

Работа призера заключительного этапа  
командной инженерной олимпиады школьников  
**Олимпиада Национальной технологической инициативы**

Профиль «Технологии беспроводных связей»

**Исаков Антон Сергеевич**

**Класс:**11

**Школа:**МАОУ Лицей №7

**Город:**Красноярск

**Регион:**Красноярский край

**Уникальный номер участника:**1417491

**Степикid:** 22414754

**Команда на заключительном**

**этапе:**Буйный чайник 2.0

**Параллель:** 10-11

**Результаты заключительного этапа (оценка по ненормированным баллам):**

Индивидуальная часть (не норм)									Командная часть (не норм)						Результат, нормир. На 100 баллов	
№	Математика			Информатика			ИТОГО	Макс балл	1	2	3	4	5	ИТОГО	Макс балл	47,075
	1	2	3	1	2	3										
	0	0	0	25	20	4,375	49,375	100	5,1	31,36	11,14	0	0	47,59	126	



# Индивидуальная часть

## МАТЕМАТИКА

Решение победителя:



Олимпиада НТИ

ФИО Исмаев Атану Керемович

Город Красноярск

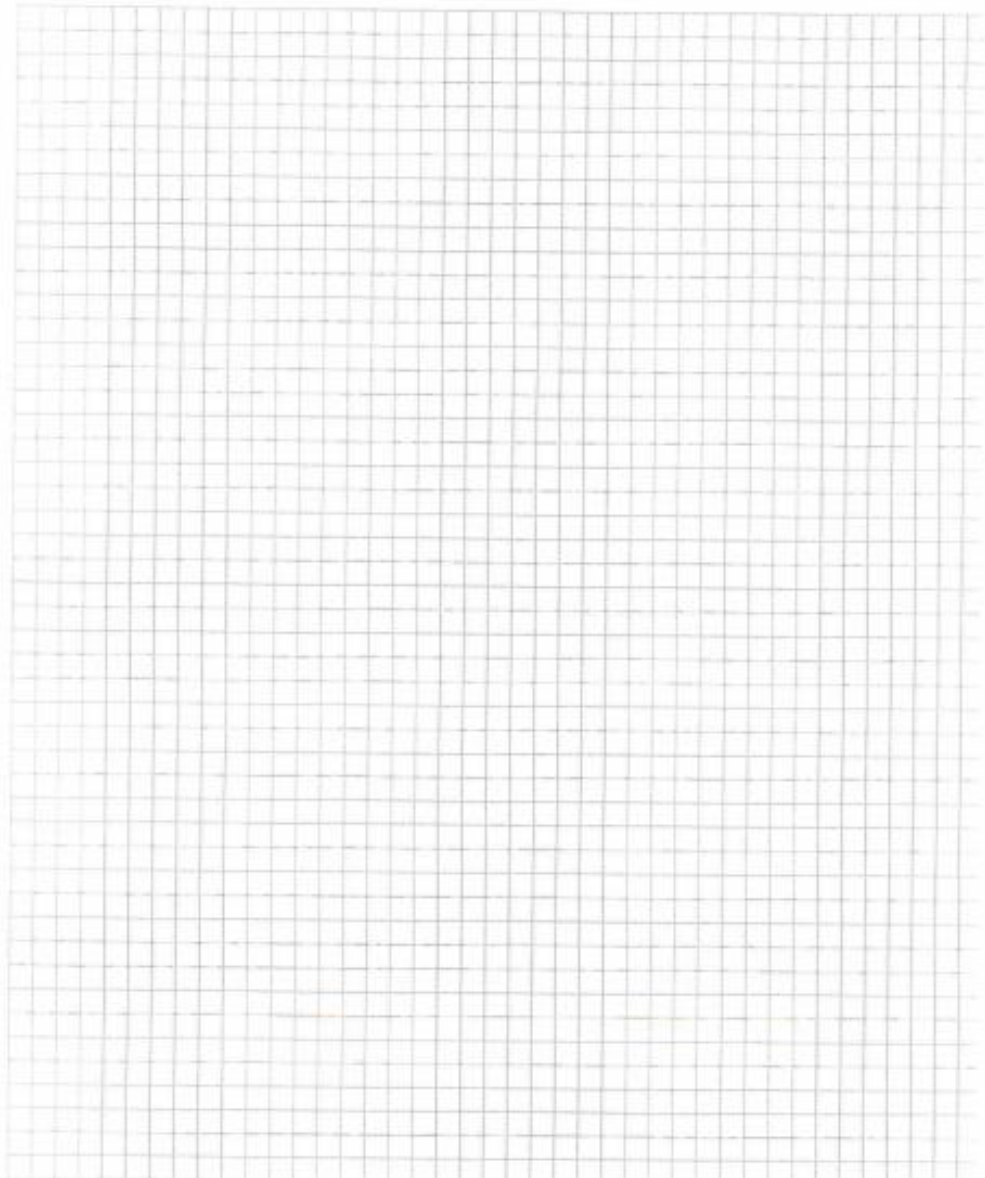
Школа № Лагуна №1

Командная инженерная олимпиада «Олимпиада ИТИ»

Направление ИТС

Предмет Математика

Номер участника 1417491



**Балл призёра: 0 за предметный тур по математике, задачи не были решены**

## ИНФОРМАТИКА

Степик-id: 22414754

### Задача 1. Платежные каналы

#### Решение призёра:

```
import java.util.*;
public class Main {
    Scanner in=new Scanner(System.in);
    ArrayList<ArrayList<Integer>> graph=new ArrayList<>();
    int n,m;
    public static void main(String args[]){
        new Main();
    }
    public Main(){
        n=in.nextInt();
        for(int i=0;i<n;i++){
            graph.add(new ArrayList<>());
        }
        m=in.nextInt();
        for(int i=0;i<m;i++){
            graph.get(in.nextInt()-1).add(in.nextInt()-1);
        }
        System.out.println(bfs(in.nextInt()-1,in.nextInt()-1)?"Yes":"No");
    }
    public boolean bfs(int a,int b){
        Stack<Integer> st=new Stack<>();
        HashSet<Integer> set=new HashSet<>();
        st.add(a);
        set.add(a);
        while (!st.isEmpty()){
