Jaylon Ignacio

CSE 2020

Lab 14

**1. Does one sorting function consistently perform better than the other? State what you**

**see.**

the insertionSort performs better than selectionSort as shown by the values held within the ops variables

**2. Explain your observations in 1. For this, go back to the implementation of both functions**

**and/or the examples of the lecture. What explains your observation (you have seen**

**something that clearly stands out)?**

Yes, selectionSort’s efficiency depends on the size of the array because through each iteration it compares every element against one another in order to create a sorted array which explains why the number of operations is unchanged throughout each test. insertionSort only compares what it needs to compare.

**3. Focus on the performance data produced by the first four runs that processed vectors of**

**numbers that were already in order, first ascending, then descending. Would you say**

**that these represent “best” and “worst” cases for the underlying sorting algorithms?**

**Does a best and worst case exist for each? State and explain your findings.**

For the insertionSort the ascending order was the best case because it resulted in the least number of operations and the descending order was the worst case for it because it resulted in the greatest number of operations to sort it. For the selectionSort the number of operations remained constant for each test which asserts that there are no best and worst case for selectionSort.