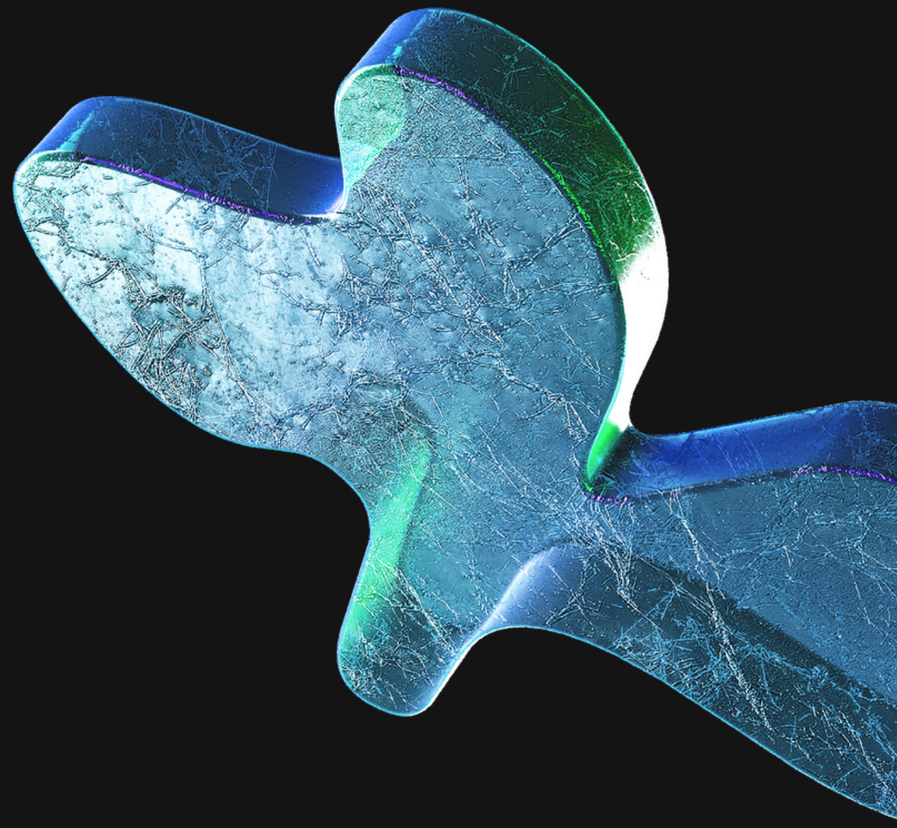
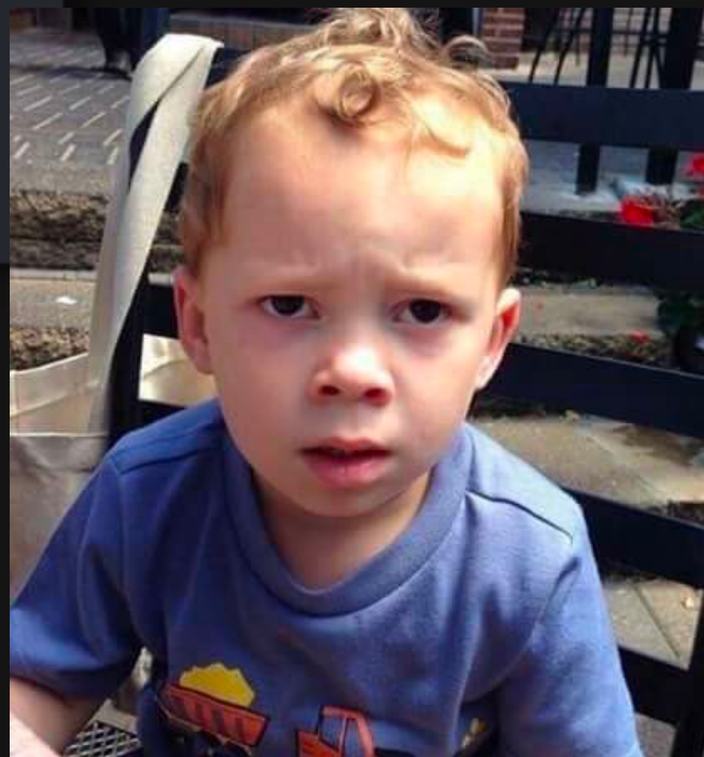
Operating Systems

