BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI RAJASTHAN

I SEMESTER 2023-2024
Assignment

Course No.:	**CC ZG501**	**Course Title:**	Introduction to Parallel & Distributed Programming
Deadline:	**As per Taxila**	**Maximum Marks:**	30M (30%)

Note: This is an individual assignment. Create two folders P1 and P2 and place all deliverables within them.

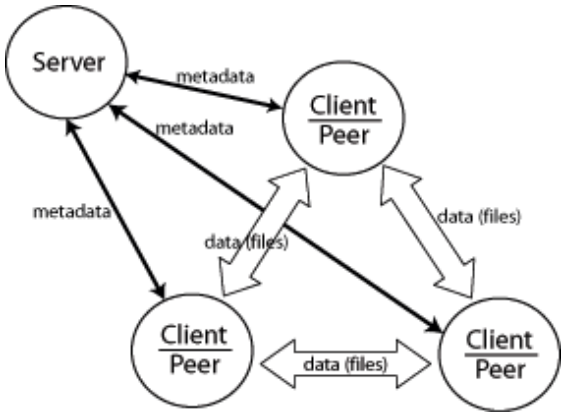
P1. Consider N processes. Each process *i* has two neighbhours *i-1* and *i+1*. Processes work in 500 iterations. In even iteration, a process *i* sends *i* to all processes. When a process receives a many numbers in even iteration, it will pickup the highest number and if that number equals its own, it will not participate in the subsequent iterations. Such processes are called neutral processes. During odd iterations, a process sends *i* to its i+1 neighbhour. Each process i subtracts one from the number it received from the previous process and sends to the next one i.e. i+1 process. Neutral processes will not subtract the number but simply send it to the next one. During odd iteration, when a process receives zero, that process will become netral process.

Deliverables:

(a) Taking each process as MPI process, implement the above using MPI API. Source code to be submitted.
(b) Document with details on the deisgn and how to execute the program
(c) Screen recording with audio explanation of your execution. If video recording is > 10MB, please share it in Google Drive and put the link in recording.txt.

[15]

P2. Consider the following diagram. This is the design of a Napster music file sharing system. The system has a network of machines. One of the machines is a server. Rest act as clients when they interact with server and peers when they interact with other machines in the network. The server receives very huge number of requests compared to peers. The system works as follows.



The server maintains a database (data structure) of client's ip, port number, and the list of files client is sharing. Client can send a request for a file to the server. Server searches for the requested file in the database and send the peer ip address and port number (where the peer can server the file) to the client. Client peer makes a connection to the specified peer and downloads the file.

Deliverables:

(a) Taking approach of Sockets API/RPC/message queues implement the above requirements in C/C++/Java/Python/Go. Source code to be submitted.
(b) Document with details on the deisgn and how to execute the program
(c) Screen recording with audio explanation of your execution. If video recording is > 10MB, please share it in Google Drive and put the link in recording.txt.

[15]

--&--