            int s=st.pop();
            for(Integer to:graph.get(s)){
                if(to==b)return true;
                if(!set.contains(to)){
                    set.add(to);
                    st.add(to);
                }
            }
        }
        return false;
    }
}
```

```
import java.util.*;
public class Main {
    Scanner in=new Scanner(System.in);
    ArrayList<ArrayList<Integer>> graph=new ArrayList<>();
    int n,m;
    public static void main(String args[]){
        new Main();
    }
}
```

```

    }
    public Main(){
        n=in.nextInt();
        for(int i=0;i<n;i++){
            graph.add(new ArrayList<>());
        }
        m=in.nextInt();
        for(int i=0;i<m;i++){
            graph.get(in.nextInt()-1).add(in.nextInt()-1);
        }
        System.out.println(bfs(in.nextInt()-1,in.nextInt()-1)?"Yes":"No");
    }
    public boolean bfs(int a,int b){
        Stack<Integer> st=new Stack<>();
        HashSet<Integer> set=new HashSet<>();
        st.add(a);
        set.add(a);
        while (!st.isEmpty()){
            int s=st.pop();
            for(Integer to:graph.get(s)){
                if(to==b) return true;
                if(!set.contains(to)){
                    set.add(to);
                    st.add(to);
                }
            }
        }
        return false;
    }
}

```

```

import java.util.*;
public class Main {
    Scanner in=new Scanner(System.in);
    ArrayList<ArrayList<Integer>> graph=new ArrayList<>();
    int n,m;
    public static void main(String args[]){
        new Main();
    }
    public Main(){
        n=in.nextInt();
        for(int i=0;i<n;i++){
            graph.add(new ArrayList<>());
        }
        m=in.nextInt();
        for(int i=0;i<m;i++){
            graph.get(in.nextInt()-1).add(in.nextInt()-1);
        }
        System.out.println(bfs(in.nextInt()-1,in.nextInt()-1)?"Yes":"No");
    }
    public boolean bfs(int a,int b){
        Stack<Integer> st=new Stack<>();
        HashSet<Integer> set=new HashSet<>();

