CSE 5306-004: DISTRIBUTED SYSTEMS

PROJECT 2 REPORT

TEAM MEMBERS:

Manideep Shanigaram (1002024224)

Abhishek Duggirala (1002031522)

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

Sign: Manideep Shanigaram, Abhishek Duggirala Date: 03/20/2023

Implementation:

We have implemented this project using java. We used 3 nodes in this project, running on different ports 8080, 8081 and 8082. We used sockets to establish connections between machines. We used Socket and ServerSocket methods to implement it. We implemented this project in two ways, one is when a node tries to send a message to other node and the second is when a node broadcast the message to all the remaining nodes. When a node sends a message to another node vector clock before and after sending will be displayed, at the receiver's end vector clock before and after receiving will be printed. In the same way when a node broadcasts the message to all the nodes vector clock is printed before and after sending the message at sender's side and also prints updated vector clock at all the receiver's side. To update the sender's vector clock, we are incrementing the vector clock of the specific node which is sending the message. To update the receiver's vector clock, we are updating the vector clock with the max value of receiver's and sender's for each node.

Learnings:

From this project we leant how to implement distributed systems using sockets. And how to use vector clocks in distributed systems. We learned how communication takes place between nodes.

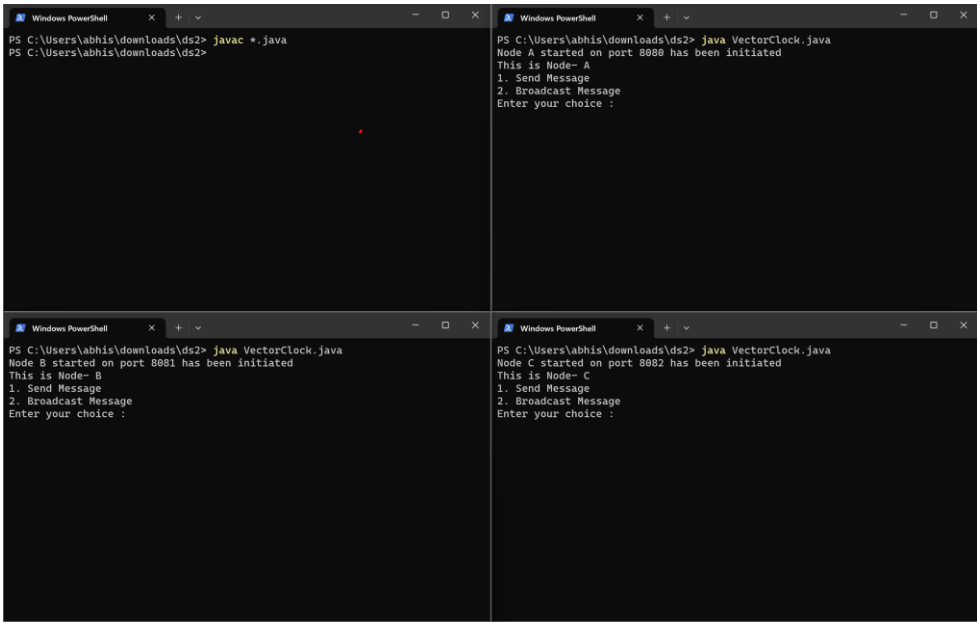
Issues:

- We faced issue while updating clock on receiver's end, then we used a tempClock to implement it.

- We faced issue in implementing the logic for broadcast, eventually we took a list of nodes and removed current node from it and sent messages to all the other nodes.

