Ain Shams University
Faculty of Engineering
Computer and Systems Engineering Department
CSE411: Distributed Computer Systems

Assignment 3

Submitted by:

Mariam mohamed reda ahmed mohamed
ID: 1701405

Table of Contents

Table of figures

introduction

This report shows how the different nodes communicate with each other, via message passing, by giving example, which is using multiple processes to calculate cos(x) value, using the MPI (**Message Passing Interface**) library. Message Passing Interface is a standardized and portable message-passing standard designed to function on parallel computing architectures

Here is some of the main functions in MPI (C Function Call):

1- `int` **`MPI_Init`**`(int *argc, char **argv)` used to Initialize MPI
2- `int` **`MPI_Comm_size`**`(MPI_Comm comm, int *size)` used to Determine number of processes within a communicator
3- `int` **`MPI_Comm_rank`**`(MPI_Comm comm, int *rank)` used to Determine processor rank within a communicator
4- `int` **`MPI_Finalize`**`()` used to Exit MPI (must be called last by all processors)
5- `int` **`MPI_Send`** `(void *buf,int count, MPI_Datatype datatype, int dest, int tag, MPI_Comm comm)` used to Send a message
6- `int` **`MPI_Recv`** `(void *buf,int count, MPI_Datatype datatype, int source, int tag, MPI_Comm comm, MPI_Status *status)` used to Receive a message

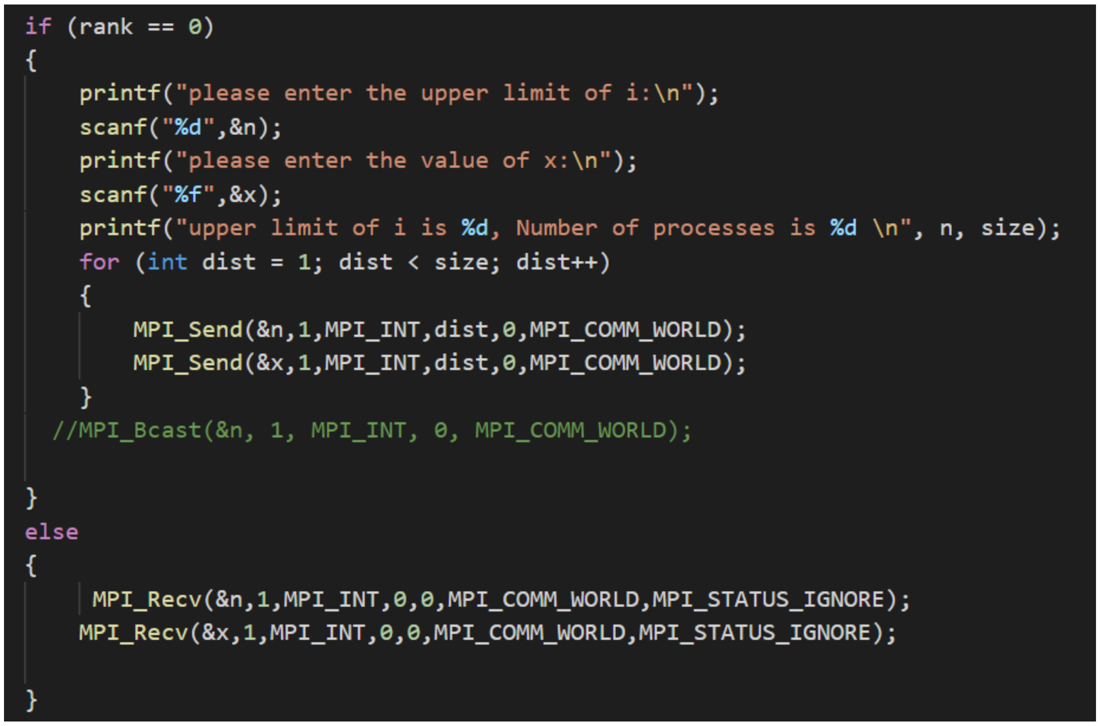
The Program Idea

We need to write a C program that uses MPI parallelization to compute the value of cos(x) using this formula $cos(x) = \sum_{k=0}^{\infty} \frac{(-1)^k x^{2k}}{(2k)!}$.