```

```

        st.add(a);
        set.add(a);
        while (!st.isEmpty()){
            int s=st.pop();
            for(Integer to:graph.get(s)){
                if(to==b)return true;
                if(!set.contains(to)){
                    set.add(to);
                    st.add(to);
                }
            }
        }
        return false;
    }
}

```

**Полученный балл:25**

## **Задача 2. Платежные каналы с комиссией**

### **Решениепризёра:**

```

import java.util.*;
public class Main {
    Scanner in=new Scanner(System.in);
    ArrayList<ArrayList<Edge>> graph=new ArrayList<>();
    int n,m,s,t,q,q0;
    public static void main(String args[]){
        new Main();
    }
    public Main(){
        n=in.nextInt();
        for(int i=0;i<n;i++){
            graph.add(new ArrayList<>());
        }
        m=in.nextInt();
        for(int i=0;i<m;i++){
            graph.get(in.nextInt()-1).add(new Edge(in.nextInt()-1,in.nextInt()));
        }
        s=in.nextInt()-1;t=in.nextInt()-1;
        q=in.nextInt();q0=in.nextInt();
        for(int i=0;i<q;i++){
            graph.get(in.nextInt()-1).add(new Edge(in.nextInt()-1,in.nextInt()+q0));
        }
        intans=bfs(s,t,n);
        System.out.println(ans>0?ans:-1);
    }
    public intbfs(inta,intb,int n){
        int[] p=new int[n];
        Stack<Integer>st=new Stack<>();
        HashSet<Integer> set=new HashSet<>();
        st.add(a);
        set.add(a);
        while (!st.isEmpty()){
            int s=st.pop();

```

```

        for(Edge to:graph.get(s)){
            if(!set.contains(to.to) || p[to.to]>p[s]+to.price){
                set.add(to.to);
                st.add(to.to);
                p[to.to]=p[s]+to.price;
            }
        }
    }
    return p[b];
}
}
}
class Edge{
    int to;
    int price;
    Edge(intt,int p){
        to=t;
        price=p;
    }
}
}

```

```

import java.io.*;
import java.util.*;
public class Main {
    //Scanner in=new Scanner(System.in);
    //BufferedReader in=new BufferedReader(new InputStreamReader(System.in));
    StreamTokenizer st=new StreamTokenizer(new BufferedInputStream(System.in));
    public int nextInt(){
        try {
            st.nextToken();
        } catch (IOException e) {
            e.printStackTrace();
        }
        return (int) st.nval;
    }
    ArrayList<ArrayList<Edge>> graph=new ArrayList<>();
    int n,m,s,t,q,q0;
    public static void main(String args[]){
        new Main();
    }
    public Main(){
        n=nextInt();
        for(int i=0;i<n;i++){
            graph.add(new ArrayList<>());
        }
        m=nextInt();
        for(int i=0;i<m;i++){
            graph.get(nextInt()-1).add(new Edge(nextInt()-1,nextInt()));
        }
        s=nextInt()-1;t=nextInt()-1;
        q=nextInt();q0=nextInt();
        for(int i=0;i<q;i++){
            graph.get(nextInt()-1).add(new Edge(nextInt()-1,nextInt()+q0));
        }
    }
}

```

```

    intans=bfs(s,t,n);
    System.out.println(ans!=2000000000?ans:-1);
}
public intbfs(inta,intb,int n){
    int[] p=new int[n];
    HashSet<Integer> set=new HashSet<>();
    set.add(a);
    intsel=a;
    int min=0,minV;
    while (!set.contains(b)){
        minV=2000000000;
        for(Edge to:graph.get(a)){
            if(p[sel]+to.price<minV){
                minV=p[sel]+to.price;
                min=to.to;
            }
        }
        p[min]=minV;
        set.add(min);
        sel=min;
    }
    return p[b];
}
}
class Edge{
    int to;
    int price;
    Edge(intt,int p){
        to=t;
        price=p;
    }
}
}