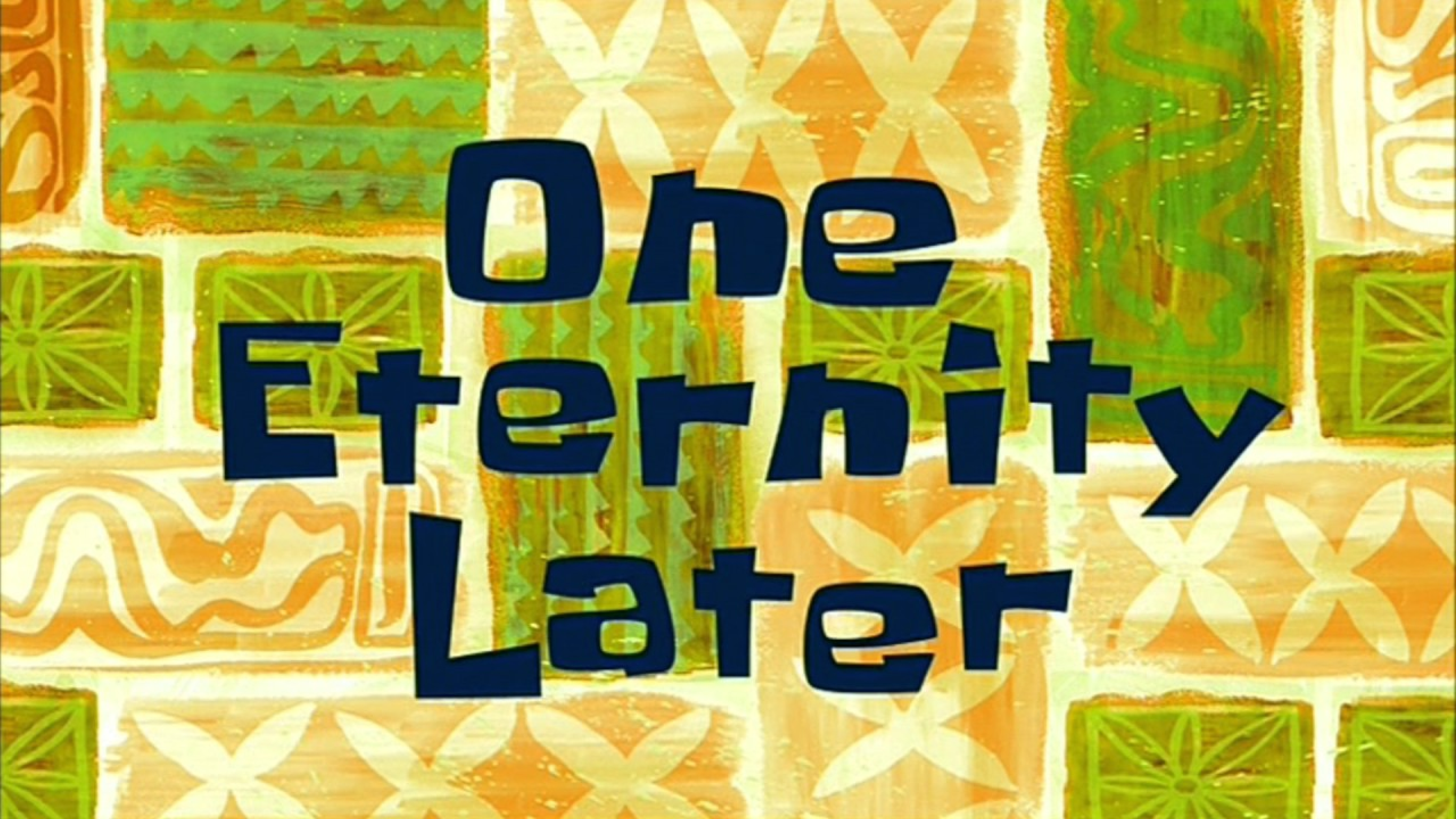


Case Study #1

Let's try!

```
Data read...Complete.


Working....
```




One
Eternity
Later


รอรักแท้

Result :

```
C:\case_study_1\problem1>dotnet run
Data read...Complete.


Working...Done.
Summation result: 888701676
Time used: 19569ms

C:\case_study_1\problem1>
```

ข้อสังเกต

First Try ?

First Try!