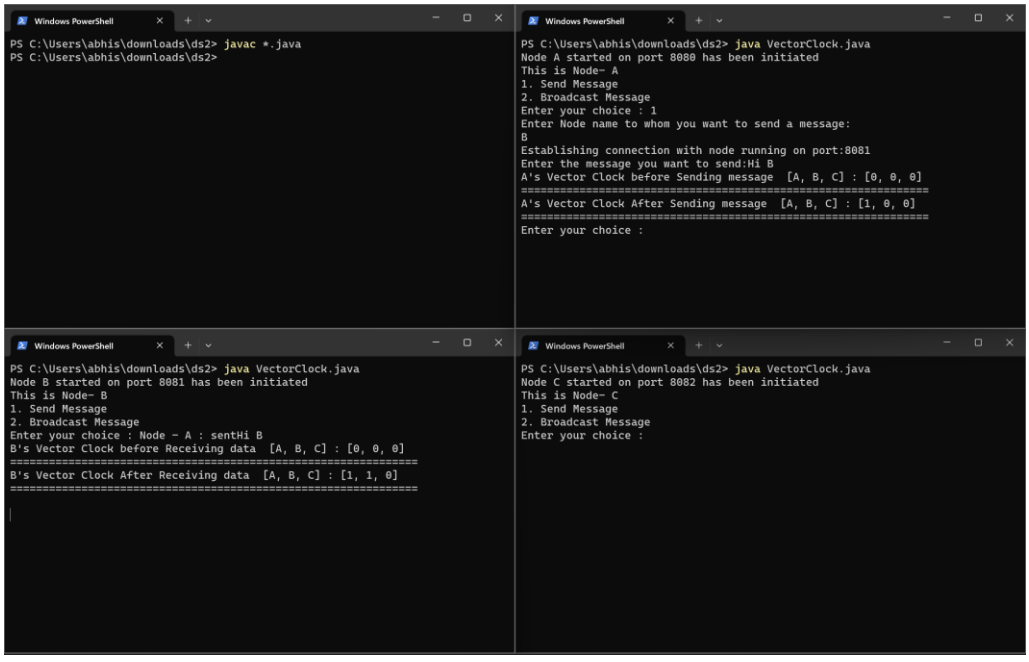
Output:

In below screenshot we can see that we have run the VectorClock file 3 times to initialize 3 nodes A, B and C.

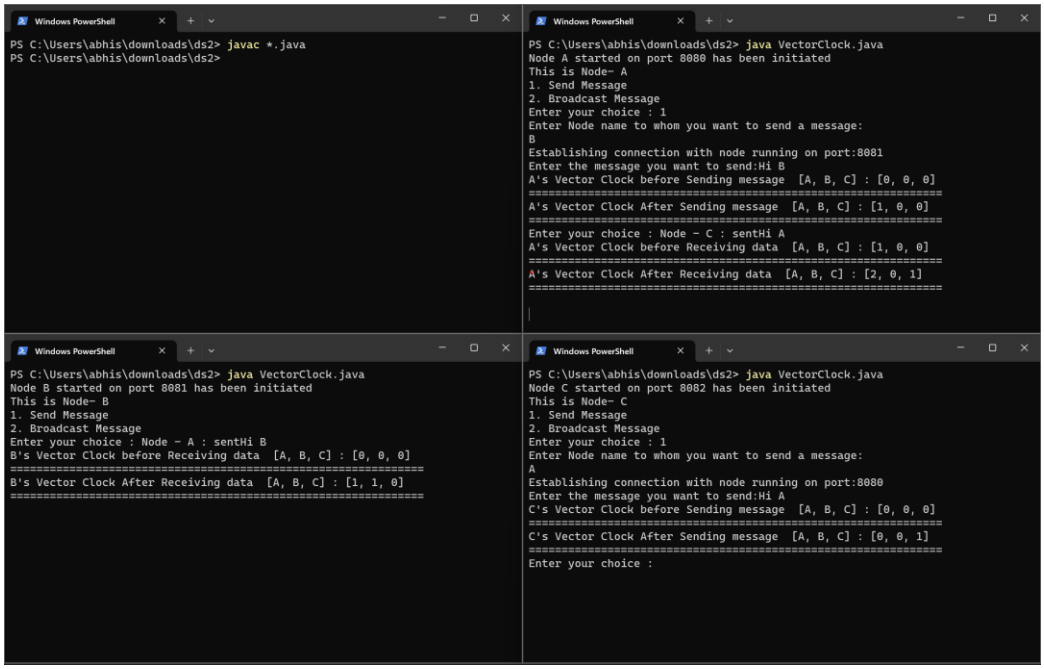


Unicast:

In below screenshot we can see that node A has sent a message to node B. Vector clocks of both sender and receiver are displayed before and after respectively.



In below screenshot we can see that node C has sent a message to node A. Vector clocks of both sender and receiver are updated and displayed before and after respectively.



Broadcast:

In below screenshot we have performed broadcast node A is broadcasting to all the remaining nodes and we can see the vector clocks displayed respectively.