PDC (Practice Lab Assignment)

Question 1 is in continuation with previous assignment of OpenMP

1. (**Matrix Multiplication**) Write **OpenMP** parallel program for matrix multiplication

   - Compare the performance of serial and parallel version of matrix multiplication.

MPI Programming

2. Write MPI hello world program. This exercise will help you to be familiar with the basic commands (compile and run) and routines of MPI programming model.

3. Study about following basic MPI subroutines

Function Purpose	C Function Call
Initialize MPI	`int MPI_Init(int *argc, char **argv)`
Determine number of processes within a communicator	`int MPI_Comm_size(MPI_Comm comm, int *size)`
Determine processor rank within a communicator	`int MPI_Comm_rank(MPI_Comm comm, int *rank)`
Exit MPI (must be called last by all processors)	`int MPI_Finalize()`
Send a message	`int MPI_Send (void *buf,int count, MPI_Datatype datatype, int dest, int tag, MPI_Comm comm)`
Receive a message	`int MPI_Recv (void *buf,int count, MPI_Datatype datatype, int source, int tag, MPI_Comm comm, MPI_Status *status)`

4. You need to use MPI to parallelize the serial program `integrate.c (given below)`, which integrates function sin(X) over the range from 0 to pi using N intervals, where N is an argument of the program.

```c
#include <stdio.h>
#include <math.h>

#define PI 3.1415926535

int main(int argc, char **argv)
{
  long long i, num_intervals;
  double rect_width, area, sum, x_middle;

  sscanf(argv[1],"%llu",&num_intervals);

  rect_width = PI / num_intervals;

  sum = 0;
  for(i = 1; i < num_intervals + 1; i++) {

    /* find the middle of the interval on the X-axis. */

    x_middle = (i - 0.5) * rect_width;
```

```c
    area = sin(x_middle) * rect_width;
    sum = sum + area;
  }

  printf("The total area is: %f\n", (float)sum);

  return 0;
}
```