**Mini Project 1: Transactional Programming**

Colin Grundey

Elton Jugi Gladstone Pushparaj

Make Notes:

Run “make” command to make all targets. Run “make test” to make the given bank account test. Run “make custom\_test” to make our custom test.

Our implementation works as intended and the performance results are shown in the graphs and tables below. These results were obtained by running the given bank account test program. The first compares the throughput values when changing the bit filter size and the thread count. The second compares the average abort count of each thread. They show a trend that is consistent with what is expected of the algorithm. In addition, running the system with disjoint access still produces aborts as expected because the Bloom Filter can give false aborts. Finally, we created a scenario to test the RingSTM where there is a fee being collected of $10,000. It runs successfully and shows the proper operation of the algorithm.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Average Abort count |  |  |  |  |
|  | 1024 | 512 | 256 | 128 |
| 1 Thread | 0 | 0 | 0 | 0 |
| 2 Threads | 118651.5 | 166807.5 | 196708.5 | 210322 |
| 4 Threads | 177759.75 | 217077.25 | 246829 | 268857.25 |
|  |  |  |  |  |
| Throughput |  |  |  |  |
|  | 1024 | 512 | 256 | 128 |
| 1 Thread | 736841 | 725897 | 749894 | 750512 |
| 2 Threads | 970425 | 1246677 | 1304937 | 1296741 |
| 4 Threads | 1246561 | 1964827 | 2003610 | 2055363 |