



Bubble Sort Complexity: Bubble sort is a relatively simple algorithm that can display high performance at its best and may outperform merge sort in its best case but on average is significantly less efficient. Bubble sort has a time complexity of $O(n^2)$ which means that it must run through many iterations for larger data sets as shown in the graph which increases exponentially. Bubble sort is only efficient for smaller data sets where it can match merge sort's efficiency.

Merge Sort Complexity: The merge sort recursive function displays much higher efficiency when compared to its bubble sort counterpart. An equation for time required to execute this function can be created resulting in $T(n) = T(n/2) + O(n)$ where $2T(n/2)$ is the time taken to solve the sub arrays and $O(n)$ is the time taken to merge the whole array. When solved, the time complexity is $O(n \log n)$, which is true for the best and worst cases as the steps taken do not change. Bubble sort's consistency can be seen by the smooth straight line the graph.