

## Lab 2

Time of execution without thread :217ns

```
Enter the number of inputs (N): 6
Enter input 1: 10
Enter input 2: 11
Enter input 3: 12
Enter input 4: 13
Enter input 5: 14
Enter input 6: 15
average:13Enter value for P (between 500 to 2500): 1000
Enter value for Q (between 1000 to 5000): 2000
Elapsed time: 217 ns
Final Average after two iterations: 1

...Program finished with exit code 0
Press ENTER to exit console.
```

Time of execution with thread :257350ns

```
Enter the number of inputs (N): 6
Enter input 1: 10
Enter input 2: 11
Enter input 3: 12
Enter input 4: 13
Enter input 5: 14
Enter input 6: 15
Average: 13
Enter value for P (between 500 to 2500): 1000
Enter value for Q (between 1000 to 5000): 2000
Elapsed time: 257350 nanoseconds
Final Average after two iterations: 1

...Program finished with exit code 0
Press ENTER to exit console.[]
```

**(\*have calculated time of execution of GCD block only , Have done roundoff of average unit to eradicate decimal values as GCD was to be done on integer values )**

As seen from Time of execution with thread is taking more time as compared to the sequential execution without using thread because

“There are overheads with threading (creating threads, switching context between them, etc) that take time. Unless the calculations can genuinely be done in parallel (e.g. multiple physical cores) it is quite normal that total elapsed time exceeds time to do everything in one thread” For more information:

<https://stackoverflow.com/questions/63522327/multiple-threads-taking-more-time-than-single-process>

Now to address this problem **there is a scope of hardware-software co-design in this system as** the GCD calculations could be offloaded to GCD hardware block , which could compute them in parallel, speeding up the overall computation. The average calculation could be performed by the software on the CPU.