{ij} = Y_{ij}$ for any $1 \leq i < j \leq n$? **Sol.** true

10. (Intermediate) In Randomized-Quicksort, explain why it is wlog to assume that the elements are $1, 2, \ldots n$ for the analysis of the running time.

11. (Advanced) In Randomized-Quicksort, explain why the probability that the ith smallest element is compared to jth smallest element is $\frac{2}{j-i+1}$, where $i < j$