Base Data File: SecDataSetFive

Number of Items per Order: 10 million

Items Data: itemsData.txt

|  |  |  |
| --- | --- | --- |
| Number of Orders (10 million items per order): | Single-Thread Time (seconds): | Multi-Thread Time (seconds): |
| 4 | 31.75 | 13.803 |
| 5 | 43.214 | 14.3 |
| 6 | 47.93 | 19.457 |

The table above depicts the advantages of using multithreading over single threading. Since the advantages of multithreading are more obvious which much larger orders, each order is set to have 10 million items. The reason multithreading allows for much shorter processing times because the CPU can concurrently process all the orders at once with different threads instead of single threading in which the CPU processes all the orders one by one. As shown in the table, regardless of the number of orders, the time it takes to process the orders with single threading is almost 2-3 times the time it takes to process the orders with multi-threading. Thus, it can be concluded that for very large orders, multithreading is indeed much more time efficient.