Divide and Conquer II

1. We have n nuts and n bolts. The nuts (and the bolts) are of different sizes. Each bolt fits in exactly 1 nut. We would like to match the nuts with the bolts which fit into them. Since the dimensions of the nuts and bolts are so small, we can not really tell if a nut (or a bolt) is bigger than another nut (or bolt). So the only operation that is allowed is comparing a nut and a bolt – with such a comparison, we can distinguish between three cases, namely, the nut fits a larger bolt, or the nut fits a smaller bolt, or the nut fits this bolt. Give a randomised algorithm that matches nuts with bolts. The expected number of comparisons (of a nut with a bolt) done by this algorithm should be $O(n \log n)$.
2. **Weighted medians**: You are given n numbers $a_1, a_2, \ldots, a_n$. The number a_i has a weight w_i. Find a number a_j so that the total weight of numbers less than a_j, as well as the total weight of numbers more than a_j, is at most half the total weight of the set.
3. [KT-Chapter5] We are interested in analysing some hard-to-obtain data from two databases. Each database contains n numerical values (so there are $2n$ values in total). Assume that these values are distinct. We would like to determine the median of these $2n$ values, which we define as the n^{th} smallest value. However, the only way to access these values is through queries to the databases. In a single query, we specify a value k to one of the two databases, and the chosen database returns the k^{th} smallest value that it contains. Give an algorithm that finds the median value using $O(\log n)$ queries only.
4. You are given a set S, $|S| = n$, of distinct integers and two numbers $1 \leq m_1, m_2 \leq n$. For any $x \in S$ define $rank(x) = |\{k \in S: k \leq x\}|$. Show how to find all elements in S whose rank falls in the interval $[m_1, m_2]$ in $O(n)$ time.
5. Given n points in the 2D plane, determine their convex hull in $O(n \log n)$ time using divide and conquer.