**Zach Carroll**

**2/5/19**

In this assignment you will be exploring the efficiency of two divide-and-conquer sorting algorithms, QuickSort and MergeSort. The assignment includes a text file containing **1024 random numbers**. Submit your code, the output, and answers to the questions in a zip file.

**Questions**

1. For Quicksort, what is best case performance and worst case performance as a function of input size *n*?

**Input size n = 1,024**

**Best Case – The reverse order of the unsorted array, came back with 11,636 compares.**

**Worst Case – The sorted array, with 424,736 compares.**

1. What is the best and worst case computational complexity for Quicksort?

**Best – O(n log n)**

**Worst – O(n2)**

1. What is the space complexity for Quicksort?

**O(n)**

1. For Mergesort, what is best case performance and worst case performance as a function of input size *n*?

**Avg, Best and Worst are all the same.**

**All compares came back as 10,240, or (n log n) (1024\*10)**

1. What is the best and worst case computational complexity for Mergesort?

**Best – n(log n)**

**Worst – n(log n)**

1. What is the space complexity for Mergesort?

**n**

Questions to research online. Cite your source web address.

1. The Java **Collections** **class** contains static methods including sort() that can operate on Java objects such as ArrayList, PriorityQueue, etc. What sorting algorithm is used in its sort() method, and what is its computational complexity?

**For Java 6, the Collections class uses a version of Merge Sort.**

**O(n log n)**

https://stackoverflow.com/questions/25492648/what-is-the-time-complexity-of-collectionssort-method-in-java/25492797

1. The Java **Arrays class** contains static methods for arrays, including sort().What sorting algorithm is used in its sort() method, and what is its computational complexity?

**In it’s sorting method, according to the Oracle Documentation it uses a “Dual-Pivot” Quicksorting algorithm developed by Vladimir Yaroslavisky, Jon Bentley, and Joshua Bloch. The computational complexity is O(n log n). This algorithm is typically faster than the traditional one pivot Quicksort algorithm.**

https://docs.oracle.com/javase/7/docs/api/java/util/Arrays.html