|  |  |  |
| --- | --- | --- |
| **ARRAY\_SIZE (# of ints)** | **Single Thread Time (ms)** | **Multi Thread Time (ms)** |
| 10 | 0 | 56 |
| 100 | 0 | 112 |
| 1000 | 0 | 102 |
| 10000 | 2 | 94 |
| 100000 | 21 | 45 |
| 1000000 | 218 | 168 |
| 10000000 | 2467 | 1562 |

2. I was expecting quite a lot of speed up. Even using just 10 threads I expected speed up factor to be a factor of around 7. I thought sorting independently per thread and then combining into a merge would not take too long. I thought the sort algorithm would take the longest time and merging would not take too much time, but evidently, merging the threads took a lot of run-time.

3. I did not meet the speed up I expected. I postulate that the reason was because the algorithm to merge the multithread sorted arrays was very run-time heavy, and waiting for the threads to join took a lot of run-time. When using a single thread, the merge algorithm took O(2n) run-time, whereas the multithread merge algorithm took O(n^2) time.