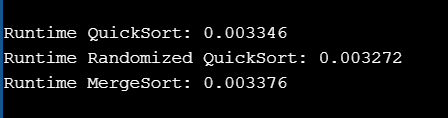
**Report for Sorting algorithms:**

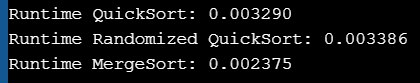
* We invoked the program for 10 times to get the average time.
* Random number generator is used for values: 0, 10000, 20000, 30000, 40000, 50000, 60000, 70000, 80000, 90000, 100000.
* To get the runtime accurately we used time functions to get the accurate percent of the start and end time. We used CLOCKS\_PER\_SEC function to divide the runtime value.

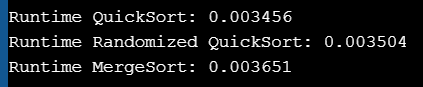
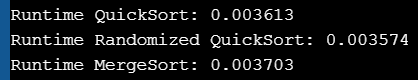
**Analysis:**

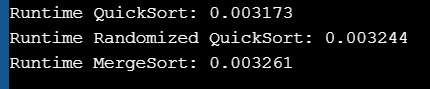
* In the given sorting algorithms quick sort have faster execution time and continued with randomized quick sort and next merge sort take the more time.
* Using the range 0-100000 quick sort still perform more faster when compared with merge sort.
* We show the complexity of the graph in different random numbers and show runtime for each sort function.

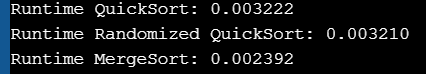
**Randomized Dataset: 20000**

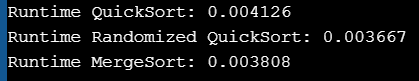
****

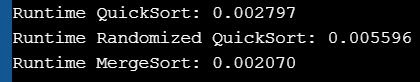
****

****

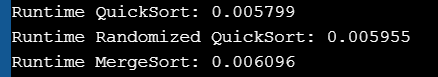
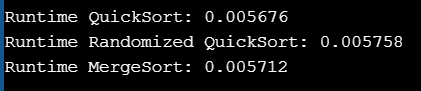
****

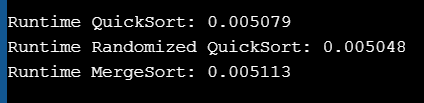
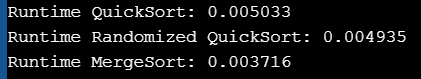
****

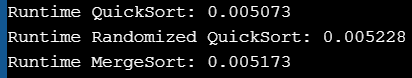
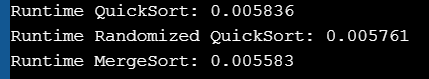
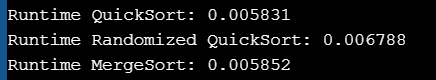
****

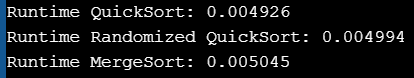
****

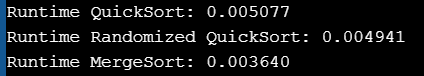
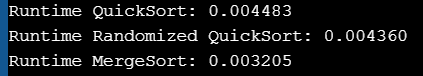
**Randomized Dataset: 30000**

****

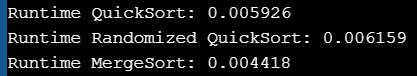
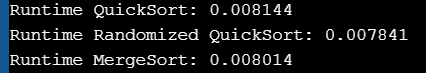
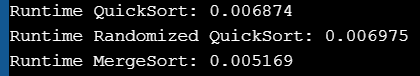
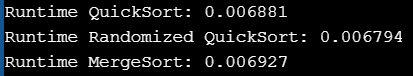
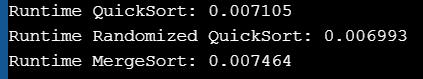
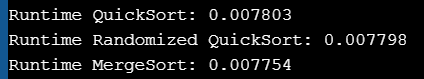
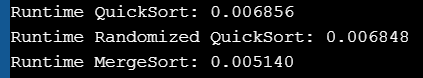
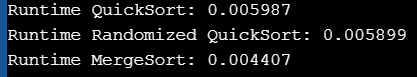
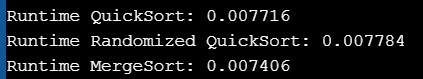
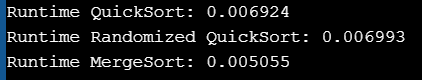
****

****

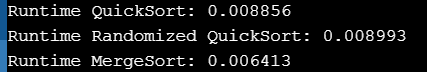
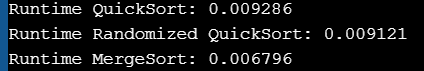
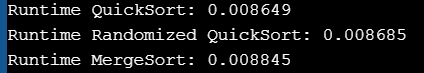
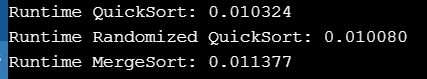
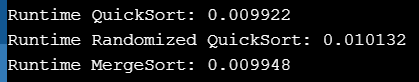
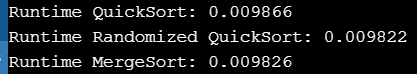
****

