1 Instructions

1. The midterm is to be turned in by 9:15 am.
2. Each question is worth 4 points.
3. Attempt as many problems as you can. You will be given partial credit, as per the policy discussed in class.

2 Problems

1. Solve the following recurrence using substitution:

\[
T(1) = 0 \\
T(n) = 2 \cdot T\left(\frac{n}{2}\right) + n \cdot \log n, \ n \geq 2
\]

2. Given an array \( A \) of \( n \) integer elements, design an algorithm that computes the number of inversion pairs and runs in \( O(n \cdot \log n) \) time. Note that an inversion pair is a pair of indices \((i, j)\), such that \( i < j \) and \( A[i] > A[j] \).

\textit{Hint: Use Divide-and-Conquer.}

3. Construct the optimal binary search tree on the following four ordered keys, \( key_1 \leq key_2 \leq key_3 \leq key_4 \), with probability distribution \( p_1 = \frac{1}{2}, p_2 = \frac{1}{8}, p_3 = \frac{1}{8} \) and \( p_4 = \frac{1}{4} \).

4. In the Fractional Knapsack problem, you are given \( n \) objects \( O = \{o_1, o_2, \ldots, o_n\} \) with respective weights \( W = \{w_1, w_2, \ldots, w_n\} \) and respective profits \( P = \{p_1, p_2, \ldots, p_n\} \). The goal is to pack these objects into a knapsack of capacity \( M \), such that the profit of the objects in the knapsack is maximized, while the weight constraint is not violated. You may choose a fraction of an object, if you so decide; if \( \alpha_i, \ 0 \leq \alpha_i \leq 1 \) of object \( o_i \) is chosen, then the profit contribution of this object is \( \alpha_i \cdot o_i \) and its weight contribution is \( \alpha_i \cdot w_i \). Design a greedy algorithm for this problem and argue its correctness.

5. Argue that Randomized Quicksort takes \( O(n \cdot \log n) \) comparisons, in the expected case, to sort an array of \( n \) elements.