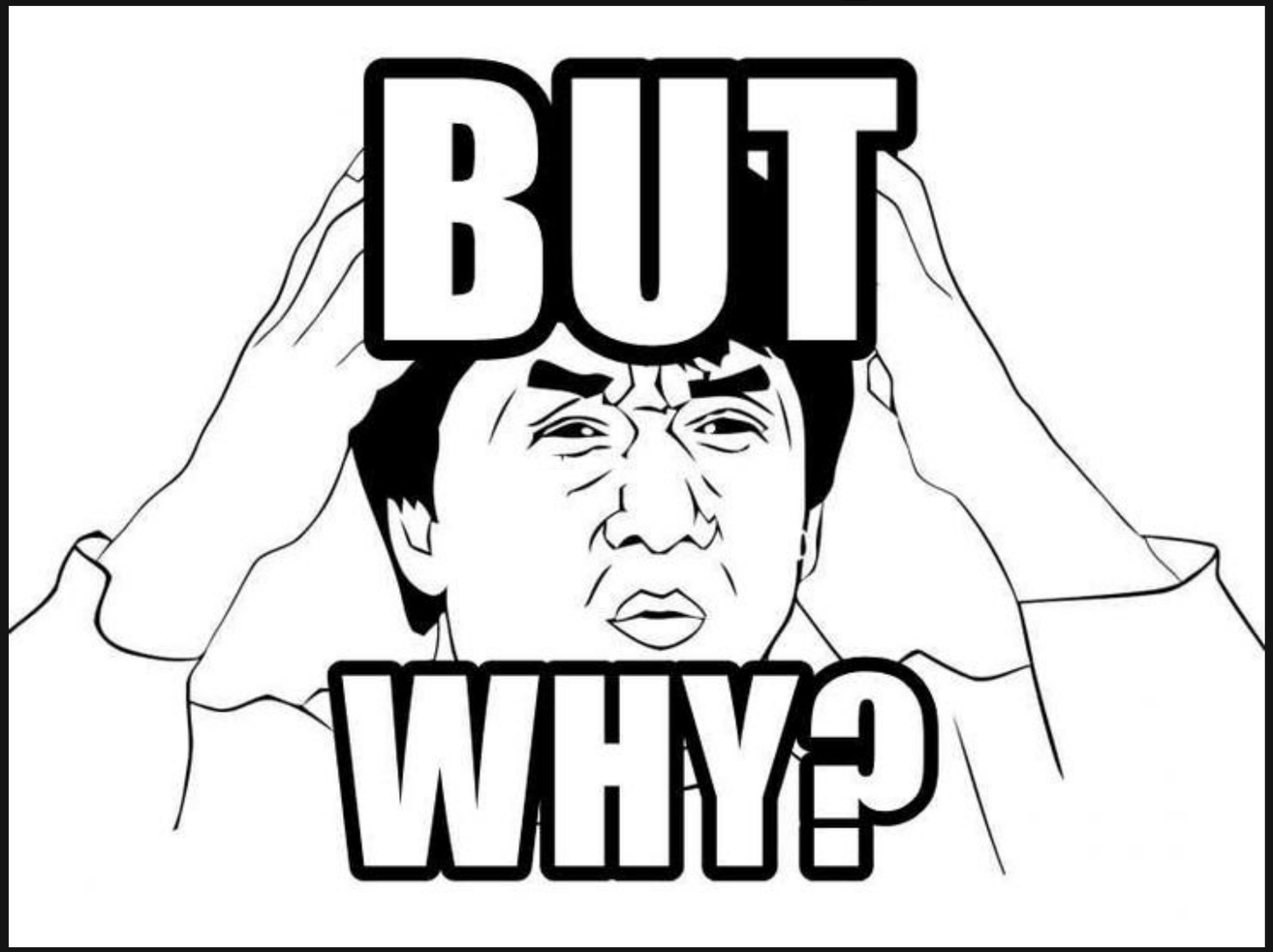
First Try

```
1
2  // fork threads
3  Thread th1 = new Thread(TestThread1);
4  Thread th2 = new Thread(TestThread2);
5
6  th1.Start();
7  th2.Start();
8  th1.Join();
9  th2.Join();
```

```
1
2  static void TestThread1()
3  {
4      for (int i=0; i<500000000; i++)
5      {
6          sum();
7      }
8  }
9
10   static void TestThread2()
11 {
12     for (int i=0; i<500000000; i++)
13     {
14         sum();
15     }
16 }
```

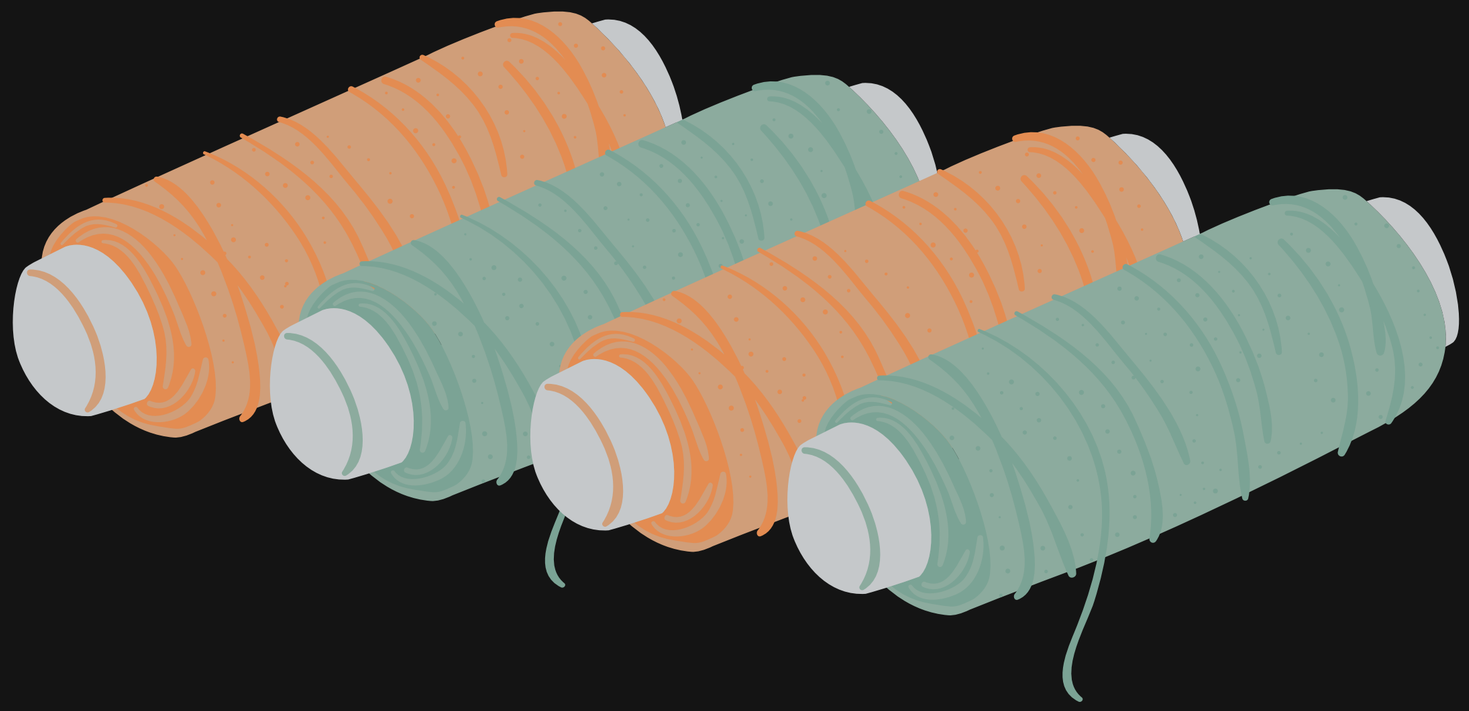
```
Data read...Complete.


Working...Done.
Summation result: 531275761
Time used: 43630ms
```

