1. Given a sequence $a_1, a_2, \ldots, a_n$ of n distinct numbers, an *inversion* is a pair $i < j$ such that $a_i > a_j$. Note that a sequence has no inversions if and only if it is sorted in ascending order. The second and third part are to think about later.

    - Adapt the merge sort algorithm to count the number of inversions in a given sequence in $O(n \log n)$ time. You can find the detailed description of this in the Kleinberg-Tardos book (Chapter 5).
    - Call a pair $i < j$ a *significant* inversion if $a_i > 2a_j$. Describe an $O(n \log n)$ time algorithm to count the number of significant inversions in a given sequence.
    - Consider a further generalization. In addition to the sequence $a_1, \ldots, a_n$ we are given weights $w_1, \ldots, w_n$ where $w_i \geq 1$ for each i. Now call a pair $i < j$ a significant inversion if $a_i > w_j a_j$. Describe an $O(n \log^2 n)$ time algorithm to count the number of significant inversions given the sequences a and w. You can in fact obtain an $O(n \log n)$ running time.

2. Give asymptotically tight solutions to the following recurrences.

    (a) $T(n) = T(\sqrt{n}) + \log n$ for $n \geq 4$ and $T(n) = 1$ for $1 \leq n < 4$.

    (b) $T(n) = T(n/5) + T(n/10) + T(7n/10) + n$ for $n \geq 20$ and $T(n) = 1$ for $1 \leq n < 20$.