Christian Clark

CS 0445

23 March 2015

**Random Ordering**

All five algorithms had fairly similar runtimes for the arrays of size 25K, 50K, and 100K. I excluded Simple QuickSort for the larger array sizes, but the other four algorithms stayed close to each other in performance. Median of Three with a base case of 20 was the fastest throughout, albeit by a narrow margin.

**Sorted Ordering**

Simple QuickSort was significantly slower than the other four algorithms when I tested it on the smallest three array sizes. Random Pivot was the second slowest; it took roughly 1.5 and 2 times as long as the Median of Three algorithms. Again, Median of Three with a base case of 20 was the fastest of the five.

**Reversed Ordering**

Like before, Simple QuickSort's performance was much slower than the other four algorithms. The other four were very tightly clustered. Median of Three with a base case of 20 was the fastest for the arrays of size 25K, 50K, and 200K; Median of Three with a base case of 10 was the fastest for 100K and 400K; and Random Pivot was the fastest for 800K and 1600K. Based on the graph, my guess is that Random Pivot would outperform the other algorithms at even larger array sizes, but more tests would be necessary to know for certain.

**General**

Based on these results, it seems safe to say that Median of Three with a base case of 20 is the best general-purpose algorithm out of these five. It was the fastest over all of the array sizes with random and sorted ordering. The results for reversed ordering were less clear-cut, but even so, Median of Three with a base case of 20 was the fastest for three of the sizes and close to the fastest for the remaining sizes.