By getting the upper value of i and the value of x from the user, where the program makes the computation by dividing i equally among the processes, it should use n processes to do this computation, where n is provided as input by user. Then, it displays the computed value of cos(x), time taken by the program to compute it.

description of the solution

1- By making the process with rank = 0 take the inputs ang sending theme to the other processes as shown in figure 1



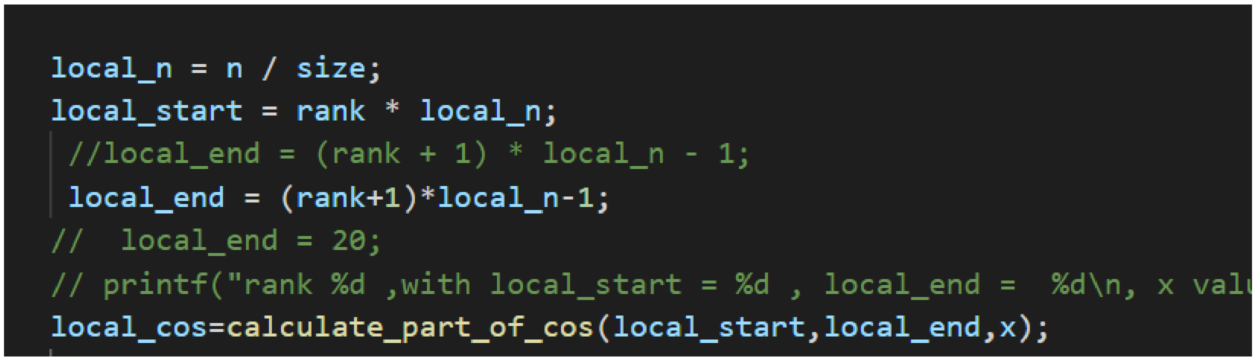
```
if (rank == 0)
{
    printf("please enter the upper limit of i:\n");
    scanf("%d",&n);
    printf("please enter the value of x:\n");
    scanf("%f",&x);
    printf("upper limit of i is %d, Number of processes is %d \n", n, size);
    for (int dist = 1; dist < size; dist++)
    {
        MPI_Send(&n,1,MPI_INT,dist,0,MPI_COMM_WORLD);
        MPI_Send(&x,1,MPI_INT,dist,0,MPI_COMM_WORLD);
    }
  //MPI_Bcast(&n, 1, MPI_INT, 0, MPI_COMM_WORLD);

}
else
{
     MPI_Recv(&n,1,MPI_INT,0,0,MPI_COMM_WORLD,MPI_STATUS_IGNORE);
    MPI_Recv(&x,1,MPI_INT,0,0,MPI_COMM_WORLD,MPI_STATUS_IGNORE);

}
```

