

18

# **National Institute of Technology Raipur**

## **Distributed Systems Lab File**

**By**

**Samarth Pillai**

**Roll No. 19115072**

**(7<sup>th</sup> Semester, Computer Science & Engineering)**



# Certificate

Name: SAMARTH PILLAI

Class: CSE, 7<sup>th</sup> sem

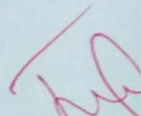
Roll No: 19115072

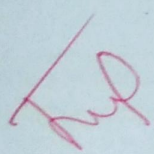
Exam No:

Institution NIT Raipur

This is certified to be the bonafide work of the student in the  
Distributed System Laboratory during the academic  
year 2022/2023.

No. of practicals certified 08 out of 08 in the  
subject of Distributed System Lab.

  
.....  
Teacher In-charge

  
.....  
Examiner's Signature

.....  
Principal

Date: 11/11/2022

Institution Rubber Stamp

(N.B: The candidate is expected to retain his/her journal till he/she passes in the subject.)



<b>S. no.</b>	<b>Experiment/ Practical</b>	<b>Page no.</b>
<b>1.</b>	Implement concurrent echo client-server application in JAVA	<b>3</b>
<b>2.</b>	Implement a Distributed Chat Server using TCP Sockets in JAVA	<b>5</b>
<b>3.</b>	Implement concurrent day -time client-server application in JAVA	<b>9</b>
<b>4.</b>	Con figure following options on server socket and tests them: SO_KEEPA LIV E, SO_LINGER, SO_SNDBUF, SO_RCV BUF, TCP_NODELAY	<b>11</b>
<b>5.</b>	Write a program to Incrementing a counter in shared memory in JAVA	<b>14</b>
<b>6.</b>	Write a program to Simulate the Distributed Mutual Exclusion.	<b>17</b>
<b>7.</b>	Write a program to Implement Java RMI mechanism for accessing method	<b>19</b>
<b>8.</b>	Write a program to Create CORBA based server-client application	<b>23</b>



### Practical 1

**Aim:** Implement concurrent echo client-server application in java.

**Theory:** TCP stands for Transmission Control Protocol, a communications standard that enables application programs and computing devices to exchange messages over a network. It is designed to send packets across the internet and ensure the successful delivery of data and messages over networks. TCP organizes data so that it can be transmitted between a server and a client. It guarantees the integrity of the data being communicated over a network. Before it transmits data, TCP establishes a connection between a source and its destination, which it ensures remains live until communication begins. It then breaks large amounts of data into smaller packets, while ensuring data integrity is in place throughout the process.

**Code:** We make two files.

**Tcpserver.java:**

```
import java.io.*;
import java.net.*;

public class TcpServer {
    public static void main(String[] args) throws Exception {
        ServerSocket ss=new ServerSocket(8088);
        System.out.println("server is ready!");
        Socket ls=ss.accept();
        while (true){
            System.out.println("Client Port is "+ls.getPort());
            //READING DATA FROM CLIENT
            InputStream is=ls.getInputStream();
            byte data[]=new byte[50];
            is.read(data);
            String mfc=new String(data);
            //mfc: message from client
            mfc=mfc.trim();
            String mfs="The message was:"+mfc;
            //mfs: message from server
            //SENDING MSG TO CLIENT
            OutputStream os=ls.getOutputStream();
            os.write(mfs.getBytes());
        }
    }
}
```

**Tcpclient.java:**

```
import java.net.*;
import java.io.*;

class TcpClient {
    public static void main(String[] args) throws Exception {
```