

Design and Analysis of Algorithms — Lab

R S Milton, C Aravindan, T T Mirnalinee
Department of CSE, SSN College of Engineering

Session 4: Divide and Conquer
18 January 2020

1 Matching Nuts and Bolts

We are given n nuts and n bolts of different sizes. Each nut matches exactly one bolt and vice versa. All nuts look so similar that we cannot tell if a nut is bigger than another nut. It is the same situation with bolts. However, if we try to match a nut with a bolt, the nut is either bigger or smaller or exactly a match. We want to design an algorithm to match each nut to its corresponding bolt.

1. Suppose we want to find the bolt that matches a particular nut. We can test the nut with every bolt until we find the matching bolt. We have to test the nut with $n - 1$ bolts. If none of the $n - 1$ bolts matches the nut, the n th bolt is the match for the nut. Therefore, this takes $n - 1$ tests. Using this idea, design a brute-force algorithm with $O(n^2)$ operations that associates each nut with the corresponding bolt. Implement the algorithm.
2. In Quicksort, we choose a pivot and partition a subarray into two smaller subarrays, one smaller than the pivot and the other larger than or equal to the pivot.

Since we cannot directly compare two nuts (or two bolts), we cannot use the standard partition algorithm. Choose a pivot bolt, and partition the nuts into two subsets, one subset smaller than the pivot bolt and the other larger than the pivot bolt (*not* larger than or equal, since the nuts are all distinct in size). Now we have the pivot nut in its final position. Now, using the pivot nut, partition the bolts. After these $2n - 1$ tests, we have one matched pair, and the remaining nuts and bolts partitioned into two subsets: those smaller than the pivot pair and those larger than the pivot pair. Construct the algorithm to partition nuts and bolts. Implement it.

3. Using partition algorithm for nuts and bolts, sort them recursively. Construct the algorithm and implement it.