

## Assignment for point to point and Collective communications in MPI

1.	Write a program in MPI to print total number of process and rank of each process.
2.	Write a simple C++ program to multiply two matrices of size MxN and NxP.
3.	Write a program in MPI where even ranked process prints Hello and odd ranked process prints World.
4.	Write a program in MPI to simulate simple calculator. Perform each operation using different process in parallel.
5.	Write a program in C++ to count the words in a file and sort it in descending order of frequency of words i.e. highest occurring word must come first and least occurring word must come last.
6.	Write a MPI program to find the prime numbers between 1 and 100 using two processes.
7.	Write a MPI program using standard send. The sender process sends a number to the receiver. The second process receives the number and prints it.
8.	Write a MPI program using synchronous send. The sender process sends a word to the receiver. The second process receives the word, toggles each letter of the word and sends it back to the first process. Both process use synchronous send operations.
9.	Write a MPI program where the master process (process 0) sends a number to each of the slaves and the slave processes receives the number and prints it. Use standard send.
10.	Write a MPI program to add an array of size N using two processes . Print the result in the root process. Investigate the amount of time taken by each process.
11.	Write a MPI program to read N elements of the array in the root process (process 0) where N is equal to the total number of process. The root process sends one value to each of the slaves. Let even ranked process find square of the received element and odd ranked process find cube of received element. Use Buffered send.
12.	Write a MPI program to read an integer value in the root process. Root process sends this value to process1, Process1 sends this value to Process2 and so on. Last process sends the value back to root process. When sending the value each process will first increment the received value by one. Write the program using point to point communication routines.
13.	Write a MPI program to read N elements of an array in the root. Search a number in this array using root and another process. Print the result in the root.

14.	Write a MPI program to read N elements of an array in the master process. Let N process including master process check the array values are prime or not.
15.	Write a MPI program to read value of N in the root process. Using N processes including root find out $1!+(1+2)+3!+(1+2+3+4)+5!+(1+2+3+4+5+6)+\dots+n!$ or $(1+2+\dots+n)$ depending on whether n is odd or even and print the result in the root process.
16.	Write a MPI program to read N values of the array in the root process. Distribute these N values among N processes. Every process finds the square of the value it received. Let every process return these value to the root and root process gathers and prints the result. Use collective communication routines.
17.	Write a MPI program to read N values in the root process. Root process sends one value to each process. Every process receives it prints the factorial of that number. Use N number of processes.
18.	Modify the above program such that every process returns the factorial to root process. Root process gathers the factorial and finds sum of it.
19.	Write a MPI program to read a value M and NxM elements in the root process. Root process sends M elements to each process. Each process finds average of M elements it received and sends these average values to root. Root collects all the values and finds the total average. Use collective communication routines. Use N number of processes.
20	Write a MPI Program to read two strings S1 and S2 of same length in the root process. Using N process including the root (string length is evenly divisible by N), produce the concatenated resultant string as shown below. Display the resultant string in the root process. Write the program using Collective communication routines.  Eg:      String S1: string   String S2: length   Resultant String : slternigtgh
21	Write a MPI Program to read a string of length M in the root process. Using N processes (N evenly divides M) including the root toggle the characters and find the ASCII values of these toggled characters. Display the toggled characters and ASCII values in the root process.
22	Write a program to read a value M and NxM number of elements in the root. Using N processes do the following task. Find the square of first M numbers, Find the cube of next M numbers and so on. Print the results in the root.