**Results For Runs:**

**Method 1:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **c** | **d** | **Sum** | **User time(s)** | **System time(s)** | **Total time(s)** |
| 880130203012 | 922823372203 | 1 | 1 | 4.053457e+16 | 148.82 | 0.01 | 2:29.49 |
| 880130203012 | 922823372203 | 2 | 1 | 2.035143e+16 | 77.86 | 0.01 | 1:18.32 |
| 880130203012 | 922823372203 | 4 | 1 | 1.016984e+16 | 40.31 | 0.03 | 40.426 |
| 880130203012 | 922823372203 | 8 | 1 | 8.228345e+15 | 40.22 | 0.26 | 38.914 |

**Method 2:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **c** | **d** | **Sum** | **User time(s)** | **System time(s)** | **Total time(s)** |
| 880130203012 | 922823372203 | 1 | 2 | 4.053457e+16 | 146.71 | 0.01 | 2:27.17 |
| 880130203012 | 922823372203 | 2 | 2 | 4.053457e+16 | 146.64 | 0.01 | 2:27.38 |
| 880130203012 | 922823372203 | 4 | 2 | 4.053457e+16 | 146.67 | 0.01 | 2:26.99 |
| 880130203012 | 922823372203 | 8 | 2 | 4.053457e+16 | 146.66 | 0.01 | 2:27.37 |
| 880130203012 | 922823372203 | 16 | 2 | 4.053457e+16 | 146.69 | 0.01 | 2:27.20 |
| 880130203012 | 922823372203 | 32 | 2 | 4.053457e+16 | 146.66 | 0.02 | 2:27.27 |

**Method 3:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **c** | **d** | **Sum** | **User time(s)** | **System time(s)** | **Total time(s)** |
| 880130203012 | 922823372203 | 1 | 3 | 4.053457e+16 | 146.67 | 0.01 | 2:26.98 |
| 880130203012 | 922823372203 | 2 | 3 | 4.053457e+16 | 75.90 | 0.01 | 1:16.16 |
| 880130203012 | 922823372203 | 4 | 3 | 4.053457e+16 | 39.35 | 0.05 | 39.478 |
| 880130203012 | 922823372203 | 8 | 3 | 4.053457e+16 | 22.29 | 0.17 | 22.827 |
| 880130203012 | 922823372203 | 16 | 3 | 4.053457e+16 | 22.23 | 0.33 | 23.187 |
| 880130203012 | 922823372203 | 32 | 3 | 4.053457e+16 | 22.30 | 0.19 | 22.740 |

**Questions and Answers:**

**1. Which method(s) provide the correct result and why?**

Method 2 and method 3 provide correct results because we use mutex that prevents accessing and changing to data at the same time.

**2. Among the method(s) providing the correct result, which method is the fastest?**

Method 3 is the fastest. We are using multiple threads and localSqrtSum for each thread. These each help run the code faster and provide correct results.

**3. Among the method(s) providing the correct result, does increasing the number of threads always result in smaller total time? Discuss this considering the number of CPU cores available in your computer (in Linux, lscpu command provides the number of CPU cores available in your computer).**

I have MacBook machine that has 8 cores CPU. Increasing the number of threads always reduce the total time of code until 8 threads. But after 8 threads, it is not going to affect my code speed.

**4. Are there any differences in user time/system time ratio of the processes as the number of threads increases? What could be the cause of these differences?**

System time seems to increase proportional to the thread count, whereas the user time is inversely proportional. We have decided to adjust the time values according to each thread on this document (i.e. A 200s time on a 2-thread system translated to 100s per thread) since we believed that represented the data better, system time was untouched by this decision. Hence, we can say the system time increases to account for the thread management and the user time decreases to account for the increased processing power of more threads as the thread count increases.