Figure 1 taking inputs from user

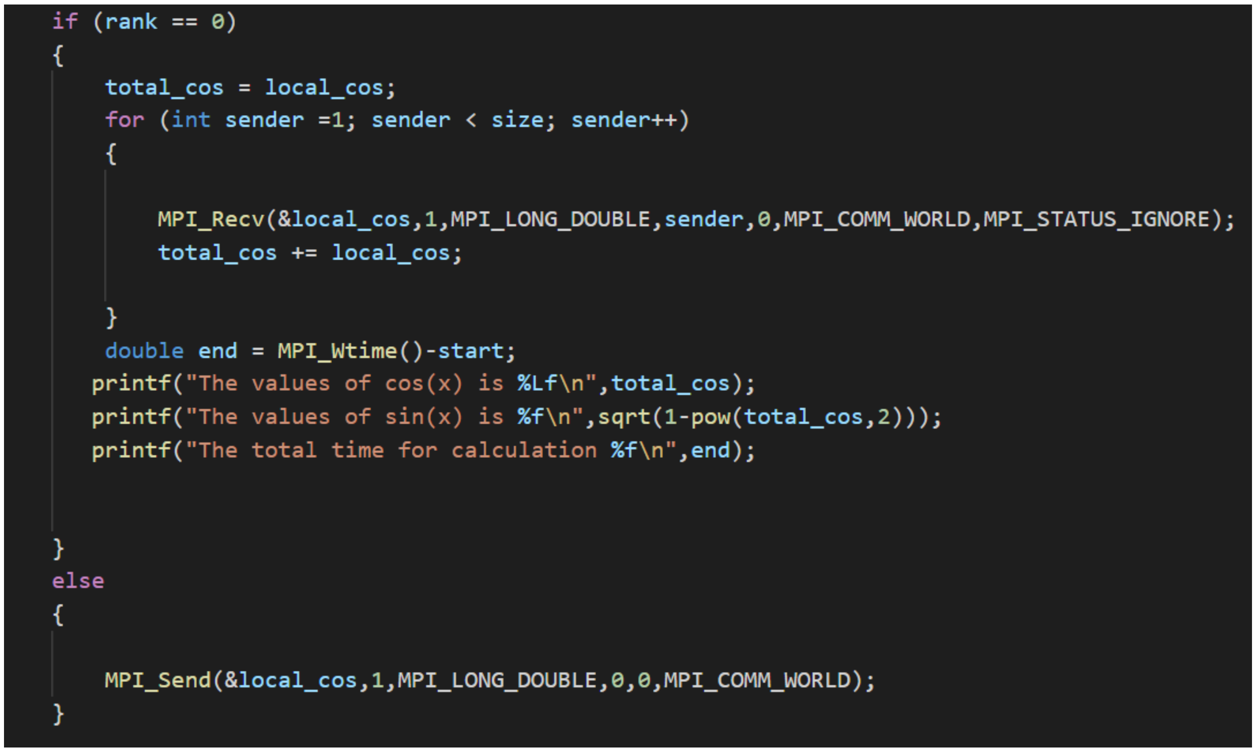
2- Calculate the local start and local end for each process and make each process calculate it's part of cos function



```
local_n = n / size;
local_start = rank * local_n;
 //local_end = (rank + 1) * local_n - 1;
 local_end = (rank+1)*local_n-1;
//  local_end = 20;
// printf("rank %d ,with local_start = %d , local_end =  %d\n, x valu
local_cos=calculate_part_of_cos(local_start,local_end,x);
```

Figure 2:local start and local end

3- Make each process send it's result to the process with rank =0 So this process can calculate the sum



```
if (rank == 0)
{
    total_cos = local_cos;
    for (int sender =1; sender < size; sender++)
    {

        MPI_Recv(&local_cos,1,MPI_LONG_DOUBLE,sender,0,MPI_COMM_WORLD,MPI_STATUS_IGNORE);
        total_cos += local_cos;

    }
    double end = MPI_Wtime()-start;
   printf("The values of cos(x) is %Lf\n",total_cos);
   printf("The values of sin(x) is %f\n",sqrt(1-pow(total_cos,2)));
   printf("The total time for calculation %f\n",end);


}
else
{

    MPI_Send(&local_cos,1,MPI_LONG_DOUBLE,0,0,MPI_COMM_WORLD);
}
```

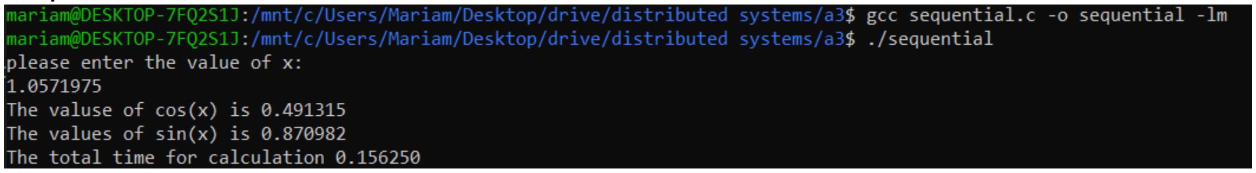
Figure 3:last calculation

The Results, and Difference in Performance

compare the time for parallel version to sequential version

with upper value of I = 5000 and x= 1.0471975.

