**Distributed Systems**

**Assignment 3**

Message Passing Interface (MPI)

Using OPENMPI implementation

Mohamed AbdelAzeem Ahmedy

محمد عبد العظيم احمدي ابراهيم

1701235

# Code description:

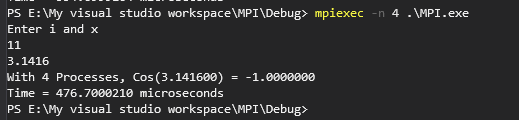
Code steps:

1. Initialize MPI.
2. Calculate world size and number of processes using:
   1. MPI\_Comm\_rank(MPI\_COMM\_WORLD, &rank);
   2. MPI\_Comm\_size(MPI\_COMM\_WORLD, &size);
3. Get Input from user using Get\_Input() Function.
4. Send all inputs form process 0 to all other processes.
5. Check input is valid. Number of iterations must be divisible by number of processes to divide load equally.
6. Divide iterations for all processes equally.
7. Start calculating start time at process 0
8. Each process has its interval to calculate its iterations by calling:
   1. proccess\_result = Calculate\_Result(Interval\_start, Interval\_End, x);
9. Each process send its result to process 0 using:
   1. MPI\_Send(&proccess\_result, 1, MPI\_DOUBLE, 0, 0, MPI\_COMM\_WORLD);
10. Process 0 calculate its result and receive all results from other processes and add all the results.
11. Process 0 calculate the end time.
12. Process 0 prints the final result.
13. Process 0 prints time calculated.

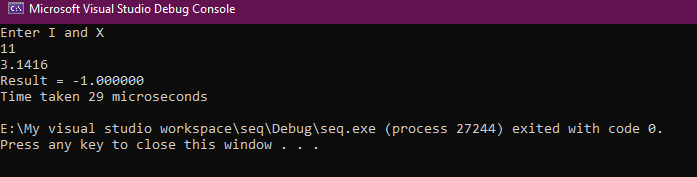
# Screen shots:

## Example 1:

MPI version:

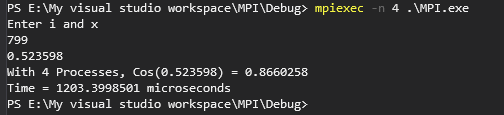


Sequential Version:

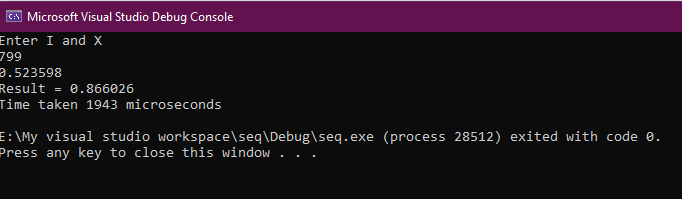


## Example 2:

MPI version:

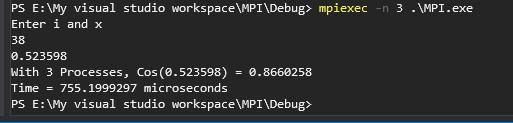


Sequential version:

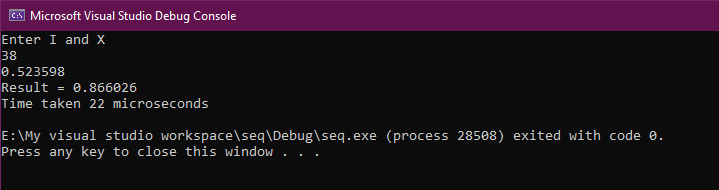


## Example 3:

MPI version:

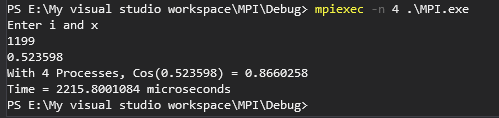


Sequential version:

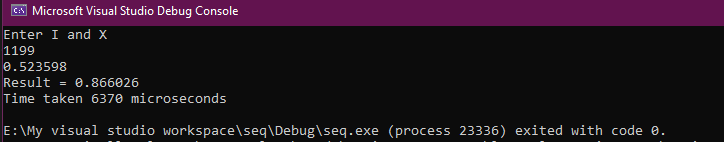


## Example 4:

MPI version



Sequential version:



# Why MPI solution takes more CPU time than Sequential version?

As in Example 2 and Example 4, MPI version is much faster than Sequential version because of large number of iterations. MPI adds overhead due to the parallelism and message passing. This overhead cannot be negligible if number of iterations is small. If number of iterations is large and take much time, MPI overhead can be negligible. Another reason why MPI version could give us larger time is when number of processes is more than number of CPU cores, Operating system runs all processes concurrently and do context switching. That’s why overhead is added to the calculations. To avoid Context switching overhead, we should assign number of processes to be as same as the number of CPU cores.