|  |  |  |
| --- | --- | --- |
| **Selection Sort** | | |
| **List Size** | **Comparisons** | **Time (seconds)** |
| **1,000 (observed)** | 499500 | 0.064 |
| **2,000 (observed)** | 1999000 | 0.253 |
| **4,000 (observed)** | 7998000 | 1.013 |
| **8,000 (observed)** | 31996000 | 4.380 |
| **16,000 (observed)** | 127992000 | 16.707 |
| **32,000 (observed)** | 511984000 | 67.562 |
| **100,000 (estimated)** | 4995000000 | 640 |
| **500,000 (estimated)** | 1.24875e11 | 16000 |
| **1,000,000 (estimated)** | 4.995e11 | 64000 |
| **10,000,000 (estimated)** | 4.995e13 | 6400000 |

|  |  |  |
| --- | --- | --- |
| **Insertion Sort** | | |
| **List Size** | **Comparisons** | **Time (seconds)** |
| **1,000 (observed)** | 247987 | 0.069 |
| **2,000 (observed)** | 1018718 | 0.321 |
| **4,000 (observed)** | 3995265 | 1.147 |
| **8,000 (observed)** | 16112196 | 4.673 |
| **16,000 (observed)** | 64667449 | 18.635 |
| **32,000 (observed)** | 257507120 | 76.040 |
| **100,000 (estimated)** | 2479870000 | 690 |
| **500,000 (estimated)** | 6.199675e10 | 17250 |
| **1,000,000 (estimated)** | 247987000000 | 69000 |
| **10,000,000 (estimated)** | 24798700000000 | 6900000 |

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

I think that selection short is better because it is slightly quicker.

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

I think insertion sort is better when a list is already sorted because there are less comparisons.

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.)

Insertion sort only compares about half as many times because it the algorithm doesn’t find the minimum of the list every time like it does in selection sort. The time does not reflect this however because of the shifting time that occurs during an insertion sort.