```

Полученный балл: 20

### Задача 3. Деревотранзакций

#### Решение призёра:

```

import java.io.*;
import java.util.*;
public class Main {
    StreamTokenizer st=new StreamTokenizer(new BufferedInputStream(System.in));
    public intnextInt(){
        try {
            st.nextToken();
        } catch (IOException e) {
            e.printStackTrace();
        }
        return (int) st.nval;
    }
    ArrayList<ArrayList<Edge>> graph=new ArrayList<>();
    intn,q,f,t,v;
    public static void main(String args[]){
        new Main();
    }
}

```



```

}
public Main(){
    n=nextInt();q=nextInt();
    for(int i=0;i<n;i++){
        graph.add(new ArrayList<>());
    }
    for(int i=0;i<n-1;i++){
        f=nextInt()-1;
        t=nextInt()-1;
        v=nextInt();
        graph.get(f).add(new Edge(t,v));
        graph.get(t).add(new Edge(f,v));
    }
    for(int i=0;i<q;i++){
        System.out.println(dfs(-1,nextInt()-1,nextInt(),0));
    }
}
public intdfs(intfr,ints,inttval,intval){
    intans=0;
    if(fr!=-1 &&tval==val){
        ans++;
    }
    for(Edge to:graph.get(s)){
        if(to.to!=fr){
            ans+=dfs(s,to.to,tval,val^to.price);
        }
    }
    return ans;
}
}
class Edge{
    int to;
    int price;
    Edge(intt,int p){
        to=t;
        price=p;
    }
}
}

```

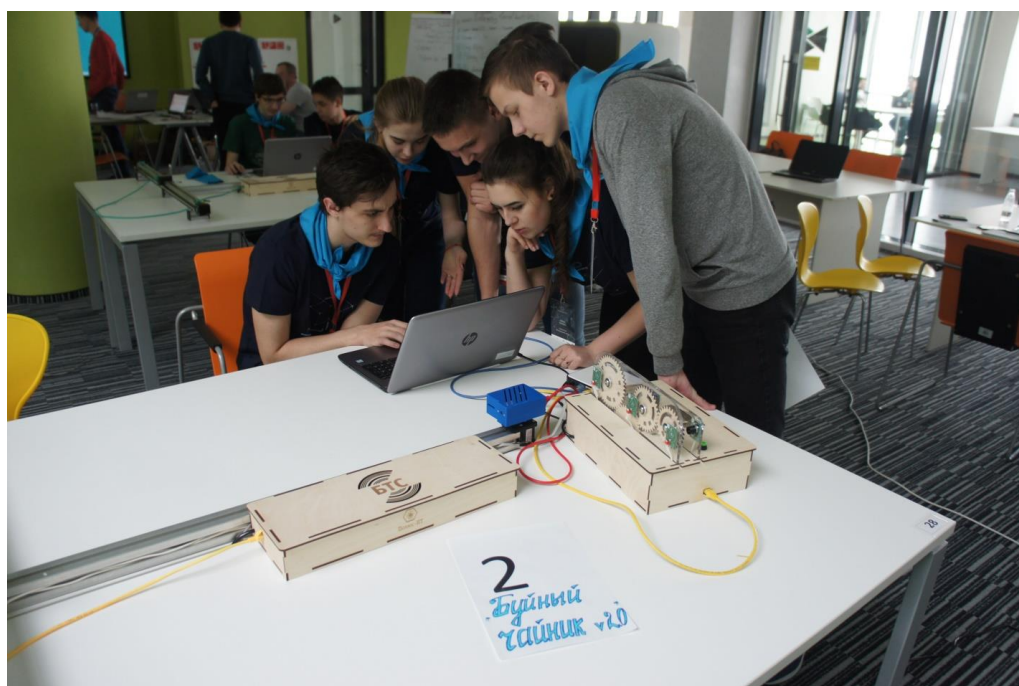
**Полученный балл: 4,375**

## Командная часть

Результаты были получены в рамках выступления команды: Буйный чайник 2.0

Личный состав команды:

