CS 380

Project 2

My repository for this class is under CS 380 – Computer Networks

<https://github.com/jarodNakamoto/College-CS-Courses.git>

Source Code Below:

import java.io.InputStream;

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.io.OutputStream;

import java.io.PrintStream;

import java.net.Socket;

import java.util.Scanner;

import java.io.IOException;

public final class PhysLayerClient {

public static final int PREAMBLE\_SIZE = 64;

public static final int MSG\_SIZE = 32;

public static int fiveBToFourB(int fiveB)

{

int fourB;

if((fiveB ^ 0x1E) == 0)

fourB = 0x0;

else if((fiveB ^ 0x009) == 0)

fourB = 0x1;

else if((fiveB ^ 0x014) == 0)

fourB = 0x2;

else if((fiveB ^ 0x015) == 0)

fourB = 0x3;

else if((fiveB ^ 0x00A) == 0)

fourB = 0x4;

else if((fiveB ^ 0x00B) == 0)

fourB = 0x5;

else if((fiveB ^ 0x00E) == 0)

fourB = 0x6;

else if((fiveB ^ 0x00F) == 0)

fourB = 0x7;

else if((fiveB ^ 0x012) == 0)

fourB = 0x8;

else if((fiveB ^ 0x013) == 0)

fourB = 0x9;

else if((fiveB ^ 0x016) == 0)

fourB = 0xA;

else if((fiveB ^ 0x017) == 0)

fourB = 0xB;

else if((fiveB ^ 0x01A) == 0)

fourB = 0xC;

else if((fiveB ^ 0x01B) == 0)

fourB = 0xD;

else if((fiveB ^ 0x01C) == 0)

fourB = 0xE;

else

fourB = 0xF;

return fourB;

}

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

try (Socket socket = new Socket("18.221.102.182", 38002)) {

//display that server connection was successful

String address = socket.getInetAddress().getHostAddress();

System.out.printf("Connected to: %s%n", address);

//recieves bytes

InputStream is = socket.getInputStream();

//receive preamble and calculate baseline

double baseline = 0.0;

for(int i = 0; i < PREAMBLE\_SIZE; i++)

{

baseline += ((double)(is.read()))/PREAMBLE\_SIZE;

}

System.out.println("Baseline established from preamble: " + baseline);

//get the encoded message

int[] bytesReceived = new int[MSG\_SIZE \* 10];

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

{

if(is.read() > baseline)

bytesReceived[i] = 1;

else

bytesReceived[i] = 0;

}

//get decodedBytes

byte[] decodedBytes = new byte[MSG\_SIZE];

int count = 1;

int countDecoded = 0;

int fiveB = bytesReceived[0];

int prev = bytesReceived[0];

for(int i = 1; i < bytesReceived.length; i++)

{

int curr = bytesReceived[i];

//shift 5B value over 1, append 0 to end

fiveB = fiveB << 1;

if(prev != curr) //then the signal was a one change end digit to one

fiveB = fiveB ^ 1;

//else it was a zero, do nothing

count++;

prev = curr;

//if 5 bits have been read then we have a half byte

if(count == 5)

{

int fourB = fiveBToFourB(fiveB);

decodedBytes[countDecoded] = (byte)fourB;

fiveB = 0;

}

//if 10 bits then we have a whole byte

if(count == 10)

{

int fourB = fiveBToFourB(fiveB);

int temp = decodedBytes[countDecoded];

temp = temp << 4;

temp = temp ^ fourB;

decodedBytes[countDecoded] = (byte)temp;

countDecoded++;

fiveB = 0;

count = 0;

}

}

//print to console the decoded bytes

System.out.print("Received 32 bytes: ");

for(byte b: decodedBytes)

System.out.print(String.format("%02X", b));

//sends bytes to server

OutputStream os = socket.getOutputStream();

os.write(decodedBytes);

//receive if msg is correct

int response = is.read();

if(response == 1)

System.out.println("\nResponse good");

else

System.out.println("\nResponse bad");

System.out.println("Disconnected from server.");

is.close();

}

}

}