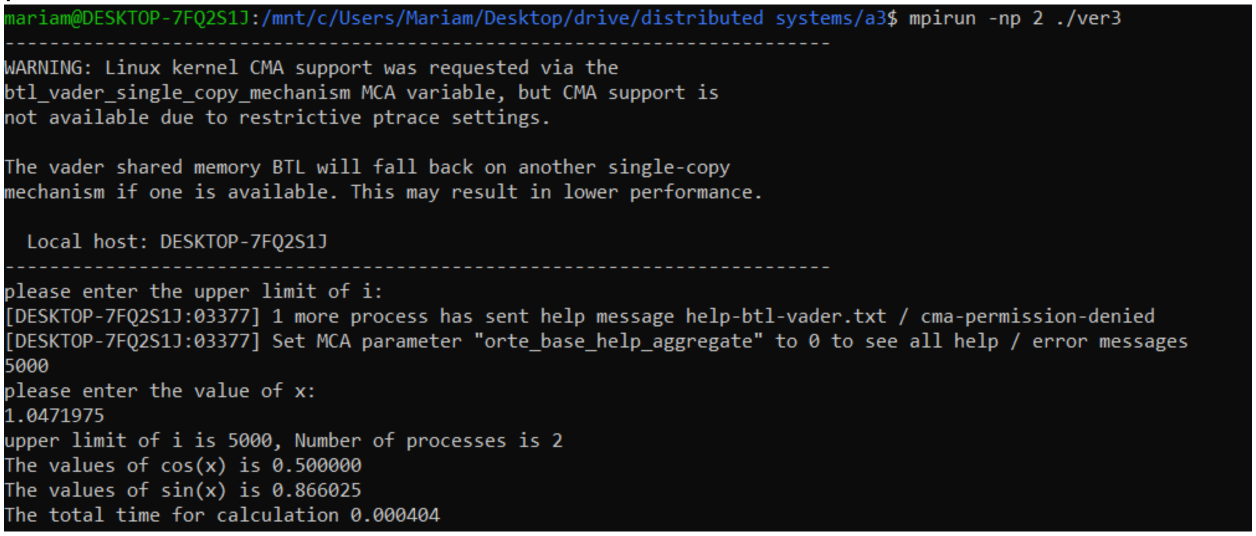
sequential version:



```
mariam@DESKTOP-7FQ2S1J:/mnt/c/Users/Mariam/Desktop/drive/distributed systems/a3$ gcc sequential.c -o sequential -lm
mariam@DESKTOP-7FQ2S1J:/mnt/c/Users/Mariam/Desktop/drive/distributed systems/a3$ ./sequential
please enter the value of x:
1.0571975
The valuse of cos(x) is 0.491315
The values of sin(x) is 0.870982
The total time for calculation 0.156250
```

Figure 4:sequential version with upper limit of i = 5000

parallel version:



```
mariam@DESKTOP-7FQ2S1J:/mnt/c/Users/Mariam/Desktop/drive/distributed systems/a3$ mpirun -np 2 ./ver3
--------------------------------------------------------------------------
WARNING: Linux kernel CMA support was requested via the
btl_vader_single_copy_mechanism MCA variable, but CMA support is
not available due to restrictive ptrace settings.

The vader shared memory BTL will fall back on another single-copy
mechanism if one is available. This may result in lower performance.

  Local host: DESKTOP-7FQ2S1J
--------------------------------------------------------------------------
please enter the upper limit of i:
[DESKTOP-7FQ2S1J:03377] 1 more process has sent help message help-btl-vader.txt / cma-permission-denied
[DESKTOP-7FQ2S1J:03377] Set MCA parameter "orte_base_help_aggregate" to 0 to see all help / error messages
5000
please enter the value of x:
1.0471975
upper limit of i is 5000, Number of processes is 2
The values of cos(x) is 0.500000
The values of sin(x) is 0.866025
The total time for calculation 0.000404
```

