CMPSC 122

Project 4

Collin Enders, Grant Celley, Tianyi Lin­­­

Description of quicksort

Quicksort is a comparison-based algorithm, meaning that it can sort any item as long as there is a "less-than" relation.

If we assume the list has n elements, then quicksort will obtain a O(n log2 n) complexity. In the worst case, quicksort will have an order of O(n2). We can express the number of elements (n) as 2­­m, assuming it is a power of 2 and m is a nonnegative integer. Each call to the function recQuickSort will produce two recursive calls and it will divide the list into two sub-lists of equal size. In order to reach a list of size one, we need to make m nested calls, and doing this gives us O(m) operations. In each nested call, the number of comparisons will be the sum of the size of all sub-lists minus one, which is O(n). Therefore, the maximum number of comparisons made by quicksort is O(nm). Given n=2m, we can get m=log2n. As the result, the maximum number of comparisons is O(n log2 n). The worst case of Quicksort will have a Big-O of O(n2) due to a bad choice of pivot value. If the pivot splits the array into one element on one side, and the rest of the array on the other, it will take n times to make the sublists contains one element only. And if this happens at every iteration, the number of operations will be O(n\*n), which is O(n2).

Quicksort uses the divide-and-conquer technique to sort a list. In general, it first partitions the list into two sub-lists, and then sorts each sub-list, and it combines the sub-lists into one list at the end. Its sorting process happens in its partition. The first step is to pick a pivot value. The pivot can be any value in the list, but we pick the middle element, hoping it can separate the list into two sub-lists with similar size. Then we put all elements smaller than the pivot on the left part of the array and elements greater than the pivot on the right. Values equal to the pivot, based on the textbook, stay in the right part of the array. At the end, the program will apply quicksort algorithm recursively to the left and the right parts until all elements are in order.

Quicksort is not a [stable sort](https://en.wikipedia.org/wiki/Stable_sort), because the sequence of 2 elements with equal values before and after sorting is not the same. This is because the swapping in Quicksort does not occur only with adjacent elements.