**CSCI B505 – Programming Assignment 1 -** Bubble sort vs Insertion Sort

**1.1 Plots**

Plot 1 - Random Inputs

Below describes execution time lines of Insertion sort vs Bubble sort where insertion sort performed Bubble sort for Random inputs.

Plot 2 - Non decreasing inputs

In the below graph, Insertion sort out performed Bubble sort and the performance is linear

Plot 3 - Non increasing inputs

For the non-increasing inputs, bubble sort out performed insertion sort

**2.1 Platform**

All the metrics are calculated in a single machine with below configuration

Processor: 2.5 Ghz Intel core i7

Memory: 16 GB

Operating System: macOS Sierra

Tools used: Pycharm, Microsoft Excel

**2.2 Coding/plotting choices**

The coding is done in Python 2.7 and plotting is made with Microsoft office tools.

**2.3 Base files**

The data used for sorting is generated through a Python script. The script will generate 3 types of files with integer elements ranging from 2000 to 50000 on each

1. Random elements (R1.txt,R2.txt….R25.txt)
2. Non Decreasing elements (I1.txt,I2.txt….I25.txt)
3. Non Increasing elements (D1.txt,D2.txt….D25.txt)

All these files are persisted in the disk and sorted in memory.

**2.4 Observation**

a. The completion time is high for Python

b. Program consumed 98% of the processor and it executed in a single thread

c. Usage of “xrange” instead of “range” retrieved better results

**3. Conclusion**

For plot 1 both the algorithm displayed quadratic running time with Insertion sort even performing better, insertion sort completed the 8 times faster than bubble sort on the same set of data.

For plot2, since the data is already sorted, the insertion sort’s execution time is logarithmic however the latter’s is quadratic as it has to complete both the condition to complete the execution.

For plot 3, we can consider the data as worst case scenario to both the algorithm, in this scenario Bubble sort out performed Insertion reason being only 2 loops are required for each element to complete the execution whereas in insertion sort multiple iterations are involved.