* Do you observe speedup when as T goes from 1 to 4 (interesting since there are 4 cores on your raspberry pi)?

Yes. I observed the speed up.

I first set the x being an array with 1,200,000 elements and moments to be calculated to be 100.

Then I set the thread numbers to be 1 and compile. It takes 1 min and 8 second to finish everything.

Then I changed the thread number to be 4 and compile and run again. It takes 20 seconds to do everything.

* What happens when T is larger than 4? Does having more threads help improve the performance?

Because 100 moments run so slow so I reduced the number of moments to 50.

I tried thread number equal to 12, It takes 8 seconds to do all the calculation which is the same as using 4 threads.

I tried thread number equal to 4000 and there is a segment fault. It is probably because I am allocating more memory than the raspberry Pi 2 actually have.

* What happens for very large N?

For larger N value than 1,200,000, I tried 2,000,000. The program with N= 2,000,000 and 4 threads uses 14 seconds to do the calculation while when N=1,200,000 it takes 8 seconds to do calculation.

For super large N = 72,000,000, the program has a segment fault error. It is probably because memory is not enough big to store those data.

|  |  |
| --- | --- |
| Thread | CPU time |
| 1 | 33 |
| 2 | 17 |
| 3 | 11 |
| 4 | 9 |
| 10 | 9 |
| 100 | 9 |