**Project Report 2**

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

**Sign:  1001959376, 1002039215                                               Date: 10/06/2022**

**Project Overview:**

This project implements vector clock over 4 nodes. Socket programming has been implemented for the nodes to communicate with each other.

**Implementation:**

This project consists of 4 nodes: P0, P1, P2 and P3, each running a process which listen on ports 2020, 2021, 2022 and 2023 respectively. Each node behaves as a server as well as a client. Each node has 2 methods, send() and listen(), which send and receive messages. Each node has its own vector clock with values of other nodes maintained. So, every time a node sends or receives a message, the clock specific to that node is incremented my 1. (Refer to the flow diagram drawn below).

**<2,0,0,1>**

**<1,0,0,1>**

P0

**<0,0,0,0>**

**<0,0,0,0>**

**<0,0,0,0>**

**<0,0,0,0>**

P3

P1

P2

**RECEIVED**

**SENT**

**<2,2,0,1>**

**<2,1,0,1>**

**<2,2,2,1>**

**<2,2,1,1>**

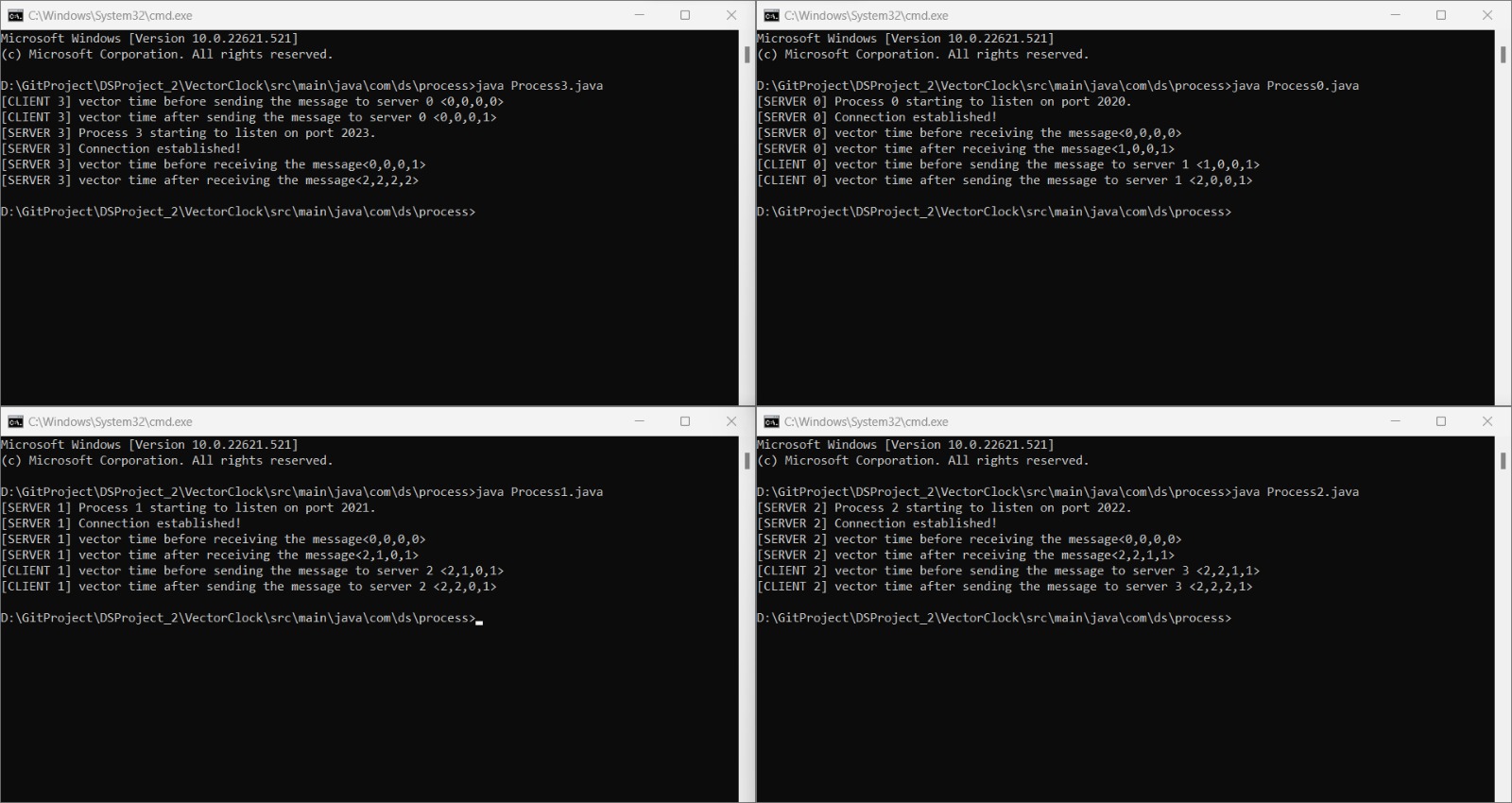
**<0,0,0,1>**

**<2,2,2,2>**

The above diagram clearly illustrates the current flow of the implementation which goes from:

**P3 -> P0 -> P1 -> P2 -> P3**

In the solution, we print the clock before and after sending or receiving a message. Each node has 2 threads, one for listening the messages and another for sending the messages. For communication between the nodes, we used socket programming. At any point the processes will not fail, join or leave the distributed system. The logs will be something like this:



**Learning:**

* Vector clocks
* Socket programming
* String manipulation
* Git

**Issues Encountered:**

* Structuring the nodes.
* Implementing vector clocks without using any libraries.
* Ensuring that a process does not sends a message to another process which is not listening yet.

**Labor Division:**

**Devyani:** Node structure, socket communication.

**Akanksha**: vector clocks

**References:**

[**https://www.geeksforgeeks.org/vector-clocks-in-distributed-systems/**](https://www.geeksforgeeks.org/vector-clocks-in-distributed-systems/)

[**https://en.wikipedia.org/wiki/Vector\_clock#:~:text=A%20vector%20clock%20is%20a,the%20sending%20process's%20logical%20clock**](https://en.wikipedia.org/wiki/Vector_clock#:~:text=A%20vector%20clock%20is%20a,the%20sending%20process's%20logical%20clock)**.**

[**https://www.tutorialspoint.com/How-to-convert-comma-seperated-java-string-to-an-array**](https://www.tutorialspoint.com/How-to-convert-comma-seperated-java-string-to-an-array)

**Prepared By:**

**Devyani Singh 1001959376**

**Akanksha Tomar 1002039215**