Δίκτυα Υπολογιστών 1 Καλαϊτζίδης Ελευθέριος 10216

```
package ergasiadiktya;

import ithakimodem.*;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.PrintWriter;



public class main {


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

        String R = "R=1000091\r";

        String echo = "E2205\r"; //Echo request code

        String imageerrorfree = "M3837\r"; //Image request code error free

        String imagewitherros = "G7236\r"; //Image request code with errors

        String gps = "P2552" ;          //GPS request code

        String ack = "Q3793\r";

        String nack = "R2692\r";

        String gps1 = gps;

        gps = gps + R;

        int k;

        Modem modem = new Modem();

        modem.setSpeed(90000);

        modem.setTimeout(2000);

        modem.open("ithaki");

        for (;;) {


            try {

                k=modem.read();

                if (k==-1) break;

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

                }
            catch (Exception x) {
                break;
                }
                }


        System.out.println(" ");


        // echo start
        PrintWriter writer = new PrintWriter("C:/Users/lekam/Desktop/ergasia ithaki/echo.txt");
        long echotime =0;
        int echocounter=0;
        while(echotime < 420000) {
            long time = System.currentTimeMillis();
            modem.write(echo.getBytes());
            echocounter+=1;
            for (;;) {


                try {
                    k=modem.read();
                    if (k==-1) break;
                    System.out.print((char)k);
                    }
                catch (Exception x) {
                    break;
                    }
                    }
            long t = System.currentTimeMillis() - time;
            echotime += t ;
            System.out.println(" " + t );
            writer.println("Echotime No"+echocounter +" : "+t);
```

```
}
writer.print(echotime);
writer.close();


// echo finish




System.out.println(" ");


//image error free start
try(FileOutputStream image1 = new FileOutputStream("C:/Users/lekam/Desktop/ergasia ithaki/image1.jpeg")){




modem.write(imageerrorfree.getBytes());


for (;;) {


    try {
        k=modem.read();
        image1.write(k);
        if (k==-1) break;
        }
    catch (Exception x) {
        break;
        }
        }
image1.flush();
```

```
image1.close();

}


System.out.println("Image error free.");

//image error free finish


System.out.println(" ");




//image with errors start






try(FileOutputStream image2 = new FileOutputStream("C:/Users/lekam/Desktop/ergasia ithaki/image2.jpeg")){



modem.write(imagewitherros.getBytes());


for (;;) {


        try {

                k=modem.read();

                image2.write(k);

                if (k==-1) break;

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

                }

        catch (Exception x) {

                break;

                }
```

```
                    }


        image2.flush();

        image2.close();

        }

        System.out.print("Image with errors.");


        //image with errors finish









        System.out.println(" ");













        //GPS start





        String dd = "";

        String ee = "";

        String zz = "";

        String aa = "";

        String bb = "";

        String cc = "";


        String T1 = "";

        String T2 = "";

        String T3 = "";
```

```
String T4 = "";
String trace = "" ;
int counter1 = 0;
int bit=0;



modem.write(gps.getBytes());
for (;;) {

        try {
                k=modem.read();
                if (k==-1) break;
                System.out.print((char)k);
                }
        catch (Exception x) {
                break;
                }
        if((char)k=='$') {
                counter1 += 1;
                bit = 0;
         }
        if(counter1 == 1) {
                bit += 1;
                if(bit == 19 || bit == 20) {
                        dd = dd + (k-48);
                }

                if(bit == 21 || bit == 22) {
                        ee = ee + (k-48);
                }
                if(bit == 24|| bit == 25 || bit == 26 || bit == 27) {
                    zz = zz + (k-48);
                }
                if(bit == 27) {
                    int o = Integer.parseInt(zz)*60;
                    zz = String.valueOf(o);
                    zz = zz.substring(0,2);
                }
                if(bit == 32 || bit == 33) {
                    aa = aa + (k-48);
                }
                if(bit == 34 || bit== 35) {
                    bb = bb + (k-48);
                }
                if(bit == 37 || bit == 38 || bit == 39 || bit == 40) {
                    cc = cc + (k-48);
                }
                if(bit == 40) {
                    int o = Integer.parseInt(cc)*60;

                    cc = String.valueOf(o);
                    cc = cc.substring(0,2);
                    T1 = gps1 + "T=" + aa + bb + cc + dd + ee + zz;

                    dd="";
                    ee="";
                    zz="";
                    aa="";
                    bb="";
                    cc="";
                }
            }
```

```
if(counter1 == 30) {
        bit += 1;
        if(bit == 19 || bit == 20) {
                dd = dd + (k-48);
        }

        if(bit == 21 || bit == 22) {
                ee = ee + (k-48);
        }
        if(bit == 24|| bit == 25 || bit == 26 || bit == 27) {
                zz = zz + (k-48);
        }
        if(bit == 27) {
               int o = Integer.parseInt(zz)*60;
               zz = String.valueOf(o);
               zz = zz.substring(0,2);
        }
        if(bit == 32 || bit == 33) {
                aa = aa + (k-48);
        }
        if(bit == 34 || bit == 35) {
                bb = bb + (k-48);
        }
        if(bit == 37 || bit == 38 || bit == 39 || bit == 40) {
                cc = cc + (k-48);
        }
        if(bit == 40) {
                int o = Integer.parseInt(cc)*60;
                cc = String.valueOf(o);
                cc = cc.substring(0,2);
