Yinuo Li ECE368 Project1 Report

Sequence time complexity: O((logn/log3)^3)
Sequence space complexity: O((logn/log3)^2)

Suppose n is the number of elements to be sorted. The rows of pyramid is logn/log3.
The basement of pyramid has logn/log3 elements.
So, sequence space complexity: 0.5*(logn/log3)^2.
I wrote the recursive sub-functions to calculate the power of two and three.
So, sequence time complexity: (logn/log3)^3

Insertion

	10000.txt	100000.txt	1000000.txt
N_Comp	4.857450e+05	7.250792e+06	1.007193e+08
N_Move	5.619710e+05	8.283011e+06	1.140415e+08
I/O time	0	3.000000e-02	3.400000e-01
Sorting time	0	4.000000e-02	4.700000e-01

Selection

	10000.txt	100000.txt	1000000.txt
N_Comp	1.491996e+08	1.498216e+10	/
N_Move	2.040360e+05	2.811402e+06	/
I/O time	0	4.000000e-02	/
Sorting time	2.400000e-01	2.361000e+01	/

Shell insertion sort time complexity: O(n*(logn)^2).
The insertion sorting time, number of moves and comparisons grows with the order of n*(logn)^2.

Shell selection sort time complexity: O(n^2).
The slection sorting time, number of moves and comparisons grows with the order of n^2.

Space complexity of of my sorting routines is O(n).
The complexity of the additional memory required by my routine is the sequence space which is O((logn/log3)^2).