CS 380

Exercise 3

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.OutputStream;

import java.net.Socket;

public final class Ex3Client {

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

try (Socket socket = new Socket("18.221.102.182", 38103)) {

//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();

//3) get how many bytes we are receiving

int numBytes = is.read();

System.out.println("Reading " + numBytes + " bytes.");

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

//4) receive the rest of bytes and store in array

byte[] bytesReceived = new byte[numBytes];

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

if(i%10 == 0)

{

System.out.println();

System.out.print(" ");

}

int byte1 = is.read();

bytesReceived[i] = (byte)(byte1);

System.out.print(String.format("%02X", bytesReceived[i]));

}

System.out.println();

//6) take array and pass it into checksum

short checkSum = checksum(bytesReceived);

//7) send checksum as sequence of bytes to server

Short val = new Short(checkSum);

byte[] byteArr = new byte[2];

Integer copy = new Integer(val.intValue());

//take the value and make it into two bytes

for(int i = byteArr.length-1; i >= 0; i--){

byteArr[i] = copy.byteValue();

copy = copy >> 8;

}

System.out.println("\nChecksum calculated: " + String.format("0x%04X", val.shortValue()) +".");

//sends bytes to server

OutputStream os = socket.getOutputStream();

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

os.write(byteArr[i]);

//8) receive if program worked

int rec = is.read();

if(rec == 1)

System.out.println("Response good");

else

System.out.println("Response bad");

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

is.close();

}

}

//5) write checksum

public static short checksum(byte[] b){

long sum = 0;

//while(count--)

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

//sum += \*buf++; add the value at refrence and then increment

int b1 = b[i];

if(b1 < 0)

b1 = b1 ^ 0xFFFFFF00;

i++;

int b2 = 0x00;

if(i < b.length)

{

b2 = b[i];

if(b2 < 0)

b2 = b2 ^ 0xFFFFFF00;

}

b1 = b1 << 8;

sum += (b1 ^ b2);

//if (sum & 0xFFFF0000)

if((sum & 0xFFFF0000) != 0x00000000){

/\*carry occurred. so wrap around \*/

sum = sum & 0xFFFF;

sum++;

}

}

//return the bit wise inverse of (sum & 0xFFFF)

//ones complement and return right most 16 bits

return (short)(~(sum & 0XFFFF));

}

}