

Lab 6

Selection Sort		
List Size	Comparisons	Time (seconds)
1,000 (observed)	5427	0.0652
2,000 (observed)	12461	0.2487
4,000 (observed)	60743	1.0288
8,000 (observed)	133699	17.4498
16,000 (observed)	287319	66.661
32,000 (observed)		
100,000 (estimated)		
500,000 (estimated)		
1,000,000 (estimated)		
10,000,000 (estimated)		

Insertion Sort		
List Size	Comparisons	Time (seconds)
1,000 (observed)	2234	0.0554
2,000 (observed)	5459	0.3333
4,000 (observed)	35667	1.0225
8,000 (observed)	77576	18.0066
16,000 (observed)	151588	70.0101
32,000 (observed)		
100,000 (estimated)		
500,000 (estimated)		
1,000,000 (estimated)		
10,000,000 (estimated)		

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

The insertion sort requires less comparisons so it is better.

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

The selection sort would have less comparisons if the list is almost sorted.

3. 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.)

The insertion sort has more non-comparison operations to perform than the selection sort.