


National University of Computer and Emerging Sciences, Lahore Campus

	Course Name:	Parallel and Distributed Computing	Course Code:	CS3006
	Program:	BS(Data Science)	Semester:	Spring 2024
	Sections	B	Total Marks:	65
	Due Date:		Weight	
	Exam Type:	Assignment 3	CLO	3

Student : Name: _____ Roll No. _____ Section: _____

Question 1 [20]

In this part you have to write an MPI program for the question given in assignment

2. Consider a 4*4 matrix with the following values:

1 3 5 7

2 4 6 8

11 13 15 17

12 14 16 18

Assume that we are in distributed environment with four processes and each row is initially stored at a different process, row 1 is stored at process P0, row 2 at P1, row 3 at P2, and row 4 at P3. The requirement is to do a matrix transpose so at the end, each one of them has a row after transpose operation. Write your program to use proper network operators to achieve the functionality.

Question 2 [20]

In this part you have to convert your multithreaded program from “assignment 1” into a remote program that has a method called sortArray which can be called using remote procedure call. Other than this, there are two more procedures: one to find minimum value in an array and other to find maximum value. The clients on the same or the remote machines can call these functions.

The flow of program will be as follows:

1. Your client program will take a matrix of size (m x n) where ‘m’ and ‘n’ values are taken as input from user. Initialize the matrix by some random values or user input.
2. Then make RPC calls equal to number of rows ‘m’ and each remote procedure call will sort a row on called server using Quick Sort in ascending order.

Test your program by sorting a matrix, finding minimum and maximum values.

Question 3 [25]

Write a distributed program using ‘MPI’ to perform matrix operations as instructed below. You can paste your code in this document. Teaching assistant can ask any student to run the code or he can conduct a viva to check the authenticity.

Part A:

- 1. Create a matrix of size (m x n) and initialize the matrix by some random values or user input in process 0.**
- 2. Distribute the rows using relevant communication operator to all processes.**
- 3. Each process then sort the assigned row using merge sort.**
- 4. Finally collect sorted rows from all processes to process 0.**
- 5. Each process should also calculate the product of each row which then can be reduced at process 0 to find the product of the complete matrix. You can use reduce function for this task.**