**SORTING WORKSHEET NAME Tanner Gordon**

**Exchange Sort**

|  |  |  |
| --- | --- | --- |
| List | Comparisons | Exchanges |
| Unsorted | 4999950000 | 250104334 |
| Sorted | 4999950000 | 0 |
| Reversed | 4999950000 | 2498856657 |
| Identical Elements | 4999950000 | 0 |

**Insertion Sort**

|  |  |  |
| --- | --- | --- |
| List | Comparisons | Exchanges |
| Unsorted | 2498956656 | 2498856657 |
| Sorted | 99999 | 0 |
| Reversed | 2501143343 | 2501043344 |
| Identical Elements | 99999 | 0 |

**Selection Sort**

|  |  |  |
| --- | --- | --- |
| List | Comparisons | Exchanges |
| Unsorted | 5000050000 | 99994 |
| Sorted | 5000050000 | 0 |
| Reversed | 5000050000 | 99990 |
| Identical Elements | 5000050000 | 0 |

Let the performance of each sort be the sum of comparisons and exchanges.

**What was the best sort for the unsorted list?**

**Insertion**

**What was the best sort for the sorted list?**

**Insertion**

**What was the best sort for the reversed list?**

**Selection**

**What was the best sort for a list of identical elements?**

**Insertion**

**What is (in your opinion) the best general-purpose simple sorting algorithm?**

**Insertion sort is very appealing to me because it uses less operations on average.**