**INFO 6205**

**Program Structures & Algorithms**

**Fall 2018**

**Assignment 3**

In this assignment, I conducted a benchmark experiment for three different sorting algorithms. *InsertionSort*, *SelectionSort* and *ShellSort*.

1. Conclusion

Some useful abbreviations:

* *n* – size of the array needs to be sorted.
* *m* – time of experiment for every algorithm.

After sufficient experiments (with 5 different large enough n and for every n we have m(m>100) times experiments), we get the conclusion for the benchmark of different algorithms.

Performance for different Data set:

Random Order: *SelectionSort > InsertionSort > ShellSort;*

All three methods shows the time complex of O()

Sorted Order: *InsertionSort / ShellSort >>> SelectionSort;*

*InsertionSort/ShellSort shows constant time for sorting a sorted array.*

*However, the time of SelectionSort increase more than quadraticly.*

Reverse Order: *SelectionSort > InsertionSort > ShellSort;*

Partly Sorted: *SelectionSort > InsertionSort > ShellSort*;

So we could draw the conclusion that:

*SelectionSort* is stable and inefficient, it is always with quadratic complexity without the condition of the case;

*InsertionSort* is fastest for best case because is linear.

*ShellSort* is the most stable one.

1. Graph and explain of various situation for different algorithms
   1. For Random data

Source: produced by nextInt() method;

As you can see from the image, that for random Data, *SelectionSort* is the best, the difference is obvious with the increment of size *n.*

* 1. For Sorted Data

Source: Produced with a for loop;

As you can see, for sorted data, it costs almost no time for *InsertionSort* and *ShellSort*, so only the *SelectionSort* could be seen due to the huge gap between running time.

* 1. For Reverse Ordered Data

Source: Produced with a for loop;

As you can see, for reversed ordered data, *ShellSort* is slowest, and *SelectSort* is fastest.

* 1. For Partly Ordered Data

Source: 1/3 is random, 1/3 is sorted, 1/3 is reversed;

As you can see, the *Shellsort* is bad for Partly Ordered Data, and *SelectionSort* is the best.