**CSE467 - Parallel and Distributed Computing**

**Assignment 4**

**Hiba Mallick - 24015**

**Objective: Write a program to sum N numbers using multi-threading**

The complexity to calculate sum of N data elements is O(N) and the execution time depends on the elements in the array. In this assignment you will learn how to pass and return values from a thread and how logically work can be divided into multiple threads to be performed in parallel and reduce total execution time. You need do device some logic for load balancing.

**Submit a word file containing all code, output snapshots and the comparison table and graph.**

**---------------------------------------------------------------------------------------------**

**Input:**

1. Array Length: **N** (to be used to create and initialize list)
2. No of Threads: **T**

**Processing:** Determine the execution time of addition with various values of T and N given in the table.

T0 = get clock cycle/system time

Perform Sum = ∑Di with load balancing among T threads.

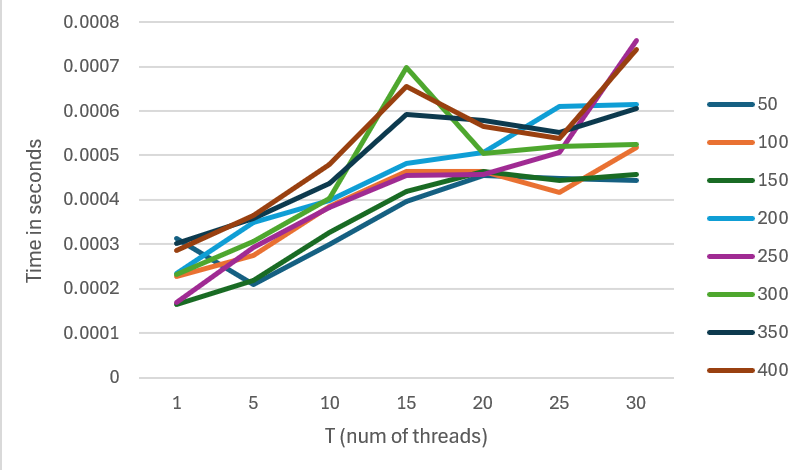
T1 = get clock cycle/system time

Time elapsed = T1 – T0

**Output:** Time elapsed - Tables show the comparison

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | N (number of data elements in array in thousands) | | | | | | | |
|  |  | 50 | 100 | 150 | 200 | 250 | 300 | 350 | 400 |
| T (no of threads) | 1 | 0.000314 | 0.000228 | 0.000164 | 0.000234 | 0.000169 | 0.000232 | 0.000302 | 0.000285 |
| 5 | 0.000209 | 0.000276 | 0.000218 | 0.000349 | 0.000293 | 0.000307 | 0.000358 | 0.000366 |
| 10 | 0.000300 | 0.000385 | 0.000326 | 0.000399 | 0.000382 | 0.000404 | 0.000437 | 0.000479 |
| 15 | 0.000397 | 0.000464 | 0.000418 | 0.000481 | 0.000455 | 0.000698 | 0.000592 | 0.000656 |
| 20 | 0.000455 | 0.000465 | 0.000465 | 0.000507 | 0.000457 | 0.000504 | 0.000579 | 0.000565 |
| 25 | 0.000448 | 0.000417 | 0.000443 | 0.000610 | 0.000506 | 0.000521 | 0.000551 | 0.000538 |
| 30 | 0.000443 | 0.000518 | 0.000458 | 0.000614 | 0.000758 | 0.000525 | 0.000605 | 0.000738 |

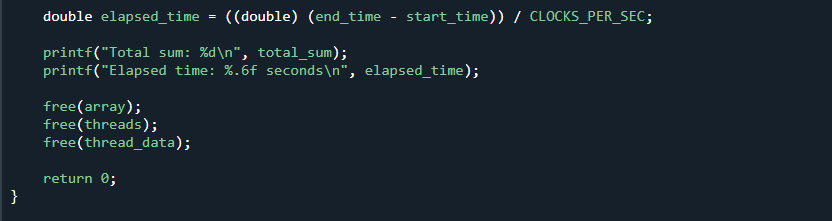
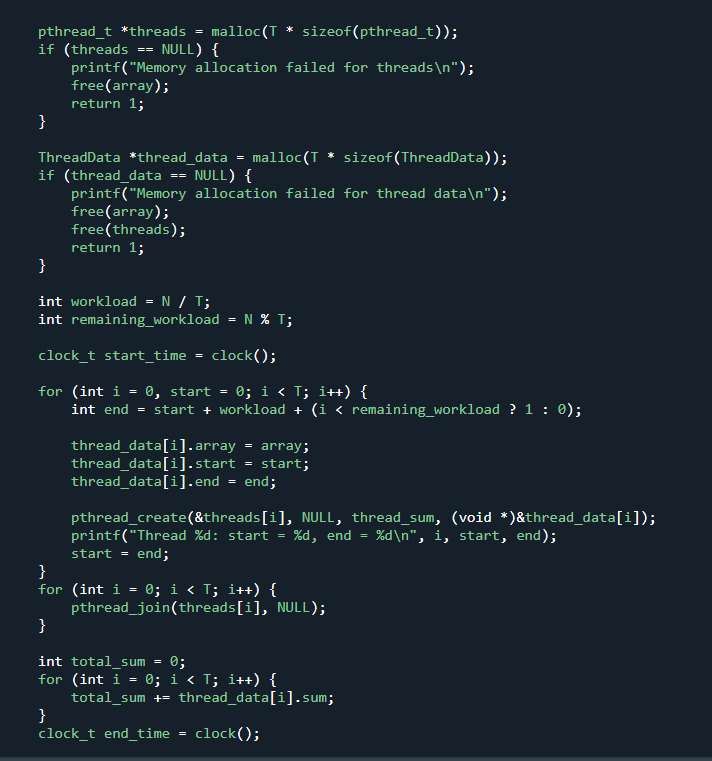
**Draw the Graph for above table.**



**Code:**

**A computer screen shot of a program code

Description automatically generated**

****

**Code Output:**

