Team 23 Ayush Mittal(2021201030), Nitin Kumar(2021202020), Swarnali Dey(2021201088)

1. **What problem you are solving.**
   We will be Implementing the Ricart-Agarwala's mutual exclusion protocol (for Non-FIFO Channels). It enables multiple processes in a distributed system to coordinate exclusive access to a shared resource. This algorithm ensures that only one process can access the shared resource at a time and that requests are granted in the order they are received, thus preventing starvation.

   The Implementation should also handle
   a. **Handle node failures:** In case of node failure rest of the processes should get to know about the failure and accordingly adjust itself so that the algorithm works, even if few nodes have failed.
   b. **Record state of different nodes:** There should be a mechanism to figure out the current state of the processes, whether it is executing CS or not, whether it has failed or not.
   c. **Add new nodes to the system:** Whenever a new process comes up (become active) and may want to enter a critical section, our system should handle this case as well.

2. **Why are you solving it.**
   We are solving this problem to provide a mechanism to avoid conflicts, race conditions, lost updates, corrupted data and inconsistencies by providing mutual exclusive access to resources, such that it requires only 2(N − 1) messages per CS execution.

3. **What software you will write.**
   Each process which will follow Ricart-Agarwala's algorithm for accessing the critical algorithm in mutually exclusive manner will contain few threads for following tasks -
   a) For accepting the request from other processes and replying to them.
   b) One for sending requests to access critical sections to other processes.
   c) One for communicating with the Failure Detector and other for the process' own working.

   Failure detector is a health check like service which requests all the existing processes and waits for their reply, if a node does not send a reply back then after waiting for a specified amount of time (timeout) the node will be declared failed and the same will be notified to other processes.
   When a new node joins the system, it will be notified to all the previous processes and the new node will know information about all the already existing processes. For communicating we will we using socket programming and for accessing and updating files we can using ssh connection and sftp protocol .

4. **What the expected results will be.**
   An architecture where processes will be able to access the resources or should say critical section in mutually exclusive fashion using Ricart-Agarwala's algorithm. Along with this, the status of all the nodes will be maintained, new nodes will be allowed to be added into the system and if an existing node fails then also the system will know about it and the operation of the algorithm will remain consistent.