```

```
                    T2 = "T=" + aa + bb + cc + dd + ee + zz ;


                    dd="";

                    ee="";

                    zz="";

                    aa="";

                    bb="";

                    cc="";

                }

        }


        if(counter1 == 60) {

                bit += 1;

                if(bit == 19 || bit == 20) {

                        dd = dd + (k-48);

                }


                if(bit == 21 || bit == 22) {

                        ee = ee + (k-48);

                }

                if(bit == 24|| bit == 25 || bit == 26 || bit == 27) {

                        zz = zz + (k-48);

                }

                if(bit == 27) {

                        int o = Integer.parseInt(zz)*60;

                        zz = String.valueOf(o);

                        zz = zz.substring(0,2);

                }

                if(bit == 32 || bit == 33) {

                        aa = aa + (k-48);

                }
```

```
    if(bit == 34 || bit == 35) {

        bb = bb + (k-48);

    }

    if(bit == 37 || bit == 38 || bit == 39 || bit == 40) {

        cc = cc + (k-48);

    }

    if(bit == 40) {

        int o = Integer.parseInt(cc)*60;

        cc = String.valueOf(o);

        cc = cc.substring(0,2);

        T3 = "T=" + aa + bb + cc + dd + ee + zz ;


        dd="";

        ee="";

        zz="";

        aa="";

        bb="";

        cc="";

    }

 }


if(counter1 == 90) {

    bit += 1;

    if(bit == 19 || bit == 20) {

        dd = dd + (k-48);

    }


    if(bit == 21 || bit == 22) {

        ee = ee + (k-48);

    }

    if(bit == 24|| bit == 25 || bit == 26 || bit == 27) {
```

```
                    zz = zz + (k-48);

                }

                if(bit == 27) {

                    int o = Integer.parseInt(zz)*60;

                    zz = String.valueOf(o);

                    zz = zz.substring(0,2);

                }

                if(bit == 32 || bit == 33) {

                    aa = aa + (k-48);

                }

                if(bit == 34 || bit == 35) {

                    bb = bb + (k-48);

                }

                if(bit == 37 || bit == 38 || bit == 39 || bit == 40) {

                    cc = cc + (k-48);

                }

                if(bit == 40) {

                    int o = Integer.parseInt(cc)*60;

                    cc = String.valueOf(o);

                    cc = cc.substring(0,2);

                    T4 = "T=" + aa + bb + cc + dd + ee + zz +"\r";


                    dd="";

                    ee="";

                    zz="";

                    aa="";

                    bb="";

                    cc="";

                }

            }
```

```
                }
        trace = T1 + T2 + T3 + T4;
        System.out.println(trace);
        modem.write(trace.getBytes());
        try(FileOutputStream traceimage = new
FileOutputStream("C:/Users/lekam/Desktop/ergasia ithaki/trace.jpeg")){
            for (;;) {

                try {
                    k=modem.read();
                    traceimage.write(k);
                    if (k==-1) break;
                    }
                catch (Exception x) {
                    break;
                    }
                    }
            traceimage.flush();
            traceimage.close();
            }

        //gps finish

        System.out.println(" ");

        //ARQ start
        int numofpacket=0;
        PrintWriter writer1 = new PrintWriter("C:/Users/lekam/Desktop/ergasia
ithaki/arq.txt");
        long arqtime = 0;
        long packet_time = 0;
        String arq = ack;
        int counter = 0;
        while (arqtime<240000 || arq == nack) {
            long time = System.currentTimeMillis();
            int fcs = 1;
            int xor = 0;
            modem.write(arq.getBytes());
            counter +=1;
            int f = 0;
            int c = 0;
            int s = 0;
            int bitcounter = 0;
            for (;;) {

                try {
                    bitcounter+=1;
                    k=modem.read();
                    if (k==-1) break;
                    System.out.print((char)k);
                    if (bitcounter==32) xor=k;
                    if (bitcounter>32 && bitcounter<=47) {
                        xor = k ^ xor;
                    }
                    if (bitcounter==50) f=k-48;
                    if (bitcounter==51) c=k-48;
                    if (bitcounter==52) s=k-48;

                    }
                catch (Exception x) {
                    break;
                    }
                    }
                fcs = f*100+c*10+s;
                System.out.print("\n");
                System.out.println(xor);
                System.out.println(fcs);
                System.out.println(xor==fcs);
                if (xor != fcs) {
                        packet_time +=System.currentTimeMillis()-time;
                        arq = nack;
                        arqtime +=  System.currentTimeMillis()-time;
                }
                if (xor == fcs) {
                        numofpacket+=1;
                        packet_time +=System.currentTimeMillis()-time;
                        System.out.println(packet_time);
                        writer1.print("Packet No" + numofpacket + ": \n" );
                        writer1.print("   Packet time : ");
                        writer1.println(packet_time);
                        writer1.print("   Sent times : ");
                        writer1.println(counter+"\n");
                        packet_time = 0;
                        arqtime +=  System.currentTimeMillis()-time;
                        arq = ack;
                        System.out.println(counter);


                        counter = 0;
                }


                System.out.println(arqtime);
                System.out.println(" ");


        }
```

```
                writer1.println("\n\n Total time : "+ arqtime);
                writer1.close();


                //ARQ finish


                modem.close();


        }



}
```