Practical 4

Date:- 18/08/21

Roll No:- 19BCE248

Course No:- 2CS502 Computer Networks

AIM: Implement Selective Repeat Sliding Window Protocol.

Code:

Client.java

import java.lang.System;

import java.net.\*;

import java.io.\*;

public class Client {

static Socket connection;

public static void main(String a[]) throws SocketException {

try {

int v[] = new int[9];

int n = 0;

InetAddress addr = InetAddress.getByName("Localhost");

System.out.println(addr);

connection = new Socket(addr, 8011);

DataOutputStream out = new DataOutputStream(

connection.getOutputStream());

DataInputStream in = new DataInputStream(

connection.getInputStream());

int p = in.read();

System.out.println("No of frame is:" + p);

for (int i = 0; i < p; i++) {

v[i] = in.read();

System.out.println(v[i]);

}

v[5] = -1;

for (int i = 0; i < p; i++)

{

System.out.println("Received frame is: " + v[i]);

}

for (int i = 0; i < p; i++)

if (v[i] == -1) {

System.out.println("Request to retransmit packet no "

+ (i+1) + " again!!");

n = i;

out.write(n);

out.flush();

}

System.out.println();

v[n] = in.read();

System.out.println("Received frame is: " + v[n]);

System.out.println("quiting");

} catch (Exception e) {

System.out.println(e);

}

}

}

Server.java:

import java.io.DataInputStream;

import java.io.DataOutputStream;

import java.io.IOException;

import java.net.ServerSocket;

import java.net.Socket;

import java.net.SocketException;

public class Server {

static ServerSocket Serversocket;

static DataInputStream dis;

static DataOutputStream dos;

public static void main(String[] args) throws SocketException {

try {

int a[] = { 30, 40, 50, 60, 70, 80, 90, 100, 110 };

Serversocket = new ServerSocket(8011);

System.out.println("waiting for connection");

Socket client = Serversocket.accept();

dis = new DataInputStream(client.getInputStream());

dos = new DataOutputStream(client.getOutputStream());

System.out.println("The number of packets sent is:" + a.length);

int y = a.length;

dos.write(y);

dos.flush();

for (int i = 0; i < a.length; i++) {

dos.write(a[i]);

dos.flush();

}

int k = dis.read();

dos.write(a[k]);

dos.flush();

} catch (IOException e) {

System.out.println(e);

} finally {

try {

dis.close();

dos.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

}

Output:

