import ithakimodem.\*;

import java.io.BufferedWriter;

import java.io.File;

import java.io.FileOutputStream;

import java.io.FileWriter;

import java.io.IOException;

import java.util.ArrayList;

public class UserApplication {

    public static void main(String[] param) {

        (new UserApplication()).Receiver();

    }

    public void Receiver() {

        Modem modem;

        modem=new Modem();

        modem.setSpeed(80000);

        modem.setTimeout(4000);

        modem.open("ithaki");

        int k=0;

        String welcomeMessage="";

        for (;;) {

            try {

                k=modem.read();

                if (k==-1){

                    System.out.print("We lost Ithaki!\n");

                    break;

                }

                System.out.print((char)k);

                welcomeMessage+=(char)k;

                if(welcomeMessage.contains("\r\n\n\n")){

                    System.out.print("\n---Rx Over---\n");

                    break;

                }

            } catch (Exception x) {

                System.out.print("Modem Exception!\n");

                break;

            }

        }

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // \*\*\*\*Echo requst\*\*\*\*

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        String echoStr="E7091\r";

        ArrayList<Long> times=new ArrayList<>();

        long durationStart=System.currentTimeMillis();

        do{

            long startTime = System.currentTimeMillis();

            String echoMessage="";

            modem.write(echoStr.getBytes());

            for (;;) {

                try {

                    k=modem.read();

                    if (k==-1){

                        System.out.print("We lost Ithaki!\n");

                        break;

                    }

                    System.out.print((char)k);

                    echoMessage+=(char)k;

                    if( echoMessage.contains("PSTART") && echoMessage.contains("PSTOP") ){

                        System.out.print("\t---Rx Over---\n");

                        break;

                    }

                }catch (Exception x) {

                    System.out.print("Modem Exception!\n");

                    break;

                }

            }

            long stopTime = System.currentTimeMillis();

            times.add(stopTime-startTime);

        }while(System.currentTimeMillis()-durationStart<240000);

        String timesTxt="C:\\Users\\Thanasis\\OneDrive - Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης\\6ο εξάμηνο\\Δίκτυα Ι\\Networks\_Ithaki\\times.txt";

        try {

            BufferedWriter timesWriter = new BufferedWriter(new FileWriter(timesTxt));

            for (int i = 0; i < times.size(); i++) {

                timesWriter.write(times.get(i)+"\n");

            }

            timesWriter.flush();

            timesWriter.close();

        } catch (IOException e) {

        }

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // \*\*\*\*Image requst\*\*\*

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        ArrayList<Integer> image=new ArrayList<Integer>();

        String imageStr = "M1207CAM=PTZ\r";

        modem.write(imageStr.getBytes());

        for (;;) {

            try {

                k = modem.read();

                if (k == -1) {

                    System.out.print("We lost Ithaki!\n");

                    break;

                }

                image.add(k);

                if((image.lastIndexOf(0xFF)==image.size()-2) && (image.lastIndexOf(0xD9)==image.size()-1)){

                    System.out.print("\n---Image Received!---\n");

                    break;

                }

            }catch (Exception x) {

                System.out.print("\nModem Exception!\n");

                break;

            }

        }

        byte[] imageBytes=new byte[image.size()-image.indexOf(0xFF)];

        for (int a = image.indexOf(0xFF) ; a < image.size(); a++) {

            imageBytes[a-image.indexOf(0xFF)] = image.get(a).byteValue();

        }

        File imageJpg=new File("C:\\Users\\Thanasis\\OneDrive - Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης\\6ο εξάμηνο\\Δίκτυα Ι\\Networks\_Ithaki\\egnatia\_wide.jpg");

        try {

            FileOutputStream im\_stream=new FileOutputStream(imageJpg);

            im\_stream.write(imageBytes);

            im\_stream.flush();

            im\_stream.close();

        } catch (Exception e) {

        }

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // \*\*\*\*Image requst\*\*\*

        // \*\*\*\*with errors\*\*\*\*

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        ArrayList<Integer> imageError=new ArrayList<Integer>();