But Why ?



Version 0

Version 0



Version 0

```
1
2  static void ThreadProc1(ref long sumThread) {
3    for(int i=0;i<250000000;i++) {
4      if (Data_Global[i] % 2 == 0)
5      {
6          sumThread -= Data_Global[i];
7      }
8      else if (Data_Global[i] % 3 == 0)
9      {
10         sumThread += (Data_Global[i]*2);
11     }
12     else if (Data_Global[i] % 5 == 0)
13     {
14         sumThread += (Data_Global[i] / 2);
15     }
16     else if (Data_Global[i] %7 == 0)
17     {
18         sumThread += (Data_Global[i] / 3);
19     }
20   }
21 }
```

```
1
2  /* Start */
3  Console.Write("\n\nWorking...");
4  sw.Start();
5
6  /*Fork Threads*/
7  Thread[] threads = new Thread[numThread];
8  long[] sumThread = new long[numThread];
9  threads[0] = new Thread(() => ThreadProc1(ref sumThread[0]));
10 threads[1] = new Thread(() => ThreadProc2(ref sumThread[1]));
11 threads[2] = new Thread(() => ThreadProc3(ref sumThread[2]));
12 threads[3] = new Thread(() => ThreadProc4(ref sumThread[3]));
13
14 for(int i=0;i<numThread;i++) {
15   threads[i].Start();
16 }
17 for(int i=0;i<numThread;i++){
18   threads[i].Join();
19 }
20 for(int i=0;i<numThread;i++){
21   Sum_Global += sumThread[i];
22 }
23
24 sw.Stop();
25 Console.WriteLine("Done.");
26
27 /* Result */
28 Console.WriteLine("Summation result: {0}", Sum_Global);
29 Console.WriteLine("Time used: " + sw.ElapsedMilliseconds.ToString() + "ms");
```

Version 0.5

```
// fork threads
Thread[] threadsArray = new Thread[n];
for (int i=0; i<n; i++) {
    int start = i*interval;
    int end = start+interval;
    int k = i;
    threadsArray[i] = new Thread(()=> TestThread(start, end, ref sumthread[k]));
}
```

```
static long TestThread(int start, int end, ref long sumthred){
    int i;
    for (i=start; i<end; i++) {
        sumthred = sum(i, sumthred);
    }
    return sumthred;
}
```

```
static long sum(int i, long sum)
{
    if (Data_Global[i] % 2 == 0)
    {
        sum -= Data_Global[i];
    }
    else if (Data_Global[i] % 3 == 0)
    {
        sum += (Data_Global[i]*2);
    }
    else if (Data_Global[i] % 5 == 0)
    {
        sum += (Data_Global[i] / 2);
    }
    else if (Data_Global[i] %7 == 0)
    {
        sum += (Data_Global[i] / 3);
    }
    Data_Global[i] = 0;
    // G_index++
    return sum;
}
```

Version 1

Version 1

```
Working...Done.
Number of Threads: 4
Summation result: 888701676
Time used: 4276ms
```