- Исаков Антон Сергеевич
- Евсеев Максим Дмитриевич
- Лебедев Алексей Павлович
- Дугаева Светлана Анатольевна
- Науман Алина Владимировна





## Задача 2.

Решение:

### Кодер

```
import java.io.*;

class encoder {
    static int[] pow={1,2,4,8,16,32,64,128};
    static final int STARTBL=129;
    static final int ENDBL=258;

    public static void main(String args[]) throws Exception {
        new encoder(new ByteArrayInputStream(get(new FileInputStream(args[0]))),new
        FileOutputStream(args[1]));
        /*OutputStream out=new FileOutputStream("tets.txt");
        InputStream in=new FileInputStream("test.txt");
        out.write(get(in));
        out.close();*/
    }

    static int[] getBitArr(byte b){
        int[] out=new int[3];
        int bin=Byte.toUnsignedInt(b);
        out[0]+=((bin&pow[7])>0?224:0);
        out[0]+=((bin&pow[6])>0?28:0);
        out[0]+=((bin&pow[5])>0?3:0);
        out[1]+=((bin&pow[5])>0?128:0);
        out[1]+=((bin&pow[4])>0?112:0);
        out[1]+=((bin&pow[3])>0?14:0);
        out[1]+=((bin&pow[2])>0?1:0);
        out[2]+=((bin&pow[2])>0?192:0);
        out[2]+=((bin&pow[1])>0?56:0);
        out[2]+=((bin&pow[0])>0?7:0);
        return out;
    }

    public encoder(InputStream in,OutputStream out){
        try {
            byte[]bin=new byte[in.available()];
            byte[] buf=new byte[in.available()*9];
            in.read(bin);
            for(int i=0;i<bin.length;i++){
                int[] b=new int[9];
                int q[]=getBitArr(bin[i]);
                for(int j=0;j<3;j++){
                    int w[]=getBitArr((byte) q[j]);
                    for(int k=0;k<3;k++){
                        b[j*3+k]=w[k];
                    }
                }
                for(int j=0;j<9;j++){
                    buf[i*9+j]=(byte)b[j];
                }
            }
        }
    }
}
```

```

        }
    }
    out.write(buf);
    out.close();
    in.close();
} catch (Exception e) {
    e.printStackTrace();
}
}

public static byte[] get(InputStream in) throws Exception {
    int size = in.available();
    int blocks = (size + 123) / 124;
    byte[] bin = new byte[blocks * 128];
    in.read(bin);
    byte[] buf = new byte[3 * blocks * 128 + 2];

    buf[0] = (byte) (blocks >> 8);
    buf[1] = (byte) (blocks);
    for (int k = 0; k < 3; k++) {
        int c = blocks * 128 * k;
        try {
            for (int i = 0; i < blocks; i++) {
                buf[i * 128 + 2 + c] = (byte) (STARTBL >> 8);
                buf[i * 128 + 3 + c] = (byte) (STARTBL);
                buf[i * 128 + 4 + c] = (byte) (i >> 8);
                buf[i * 128 + 5 + c] = (byte) (i);
                for (int j = 0; j < 120; j++) {
                    buf[i * 128 + 6 + j + c] = bin[i * 120 + j];
                }
                buf[i * 128 + 126 + c] = (byte) (i >> 8);
                buf[i * 128 + 127 + c] = (byte) (i);
                buf[i * 128 + 128 + c] = (byte) (ENDBL >> 8);
                buf[i * 128 + 129 + c] = (byte) (ENDBL);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    return buf;
}
}

```

## Декодер

```

import java.io.*;

class decoder {

    static int[] pow = {1, 2, 4, 8, 16, 32, 64, 128};
    byte[] bin, buf;
    int k, blocks, blockNum;
    static final int STARTBL = 129;
    static final int ENDBL = 258;

