**REPORT FOR SP3\_Q2**

Output for running times of Single Pivot Quick Sort vs Dual Pivot Quick Sort (in milliseconds) for an array consisting of random elements, and is of given length:

|  |  |  |
| --- | --- | --- |
| No. of Elements | Single Pivot Quick Sort (ms) | Dual Pivot Quick Sort (ms) |
| 10,000 | 16 | 3 |
| 100,000 | 35 | 32 |
| 1,000,000 | 486 | 225 |
| 10,000,000 | 33592 | 3380 |

Output for running times of Single Pivot Quick Sort vs Dual Pivot Quick Sort (in milliseconds) for an array consisting of all identical elements, and is of given length:

|  |  |  |
| --- | --- | --- |
| No. of Elements | Single Pivot Quick Sort (ms) | Dual Pivot Quick Sort (ms) |
| 10,000 | 143 | 0 |
| 100,000 | 287 | 0 |
| 1,000,000 | 486 | 15 |
| 10,000,000 | - | 30 |

As we can see, for smaller size of inputs (say below 10,000), performance of Single and Dual pivot quick sort is similar. As the number of elements increases, the performance of Single pivot quicksort degrades much quicker.

In the case of an array consisting of all identical elements, the performance of Dual pivot quicksort ends up being even better than an array consisting of dissimilar elements. This however is not the case for Single pivot quicksort. The running time increases drastically in an array of identical elements and also as the size of the array increases.