**Algorithms Analysis Report**

**Test 1**

Given Array size 100, the following results were generated, in which worst algorithm was Wrong Selection Sort (WSS). All others algorithms performed same with execution time of around 0 seconds but Wrong Selection Sort (WSS) took 0.0001 seconds to sort a random array.

**Test 2**

Given Array size 1000, the following results were generated, in which slowest algorithm was Wrong Selection Sort (WSS), which took 0.004006 seconds to sort a sorted-descending order array, on second hand, Bubble sort (BS), performed worst for sorted-descending order array, took 0.003 seconds, however, Bubble sort (BS) performed well on sorted-asc array. The best among all remained Insertion sort (IS) which performed well on every type of array and it took 0.0003503 seconds to sort a random array.

**Test 3**

Given Array size 10000, the following results were generated, in which slowest algorithm was Bubble Sort (BS), which took 0.242333 seconds to sort a sorted-descending order array. Second slowest algorithm was Wrong Selection Sort (WSS), performed worst for sorted-descending order array and, took 0.228814 seconds. Here again, Bubble sort (BS) performed well on sorted-asc array. The best among all remained Insertion sort (IS) which performed well on every type of array and it took 0.029642 seconds to sort a random array.

**Test 4**

Given Array size 100000, the following results were generated, in which the fastest algorithm was Insertion Sort (IS) which records 3.0349 seconds to sort a random array with average, and slowest algorithm was Bubble Sort (BS), which took 24.2168 seconds to sort a sorted-descending order array. Second slowest algorithm was Wrong Selection Sort (WSS), performed worst for sorted-descending order array and, took 22.753 seconds. But, Bubble sort (BS) performed well on sorted-asc array. The Correct Selection Sort (CSS) in all four tests remain average algorithm to sort any type of array.

**Discuss in detail which algorithms perform the best under what conditions? And why? Which are worst? And why?**

**Best Algorithms:** Insertion Sort was the best on all types of arrays, except for sorted-asc array type. It works very well with already sorted or nearly sorted arrays, which is why it performs well than other algorithms.

* **Random arrays:** Insertion Sort is faster because it performs few steps compared to Bubble Sort and Selection Sort to sort a random type array.
* **Sorted arrays:** Both, Insertion Sort and Bubble Sort, perform well, especially when the array was sorted in ascending order.

**Worst Algorithms:** Bubble Sort is the worst for random and sorted-descending arrays. It performs worst due to its swapping behaviour that make it very slow. It’s slow especially for sorted-des arrays.

* **Wrong Selection Sort:** It performs worst on sorted-des arrays, as it still has to perform inefficient searches for the minimum element in each pass.
* **Selection Sort:** Both Correct Selection Sort and Wrong Selection Sort are slower than Insertion Sort and Bubble Sort for sorted data but perform similarly on random data.

From our fours tests, we can conclude the Insertion Sort as best algorithm due to its time complexity in best case O(n) and it does not perform swapings to sort an array and its best for arrays which are already or somehow sorted so it quickly-sort them in less time.