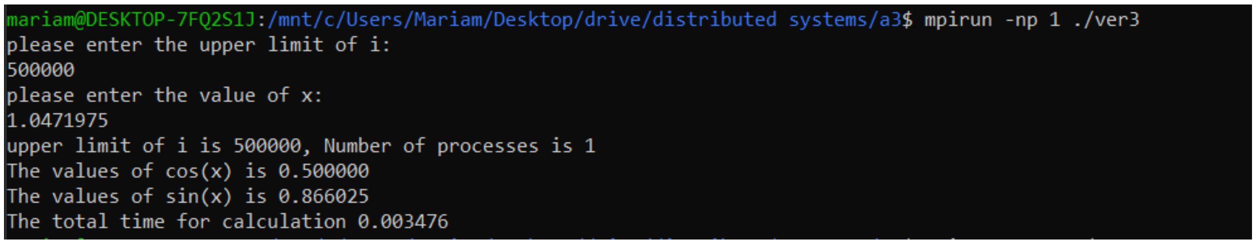
Figure 5:parallel version with upper limit of i = 5000

Time of parallel version is less than time of sequential version as shown.

number of computing processes changes.

discuss the differences in these times as the number of computing processes changes.

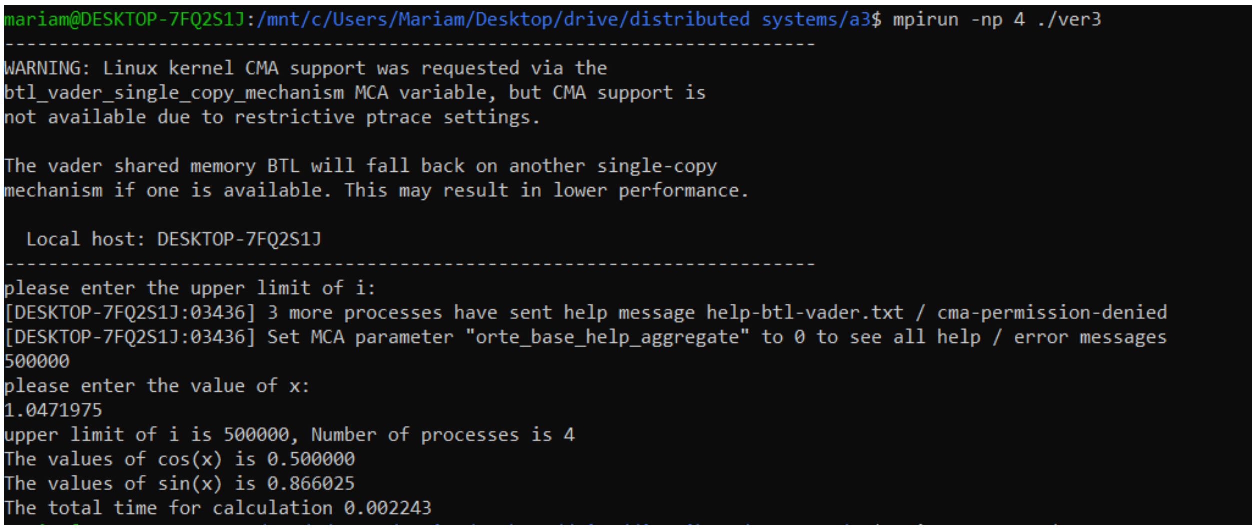
Number of processes = 1 with I = 500000 and x= 1.0471975



```
mariam@DESKTOP-7FQ2S1J:/mnt/c/Users/Mariam/Desktop/drive/distributed systems/a3$ mpirun -np 1 ./ver3
please enter the upper limit of i:
500000
please enter the value of x:
1.0471975
upper limit of i is 500000, Number of processes is 1
The values of cos(x) is 0.500000
The values of sin(x) is 0.866025
The total time for calculation 0.003476
```