```

```

public static void main(String args[]) throws Exception {
    new decoder(new FileInputStream(args[0]),new FileOutputStream(args[1]));
}

static int getBit(int in,int pos){
    return ((in&pow[pos])>0)?1:0;
}

static byte getArrBit(byte b1,byte b2,byte b3){
    int[]
in={Byte.toUnsignedInt(b1),Byte.toUnsignedInt(b2),Byte.toUnsignedInt(b3)};
    int out=0;
    int c;
    //0
    c=getBit(in[0],7)+getBit(in[0],6)+getBit(in[0],5);
    out+=c>1?128:0;
    //1
    c=getBit(in[0],4)+getBit(in[0],3)+getBit(in[0],2);
    out+=c>1?64:0;
    //2
    c=getBit(in[0],1)+getBit(in[0],0)+getBit(in[1],7);
    out+=c>1?32:0;
    //3
    c=getBit(in[1],6)+getBit(in[1],5)+getBit(in[1],4);
    out+=c>1?16:0;
    //4
    c=getBit(in[1],3)+getBit(in[1],2)+getBit(in[1],1);
    out+=c>1?8:0;
    //5
    c=getBit(in[1],0)+getBit(in[2],7)+getBit(in[2],6);
    out+=c>1?4:0;
    //6
    c=getBit(in[2],5)+getBit(in[2],4)+getBit(in[2],3);
    out+=c>1?2:0;
    //7
    c=getBit(in[2],2)+getBit(in[2],1)+getBit(in[2],0);
    out+=c>1?1:0;
    return (byte)out;
}

public byte decodeByte(int pos){
    return
getArrBit(getArrBit(bin[pos],bin[pos+1],bin[pos+2]),getArrBit(bin[pos+3],bin[pos+4],b
in[pos+5]),getArrBit(bin[pos+6],bin[pos+7],bin[pos+8]));
}

public int isValid(int pos){
    int
start=(Byte.toUnsignedInt(decodeByte(pos))<<8)+Byte.toUnsignedInt(decodeByte(pos+1*9)
);
    int
finish=(Byte.toUnsignedInt(decodeByte(pos+126*9))<<8)+Byte.toUnsignedInt(decodeByte(p

```

```

os+127*9));
    if(start!=STARTBL || finish!=ENDBL)return -1;

start=(Byte.toUnsignedInt(decodeByte(pos+2*9))<<8)+Byte.toUnsignedInt(decodeByte(pos+
3*9));

finish=(Byte.toUnsignedInt(decodeByte(pos+124*9))<<8)+Byte.toUnsignedInt(decodeByte(p
os+125*9));
    if(start==finish){
        return start;
    } else {
        return -1;
    }
}

public void decodeBlock(int numB){
    k+=36;
    for(int i=0;i<120;i++){
        buf[numB*120+i]=decodeByte(k+i*9);
    }
    k+=1080;
}

public decoder(InputStream in,OutputStream out){
    try {
        bin=new byte[in.available()];
        buf=new byte[bin.length/27];
        in.read(bin);

blocks=(Byte.toUnsignedInt(decodeByte(0))<<8)+Byte.toUnsignedInt(decodeByte(9));
        k=18;
        try{
            while (true) {
                while ((blockNum = isValid(k)) == -1) {
                    k++;
                }
                decodeBlock(blockNum);
            }
        }catch (ArrayIndexOutOfBoundsException e){

        }
        out.write(buf);
        out.close();
        in.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
}
}

```

**Полученный балл за задачу: 31,36**

### Задача 3

Алгоритм сопровождения.

Решение:

```
import java.net.*;
import java.io.*;
import java.math.*;

class example {

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

        Client client=new Client();
        int speed=75;
        int dir=1;
        long dx=0;
        client.start();

        while(dir==1){
            while (!client.readstatus()){
                System.out.println("a");
                Thread.sleep(1000);
            }
            dx=client.getDx();
            speed=(int)Math.abs(dx)/2;
            //System.out.println(dx);
            if(dx>0)client.left(speed);
            else client.right(speed);
            //Thread.sleep(1);
        }

        client.quit();
        client.stop();
    }
}
```

Полученный результат: реализовано автоматическое наведение «радара» на «спутник»

**Полученный балл за задачу: 11,14**

### Задача 4.

Решение: не было предоставлено

Полученный балл за задачу: 0

### Задача 5

Решение: не было предоставлено

Полученный балл за задачу: 0