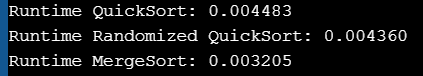
****

**Randomized Dataset: 40000**

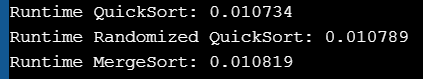
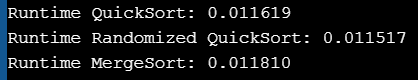
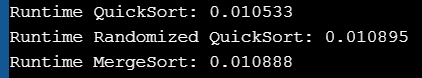
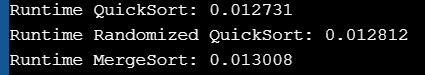
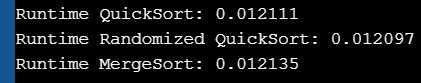
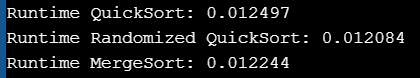
****

**Randomized Dataset: 50000**

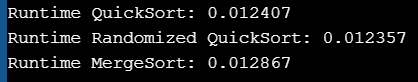
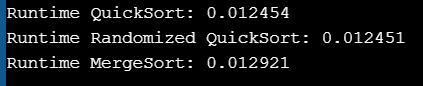
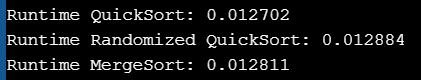
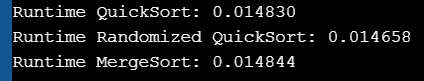
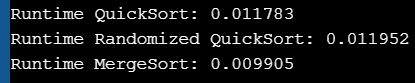
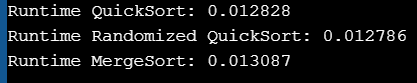
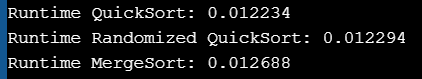
****

****

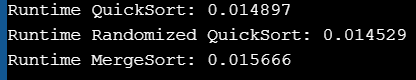
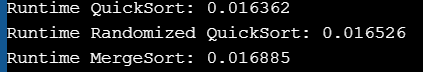
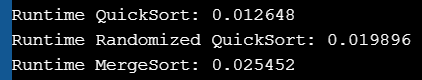
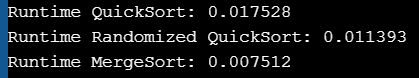
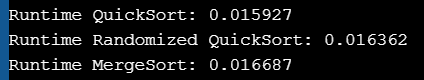
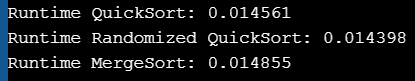
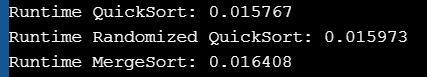
**Randomized Dataset: 60000**

****

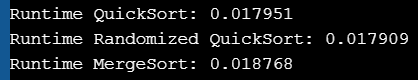
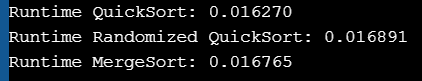
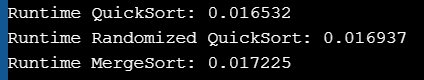
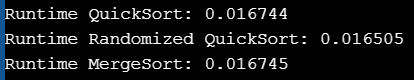
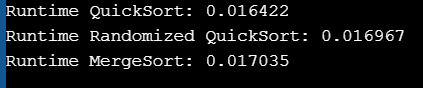
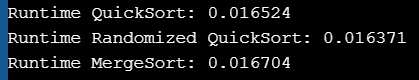
**Randomized Dataset: 70000**

****

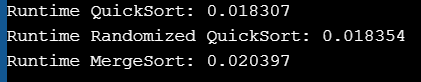
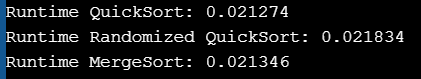
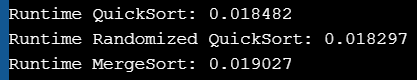
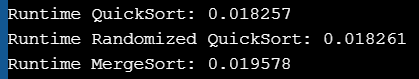
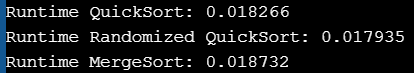
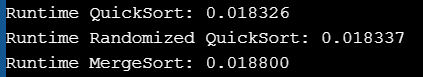
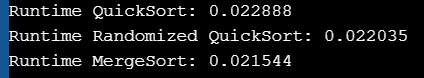
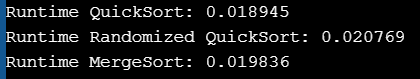
**Randomized Dataset: 80000**

****

**Randomized Dataset: 90000**

****

**Randomized Dataset: 100000**

****