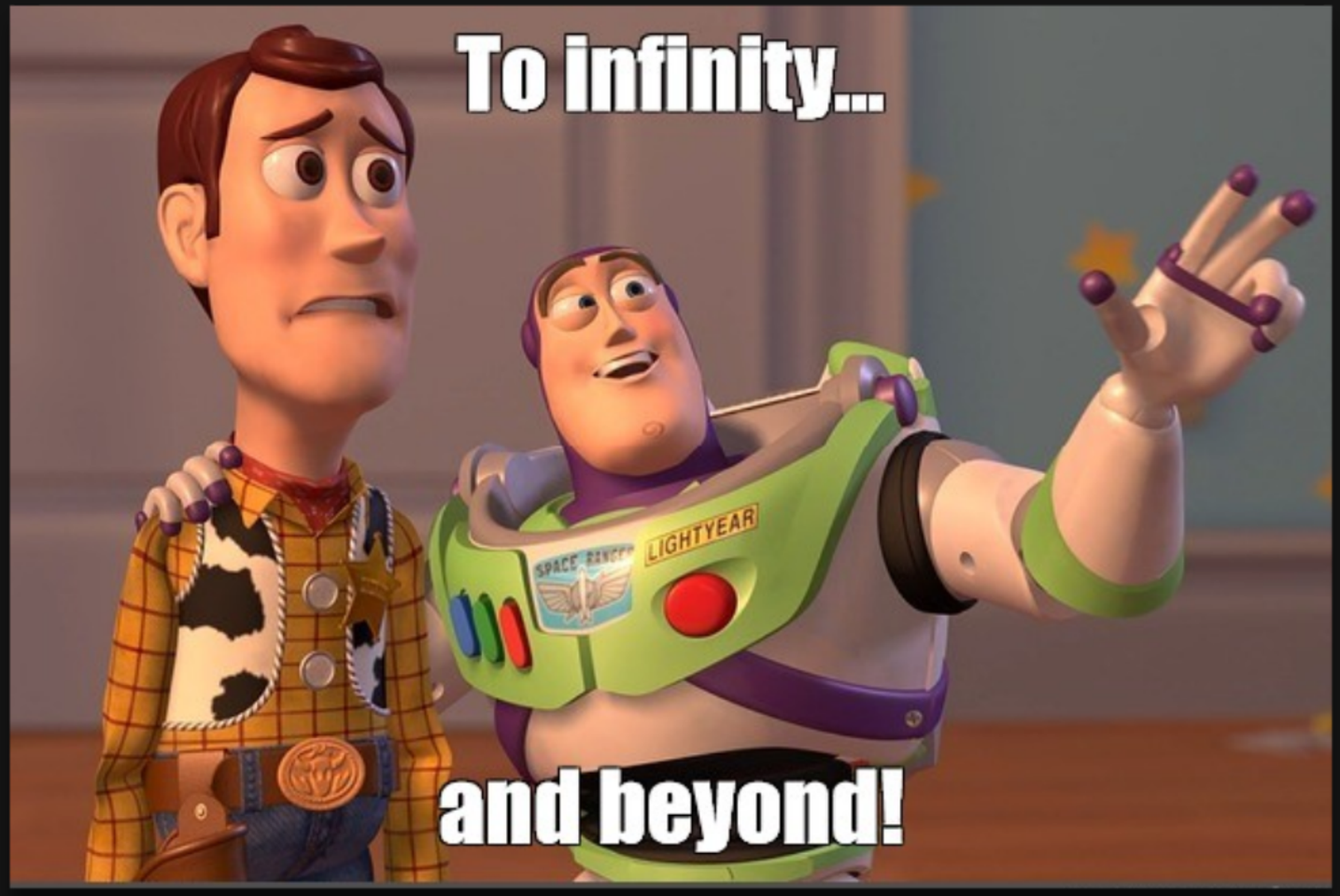
```java

// fork threads
Thread[] threadsArray = new Thread[n];
for (int i=0; i<n; i++) {
    int start = i*interval;
    int end = start+interval;
    int k = i;
    threadsArray[i] = new Thread(()=> {keepSumthread[k] = TestThread(start, end, sumthread[k]);});
}
```

```java

static long TestThread(int start, int end, long sumthred){
    int i;
    for (i=start; i<end; i++) {
        sumthred = sum(i, sumthred);
    }
    return sumthred;
}
```

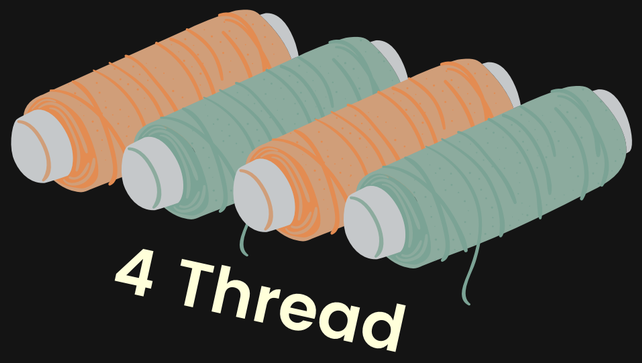
```java

static long sum(int i, long sum)
{
    if (Data_Global[i] % 2 == 0)
    {
        sum -= Data_Global[i];
    }
    else if (Data_Global[i] % 3 == 0)
    {
        sum += (Data_Global[i]*2);
    }
    else if (Data_Global[i] % 5 == 0)
    {
        sum += (Data_Global[i] / 2);
    }
    else if (Data_Global[i] %7 == 0)
    {
        sum += (Data_Global[i] / 3);
    }
    Data_Global[i] = 0;
    // G_index++
    return sum;
}
```

