Alex Fyffe

ECE 368

PUID: 00279-00213

7/9/17

Programming Assignment 2

Results:

|  |  |  |
| --- | --- | --- |
| SHELL SORT (IMPLEMENTING INSERTION SORT) | | |
| Test Cases | **Sorting Time** | **I/O Time** |
| 15 | Approximately 0 | Approximately 0 |
| 1000 | Approximately 0 | Approximately 0 |
| 10000 | 0.17 | Approximately 0 |
| 100000 | 43.200001 | 0.01 |
| 1000000 | N/A | N/A |

From the tables, you can see that the sorting time and the I/O time increases as the number needing to be sorted gets larger. Based off of the five test cases, if n represents the number of elements in the sequence, the space complexity is O(n), with an additional O(1) because of the sequence linked list. The time complexity is probably closer to O(n2) due to how long each test case takes.

|  |  |
| --- | --- |
| SHELL SORT WITH INSERTION SORT | |
| Test Case | **Total Time** |
| 15 | Approximately 0 |
| 1000 | Approximately 0 |
| 10000 | Approximately 0 |
| 100000 | 0.02 |
| 1000000 | 0.038 |

This table from the first assignment shows that the shell sort with an array may be faster than a linked list. This is probably because it divided up the array instead of having to traverse through it every single time. The I/O run time seems to be near the same though. Therefore, although space is optimized in this algorithm, the time is not optimized very well.