

## Runtime Results

| Bubble Sort |            |           |
|-------------|------------|-----------|
| Array Size  | Array Type | Time (μs) |
| 7           | ascending  | 0.33      |
| 8           | descending | 2.00      |
| 9           | random     | 1.33      |
| 10          | nearly     | -         |
| 11          | ascending  | 0.33      |
| 12          | descending | 4.67      |
| 13          | random     | 3.67      |
| 1300        | random     | 2041.67   |

| Selection Sort |            |           |
|----------------|------------|-----------|
| Array Size     | Array Type | Time (μs) |
| 7              | ascending  | 1.00      |
| 8              | descending | 1.33      |
| 9              | random     | 1.67      |
| 10             | nearly     | 2.00      |
| 11             | ascending  | 2.00      |
| 12             | descending | 2.00      |
| 13             | random     | 3.33      |
| 1300           | random     | 7312.33   |

| Insertion Sort |            |           |
|----------------|------------|-----------|
| Array Size     | Array Type | Time (μs) |
| 7              | ascending  | 0.33      |
| 8              | descending | 1.33      |
| 9              | random     | 1.33      |
| 10             | nearly     | 1.00      |
| 11             | ascending  | 0.67      |
| 12             | descending | 2.33      |
| 13             | random     | 2.33      |
| 1300           | random     | 2041.67   |

## Discussion of Results

The above data provides insights into the 3 implemented algorithms. As you can see, it's performance deteriorates significantly even with smaller datasets. According to the data, the runtime of Bubble Sort algorithm grows quadratically with each increase of input size. Hence, the algorithm has a worst case time complexity of  $O(n^2)$ . However, the value for time increases linearly for Selection Sort and so performs better than Bubble Sort. Although, it does have quadratic time complexity for larger datasets. Insertion Sort generally performs better than the first two algorithms for smaller datasets but still has a worst case time complexity when the dataset becomes large. In conclusion, none of these algorithms have a high level of scalability when it comes to large datasets which is why their performances decreases.