# Michael Chillemi

# 07/02/2023

# CS 590 - Algorithms

# M7.B3: Module 7 Sorting and Selection Application Exercises

Problem 8.5.24

Bob has a set, A, of n nuts and a set, B, of n bolts, such that each nut has a unique matching bolt. Unfortunately, the nuts in A all look the same, and the bolts in B all look the same as well. The only comparison that Bob can make is to take a nut-bolt pair (a,b), such that a ∈ A and b∈B, and test if the threads of a are larger, smaller, or a perfect match with the threads of b. Describe an efficient algorithm for Bob to match up all of his nuts and bolts. What is the running time of this algorithm?

Answer:

To solve this problem I would use the end of the bolts array as the pivot in the quicksort function. Next divide the Nuts array, depending on this pivot using the partition logic. In the partition method, you must introduce a location variable that sets the location as i, where Nut[i]==pivot comparing nut with bolt. After returning the p-index after replacing it with this place. Now, split the bolts array as you did above using this nut pivot (p-index from nut). The quicksort mechanism for the else condition stays the same. The running time for this algorithm will be O(n log n).