PROG20799 Project, Prof. Georg Feil

Student Name: Heyzzen Flores

Student ID: 991 452 883

StudentList.txt

|  |  |  |
| --- | --- | --- |
| **Run** | **Sorting A Time (sec)** | **Sorting B Time (sec)** |
| 1 | 1.276110 | 0.017800 |
| 2 | 1.432236 | 0.019312 |
| 3 | 1.071757 | 0.029284 |
| 4 | 1.221190 | 0.011007 |
| 5 | 1.714489 | 0.022075 |

Shortest time for A: 1.071757 (run 3)

Shortest time for B: 0.011007 (run 4)

StudentListHalf.txt

|  |  |  |
| --- | --- | --- |
| **Run** | **Sorting A Time (sec)** | **Sorting B Time (sec)** |
| 1 | 0.287274 | 0.008432 |
| 2 | 0.323879 | 0.005744 |
| 3 | 0.256493 | 0.009333 |
| 4 | 0.283705 | 0.006895 |
| 5 | 0.321120 | 0.011747 |

Shortest time for A: 0.256493 (run 3)

Shortest time for B: 0.005744 (run 2)

Running both sorts I found that Sort A takes more time to sort a list with 50000 data, while Sort B is much quicker. This is true because I used Insertion sort for Sorting A, which takes a computational complexity of O(n^2) while Sorting B takes O(n log n), and while running the program 5 times each. I have also noticed that the more data that Sorting A needs to sort, the longer it takes, this can be shown with the data above, the first data file contained around 50,000 data while the second data contained only half of the first data file. Also, when running this program, Sorting B had a small difference during the time running both the first and second data set.