Version 2

Version 2

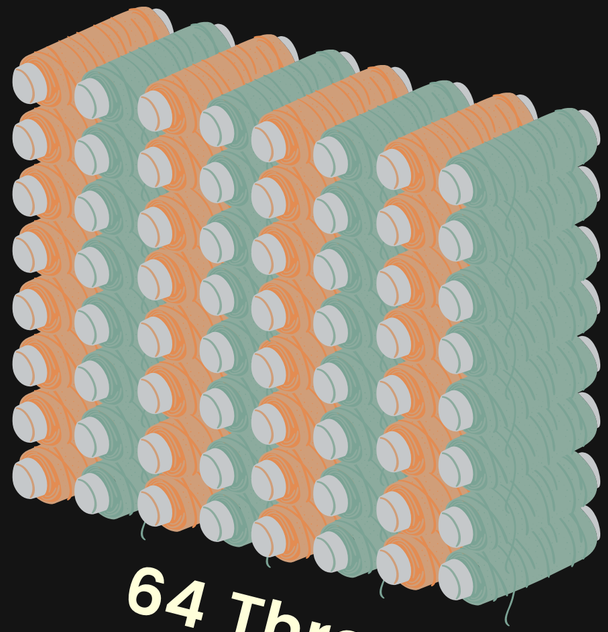


Version 2



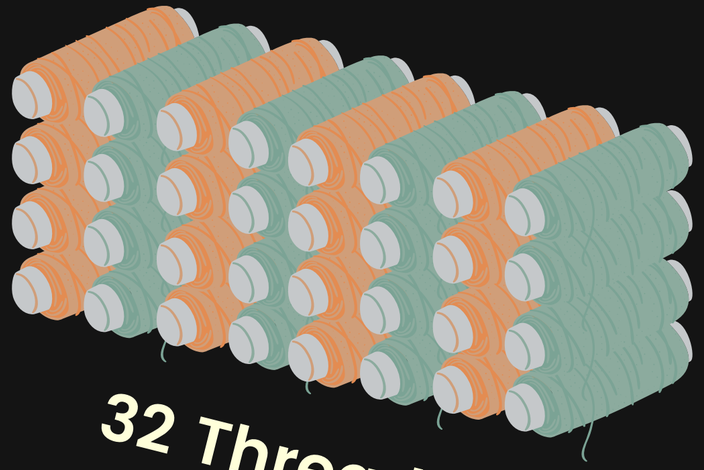
4 Thread

16 Thread



64 Thread

8 Thread



32 Thread

Version 2

Computer I

CPU: Intel Core i5 12400F
6 Cores 12 Threads
GPU: RTX 3060Ti
RAM: 32 Gbs

Computer II

CPU: AMD Rysen 5 4600H
6 Cores 12 Threads
GPU: GTX 1650Ti
RAM: 8 Gbs

Result :

```
Working...Done.
Number of Threads: 4
Summation result: 888701676
Time used: 4276ms
```

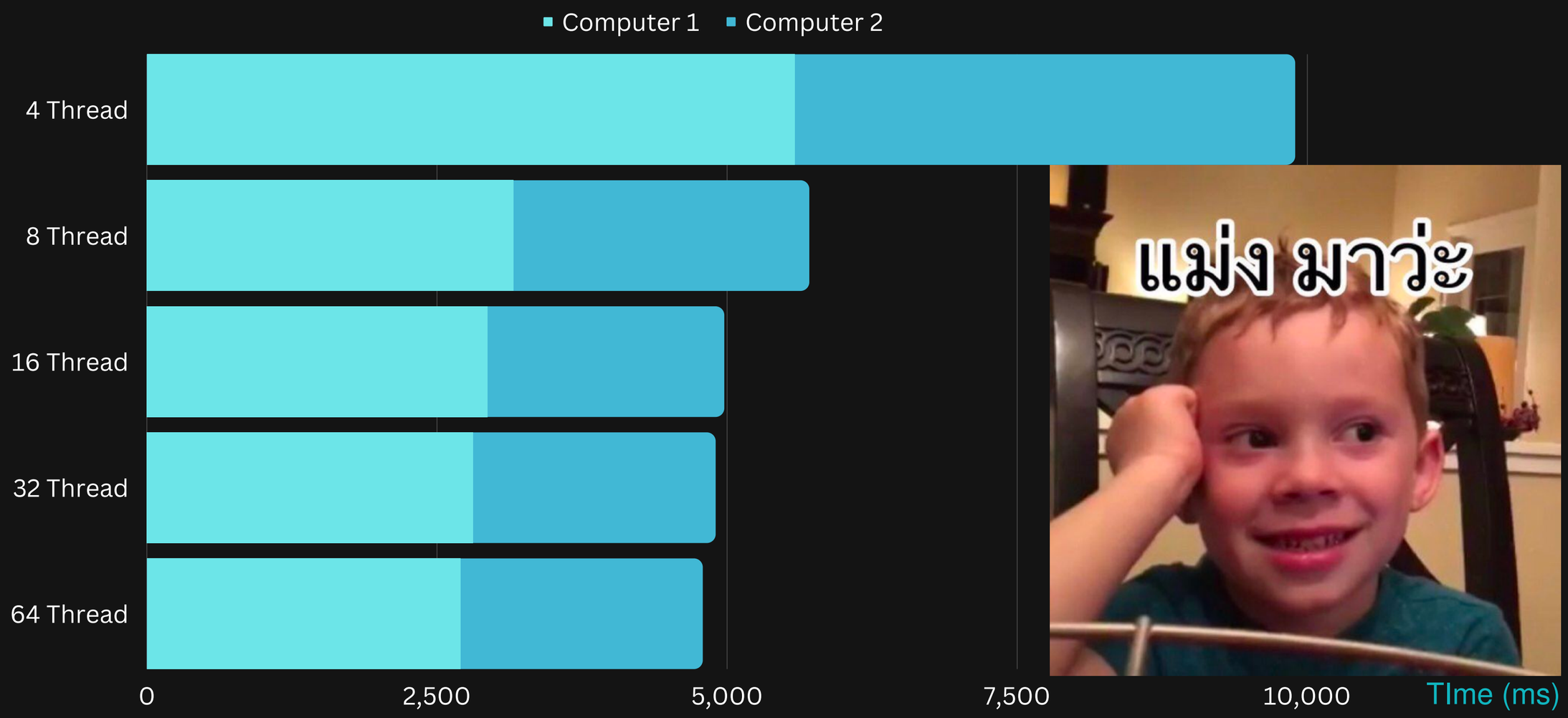
```
Working...Done.
Number of Threads: 16
Summation result: 888701676
Time used: 2005ms
```

```
Working...Done.
Number of Threads: 8
Summation result: 888701676
Time used: 2457ms
```

