Edgar Chavez

Student ID: 1002091846

Project Report

**I have neither given nor received unauthorized assistance on this work.**

My implementation for this program involved creating a ‘VectorClock’ class that initializes the vector clock to the total number of processes, an increment function to update the initialized clock and the update function to update the clock depending on the other clock. Next the listener and sender thread functions were created. In the ‘listenerThread’ function it listens for any incoming messages after a message is received the function calls the update function and prints the before and after states of the clock. In the ‘senderThread’ function it sends a message through sockets either unicast or broadcast and calls the increment function. In the ‘runProcesses’ function this starts the listener and sender threads for the processes, initializes vector clock, creates/binds socket, and creates list of all the processes. In the main its created to take in three command line arguments: PID, port number, and total number of processes.

I’ve learned vector clocks are essential in distributed systems since they can help maintain consistency by tracking the order of events, coordination/synchronization since even if one processor received multiple messages another processor can receive a message and be in synch. They can also be used in debugging and monitoring the system. Some issues I ran into were just having little experience with socket programming and threading, and I had an issue were I would receive this error “OSError: [WinError 10048] Only one usage of each socket address (protocol/network address/port) is normally permitted”.