        String imageErrorStr = "G8501CAM=PTZ\r";

        modem.write(imageErrorStr.getBytes());

        for (;;) {

            try {

                k = modem.read();

                if (k == -1) {

                    System.out.print("We lost Ithaki!\n");

                    break;

                }

                imageError.add(k);

                if((imageError.lastIndexOf(0xFF)==imageError.size()-2) && (imageError.lastIndexOf(0xD9)==imageError.size()-1)){

                    System.out.print("\n---Image with Errors Received!---\n");

                    break;

                }

            }catch (Exception x) {

                System.out.print("\nModem Exception!\n");

                break;

            }

        }

        byte[] imageErrorBytes=new byte[imageError.size()-imageError.indexOf(0xFF)];

        for (int a = imageError.indexOf(0xFF) ; a < imageError.size(); a++) {

            imageErrorBytes[a-imageError.indexOf(0xFF)] = imageError.get(a).byteValue();

        }

        File imageErrorJpg=new File("C:\\Users\\Thanasis\\OneDrive - Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης\\6ο εξάμηνο\\Δίκτυα Ι\\Networks\_Ithaki\\egnatiaErrors\_wide.jpg");

        try {

            FileOutputStream im\_er\_stream=new FileOutputStream(imageErrorJpg);

            im\_er\_stream.write(imageErrorBytes);

            im\_er\_stream.flush();

            im\_er\_stream.close();

        } catch (Exception e) {

        }

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // \*\*\*\*GPS requst\*\*\*\*\*

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        String GPSMessage="";

        String GPSStr = "P1196R=1064430\r";

        modem.write(GPSStr.getBytes());

        for (;;) {

            try {

                k = modem.read();

                if (k == -1) {

                    System.out.print("We lost Ithaki!\n");

                    break;

                }

                System.out.print((char)k);

                GPSMessage+=(char)k;

                if((GPSMessage.contains("START ITHAKI GPS TRACKING\r")) && (GPSMessage.contains("STOP ITHAKI GPS TRACKING\r"))){

                    System.out.print("\n---GPS Coordinates Received!---\n");

                    break;

                }

            }catch (Exception x) {

                System.out.print("\nModem Exception!\n");

                break;

            }

        }

        String[] messageLine=GPSMessage.split("\n");

        String[][] messageArr=new String[30][15];

        for(int i=0; i<30;i++){

            messageArr[i]=messageLine[i+1].split(",");

        }

        String TCommands="";

        for(int i=0; i<30; i+=4){

            int latDeg=(int)Double.parseDouble(messageArr[i][2])/100;               /\*DD \*/

            int latMin=(int)Double.parseDouble(messageArr[i][2])%100;              /\*EE \*/

            int latMinMin=(int)(Double.parseDouble(messageArr[i][2])\*100)%100;    /\*ZZ \*/

            int latSec=(int)(latMinMin\*0.6); //minute to second conversion

            int longDeg=(int)Double.parseDouble(messageArr[i][4])/100;              /\*AA \*/

            int longMin=(int)Double.parseDouble(messageArr[i][4])%100;             /\*BB \*/

            int longMinMin=(int)(Double.parseDouble(messageArr[i][4])\*100)%100;   /\*CC \*/

            int longSec=(int)(longMinMin\*0.6); //minute to second conversion

            TCommands=TCommands+"T="+Integer.toString(longDeg)+Integer.toString(longMin)+Integer.toString(longSec)

                +Integer.toString(latDeg)+Integer.toString(latMin)+Integer.toString(latSec);

            System.out.println(TCommands);

        }

        ArrayList<Integer> GPSImage=new ArrayList<Integer>();

        modem.write(("P1196"+TCommands+"\r").getBytes());

        for (;;) {

            try {

                k = modem.read();

                if (k == -1) {

                    System.out.print("We lost Ithaki!\n");

                    break;

                }

                GPSImage.add(k);

                if((GPSImage.lastIndexOf(0xFF)==GPSImage.size()-2) && (GPSImage.lastIndexOf(0xD9)==GPSImage.size()-1)){

                    System.out.print("\n---Google Maps Image Received!---\n");

                    break;

                }

            }catch (Exception x) {

                System.out.print("\nModem Exception!\n");

                break;

