# Problem Statement

**Project Description**

In this programming project, you will develop an n- node distributed system that implements a vector clock. The distributed system uses a logical clock to timestamp messages sent/received among the nodes. To simplify the design and testing, the distributed system will be emulated using multiple processes on a single machine. Each process represents a machine and has a unique port number for communication. You can use any programming language.

Implement the vector clock for your distributed system. You can create two threads for each process, one for sending messages to other nodes and one for listening to its communication port. Communication among nodes can be done using RPC or using sockets. Once a process sends a message, it should print its vector clock before and after sending a message. Similarly, once a process receives a message, it should print its vector clock before and after receiving the message. You can assume that the number of processes (machines) is fixed (equal to or larger than 3) and processes will not fail, join, or leave the distributed system.

**Part-1(40 points)** Each process can send a unicast message to another process. Update the vector clocks of the processes involved and print their vector clocks before and after sending/receiving the messages.

**Part-2(40 points)** Each process can broadcast a message to all other processes. Update the vector clocks of all the processes and print the vectors of all the processes before and after sending/receiving the messages.

**Deliverables**

The deliverables include the source code of the programs, a README file (10 points) containing instructions on how to compile and run your programs, and a report (10 points) that briefly describes how you implemented the programs, what you have learned, and what issues you encountered. Put all the required documents into a zipped folder. Make sure you clearly list your names and student IDs in the report.