1. Write A MPI program that should determine the partner process and then send or receive a message (your number or a name) with it.

Use non-blocking send/recv routines

1. Observe the difference between the blocking and non-blocking communication.
2. WAP a C program to calculate the value of pi in parallel programming. Use dart board algorithm.
3. WAP in MPI that prints your name only if number of processes is even, otherwise should return even process.
4. WAP a C program that should initialise Matrix A and B. The program should multiply these matrices and store the result in matrix C.
5. WAP to calculate the value of Pi in MPI.
6. Write a MPI program that should take data(may be name or number) and send all the data to all the processes in the communicator. Also print the information.
7. Write a MPI program that should return SUM of all processes involved.

(all processes should be involved)

1. Write an MPI program such that it should initialise an array having values 1-25. The process should divide the values among 5 processes equally.
2. WAP in MPI for simple data decomposition such that master task should first initialise an array and then distribute an array to other tasks. After that the other tasks should receive an ADDITION operation in elements of array.

Email them to omjadhav@cdac.in