            }

        }

        byte[] GPSImageBytes=new byte[GPSImage.size()-GPSImage.indexOf(0xFF)];

        for (int a = GPSImage.indexOf(0xFF) ; a < GPSImage.size(); a++) {

            GPSImageBytes[a-GPSImage.indexOf(0xFF)] = GPSImage.get(a).byteValue();

        }

        File GPSImageJpg=new File("C:\\Users\\Thanasis\\OneDrive - Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης\\6ο εξάμηνο\\Δίκτυα Ι\\Networks\_Ithaki\\gps\_coords.jpg");

        try {

            FileOutputStream gps\_im\_stream=new FileOutputStream(GPSImageJpg);

            gps\_im\_stream.write(GPSImageBytes);

            gps\_im\_stream.flush();

            gps\_im\_stream.close();

        } catch (Exception e) {

        }

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // \*\*\*\*ARQ Implementation\*\*\*\*

        // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        String[] arqStr={"Q0267\r","R5043\r"};

        ArrayList<Long> arqTimes=new ArrayList<>();

        ArrayList<Integer> arqResends=new ArrayList<>();

        int packetCount=0;

        int errorCount=0;

        int errorSum=0;

        int errorFlag=0;

        long arqStartTime=0;

        long arqDurationStart=System.currentTimeMillis();

        do{

            String arqMessage="";

            if(errorFlag==0){

                arqStartTime = System.currentTimeMillis();

            }

            modem.write(arqStr[errorFlag].getBytes());

            for (;;) {

                try {

                    k=modem.read();

                    if (k==-1){

                        System.out.print("We lost Ithaki!\n");

                        break;

                    }

                    System.out.print((char)k);

                    arqMessage+=(char)k;

                    if( arqMessage.contains("PSTART") && arqMessage.contains("PSTOP") ){

                        System.out.print("\t---Rx Over---\n");

                        break;

                    }

                }catch (Exception x) {

                    System.out.print("Modem Exception!\n");

                    break;

                }

            }

            String arqData=arqMessage.substring(31, 47);

            String arqFCS=arqMessage.substring(49, 52);

            int xor=arqData.charAt(0);

            for(int i=0; i<arqData.length()-1;i++){

                xor=xor^arqData.charAt(i+1);

            }

            if(xor==Integer.parseInt(arqFCS)){

                long arqStopTime = System.currentTimeMillis();

                arqTimes.add(arqStopTime-arqStartTime);

                arqResends.add(errorCount);

                errorCount=0;

                System.out.println("No errors in packet!");

                errorFlag=0;

                packetCount++;

            }else{

                System.out.println("Errors in packet,resend requested!");

                errorFlag=1;

                errorCount++;

                packetCount++;

            }

            errorSum+=errorCount;

        }while(System.currentTimeMillis()-arqDurationStart<240000);

        System.out.println("Packet error count:"+errorSum);

        System.out.println("Packet count:"+packetCount);

        System.out.println("PacketLoss:"+((double)errorSum)/packetCount);

        String arqTimesTxt="C:\\Users\\Thanasis\\OneDrive - Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης\\6ο εξάμηνο\\Δίκτυα Ι\\Networks\_Ithaki\\arq\_times.txt";

        try {

            BufferedWriter arqTimesWriter = new BufferedWriter(new FileWriter(arqTimesTxt));

            for (int i = 0; i < arqTimes.size(); i++) {

                arqTimesWriter.write(arqTimes.get(i)+"\n");

            }

            arqTimesWriter.flush();

            arqTimesWriter.close();

        } catch (IOException e) {

        }

        String arqResendsTxt="C:\\Users\\Thanasis\\OneDrive - Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης\\6ο εξάμηνο\\Δίκτυα Ι\\Networks\_Ithaki\\arq\_resends.txt";

        try {

            BufferedWriter arqResendsWriter = new BufferedWriter(new FileWriter(arqResendsTxt));

            for (int i = 0; i < arqResends.size(); i++) {

                arqResendsWriter.write(arqResends.get(i)+"\n");

            }

            arqResendsWriter.flush();

            arqResendsWriter.close();

        } catch (IOException e) {

        }

        modem.close();

        System.out.println("Router Down!");

    }

}