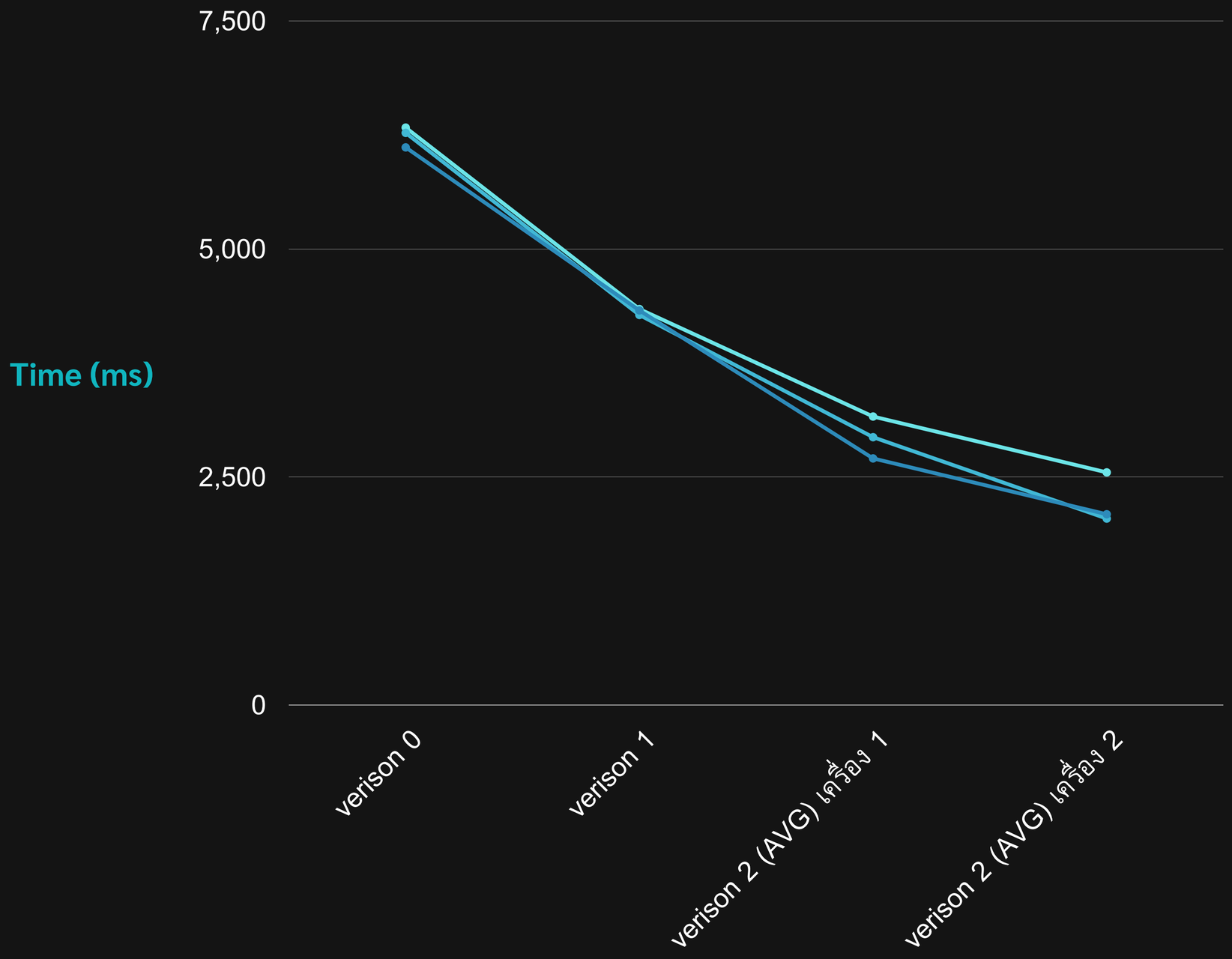
```
Working...Done.
Number of Threads: 32
Summation result: 888701676
Time used: 2106ms
```

