

Lab 7

| Selection Sort | | |
|------------------------|-------------|----------------|
| List Size | Comparisons | Time (seconds) |
| 1,000 (observed) | | |
| 2,000 (observed) | | |
| 4,000 (observed) | | |
| 8,000 (observed) | | |
| 16,000 (observed) | | |
| 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) | | |
| 2,000 (observed) | | |
| 4,000 (observed) | | |
| 8,000 (observed) | | |
| 16,000 (observed) | | |
| 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?
2. Which sort is better when sorting a list that is already sorted (or mostly sorted)? Why?
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.)

Lab 7

| | Number of Quicksort Comparisons | |
|--|---------------------------------|---------------------|
| Starting List | pivot = first | pivot = median of 3 |
| Ordered, ascending | | |
| n = 100 | | |
| n = 200 | | |
| n = 400 | | |
| n = 800 | | |
| Random | | |
| n = 100 (average 10 runs) | | |
| n = 200 (average 10 runs) | | |
| n = 400 (average 10 runs) | | |
| n = 800 (average 10 runs) | | |
| | | |
| Observed Big O() behavior, ordered with pivot = first : | | |
| Observed Big O() behavior, ordered with pivot = median of 3 : | | |
| Observed Big O() behavior, random with pivot = first : | | |
| Observed Big O() behavior, random with pivot = median of 3 : | | |
| For random list, observation regarding using first vs. median of 3 : | | |