**Input:**

1. An integer N
2. Another integer T where N is divisible by T
3. N integer numbers

**Problem:** Suppose you an array of N integers. Create T threads to calculate the sum of N subarrays.

For example, if N is 12, T is 3 and the integers are 1, 1, 2, 6, 3, 5, 1, 9, 4, 7, 2, 3.

Then thread 1 will calculate the sum of 1,1,2,6

Then thread 2 will calculate the sum of 3,5,1,9

Then thread 3 will calculate the sum of 4,7,2,3

Finally, total sum will be calculated with a synchronized method (in a class named SyncSum) where each thread will pass its local sum as argument.

Pseudocode for SyncSum

***Global Variable Sum = 0***

***SyncSum (Sub\_Sum, thread\_id)***

***{***

***Print(Sum)***

***Print(Sub\_Sum, thread\_id)***

***Sum = Sum + SubSum***

***Sleep(1000)***

***Print(Sum)***

***}***

Also insert sleep in thread’s run function while calculating sum of the sub array and print sub\_sum of each thread after each operation.