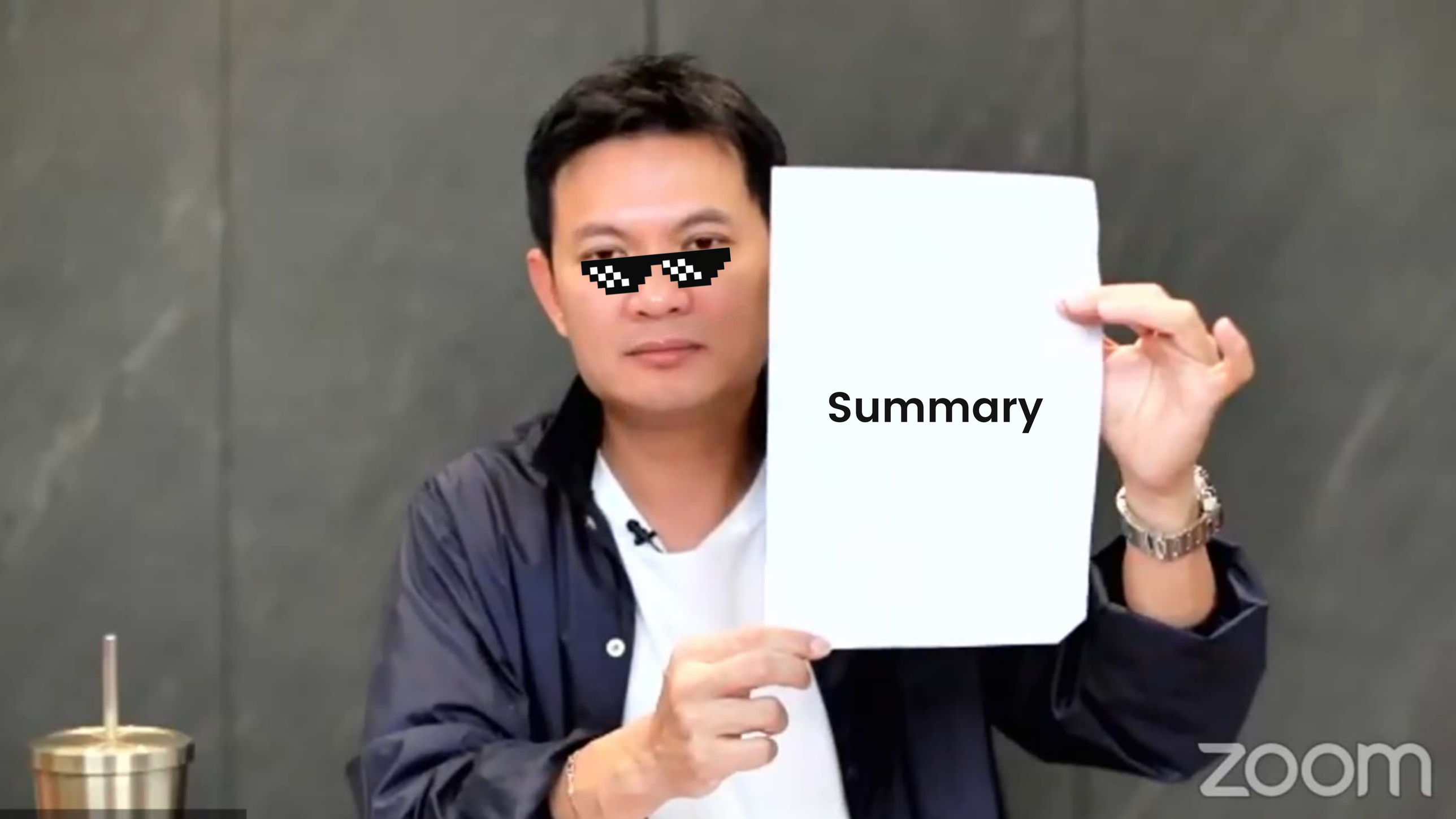
```
Working...Done.
Number of Threads: 64
Summation result: 888701676
Time used: 1984ms
```

Version 2


Computer 1
Computer 2
4 Thread
8 Thread
16 Thread
32 Thread
64 Thread
0
2,500
5,000
7,500
10,000
TIme (ms)

แม่ง มาว่ะ


7,500
5,000
2,500
0
Time (ms)
verison 0
verison 1
verison 2 (AVG) เครื่อง 1
verison 2 (AVG) เครื่อง 2


Summary
zoom


CENSORED


ขี้ยู!