Figure 6:parallel version with number of processes = 1

Number of processes = 4 with I = 500000 and x= 1.0471975



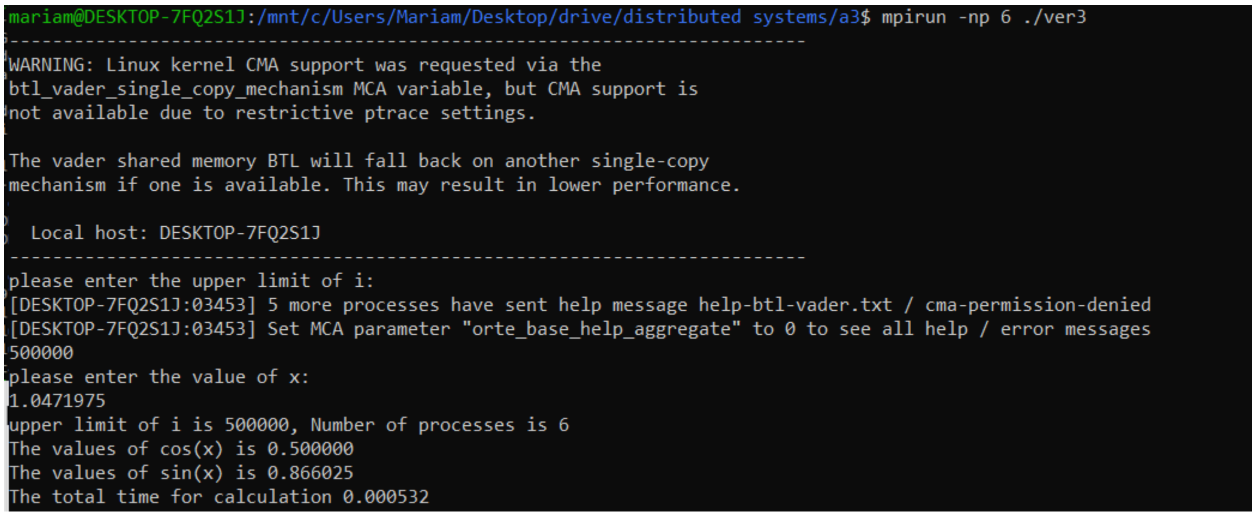
```
mariam@DESKTOP-7FQ2S1J:/mnt/c/Users/Mariam/Desktop/drive/distributed systems/a3$ mpirun -np 4 ./ver3
--------------------------------------------------------------------------
WARNING: Linux kernel CMA support was requested via the
btl_vader_single_copy_mechanism MCA variable, but CMA support is
not available due to restrictive ptrace settings.

The vader shared memory BTL will fall back on another single-copy
mechanism if one is available. This may result in lower performance.

  Local host: DESKTOP-7FQ2S1J
--------------------------------------------------------------------------
please enter the upper limit of i:
[DESKTOP-7FQ2S1J:03436] 3 more processes have sent help message help-btl-vader.txt / cma-permission-denied
[DESKTOP-7FQ2S1J:03436] Set MCA parameter "orte_base_help_aggregate" to 0 to see all help / error messages
500000
please enter the value of x:
1.0471975
upper limit of i is 500000, Number of processes is 4
The values of cos(x) is 0.500000
The values of sin(x) is 0.866025
The total time for calculation 0.002243
```

Figure 7:parallel version with number of processes = 4

Number of processes = 6 with I = 500000 and x= 1.0471975



```
mariam@DESKTOP-7FQ2S1J:/mnt/c/Users/Mariam/Desktop/drive/distributed systems/a3$ mpirun -np 6 ./ver3
--------------------------------------------------------------------------
WARNING: Linux kernel CMA support was requested via the
btl_vader_single_copy_mechanism MCA variable, but CMA support is
not available due to restrictive ptrace settings.

The vader shared memory BTL will fall back on another single-copy
mechanism if one is available. This may result in lower performance.

  Local host: DESKTOP-7FQ2S1J
--------------------------------------------------------------------------
please enter the upper limit of i:
[DESKTOP-7FQ2S1J:03453] 5 more processes have sent help message help-btl-vader.txt / cma-permission-denied
[DESKTOP-7FQ2S1J:03453] Set MCA parameter "orte_base_help_aggregate" to 0 to see all help / error messages
500000
please enter the value of x:
1.0471975
upper limit of i is 500000, Number of processes is 6
The values of cos(x) is 0.500000
The values of sin(x) is 0.866025
The total time for calculation 0.000532
```

Figure 8:parallel version with number of processes = 6

As shown, by increasing the number of processes the time required for calculation decrease while all resulting in the same answer, which is expected.