|  |  |  |
| --- | --- | --- |
| **Selection Sort** | | |
| **List Size** | **Comparisons** | **Time (seconds)** |
| **1,000 (observed)** | 499500 | 0.062226295471191406 |
| **2,000 (observed)** | 1999000 | 0.2281038761138916 |
| **4,000 (observed)** | 7998000 | 0.8923320770263672 |
| **8,000 (observed)** | 31996000 | 3.3776450157165527 |
| **16,000 (observed)** | 127992000 | 13.139045000076294 |
| **32,000 (observed)** | 511984000 | 53.39957594871521 |
| **100,000 (estimated)** | 49999500000 | 212 |
| **500,000 (estimated)** | 1.25 \* 10^11 | 848 |
| **1,000,000 (estimated)** | 5E+11 | 3392 |
| **10,000,000 (estimated)** | 5E+13 | 13568 |

|  |  |  |
| --- | --- | --- |
| **Insertion Sort** | | |
| **List Size** | **Comparisons** | **Time (seconds)** |
| **1,000 (observed)** | 246992 | 0.055228233337402344 |
| **2,000 (observed)** | 1016724 | 0.1919231414794922 |
| **4,000 (observed)** | 3991272 | 0.7501962184906006 |
| **8,000 (observed)** | 16104203 | 3.039889335632324 |
| **16,000 (observed)** | 64651458 | 12.393481016159058 |
| **32,000 (observed)** | 257475128 | 47.82621908187866 |
| **100,000 (estimated)** | 2499975000 | 188 |
| **500,000 (estimated)** | 6.25E+10 | 752 |
| **1,000,000 (estimated)** | 2.5E+11 | 3008 |
| **10,000,000 (estimated)** | 2.5E+13 | 12032 |

1. Which sort do you think is better? Why?

* Insertion sort is better because it has better times and fewer comparisons

1. Which sort is better when sorting a list that is already sorted (or mostly sorted)? Why?

* Insertion sort is better because it only swaps based on the minimum value, and has a best case time complexity of O(n)

1. You probably found that insertion sort had about half as many comparisons as selection sort. Why? Why are the times for insertion sort not half what they are for selection sort? (For part of the answer, think about what insertion sort has to do more of compared to selection sort.)

* Because swapping items in selection sort takes longer than searching for the desired item and then swapping