|  |  |  |  |
| --- | --- | --- | --- |
| File Base Name | Number of Orders processed | Single Thread time (msec) | Multiple Thread time (msec) |
| LargeSet | 100 | 7422 | 3606 |
| LargeSet | 200 | 9195 | 4341 |
| LargeSet | 300 | 12856 | 7827 |
| LargeSet | 400 | 16989 | 15308 |
| LargeSet | 500 | 22877 | 22111 |

**Report**

While processing the LargeSet.txt order files, the multithreaded approach was faster than the single threaded approach. I think the multiple threading is faster because it allows the compiler to concurrently evaluate multiple order files at a time. Whereas for the single threaded approach, it might take some time for the compiler to evaluate each order sequentially once at a time. Another reason can be that the multithreaded approach creates multiple threads at the same time and runs them all at once, this might save time. However, the single threaded process keeps re-initializing/using the same thread and only process one order at a time. I believe this might compromise the speed since the data set provided has high amount of order receipts.