

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

Профиль «АВТОНОМНЫЕ ТРАНСПОРТНЫЕ СИСТЕМЫ»

Дунаев Алексей Игоревич

Класс: 11

Город: Йошкар-Ола

Школа: ГБОУ РМЭ "Политехнический
лицей-интернат"

Регион: Республика Марий Эл

Уникальный номер участника: 331

**Команда на заключительном
этапе:** Бритые ежики

Результаты заключительного этапа:

№	Индивидуальный этап												Командный этап		ИТОГ		
	Физика				Информатика								За задачи	Коэффициент 40%		баллы	Коэффициент 60%
331	4	6	4	4	0	0	0	0	0	0	0	0	18	4,8	63,0	15,12	19,920

Индивидуальная часть

Персональный лист участника с номером 331:



Олимпиада НТИ

ФИО Дунаев Алексей Игоревич

Город Кокшар-Аул

Школа № ГБОУ Республика Марий Эл
"Технический музей-интернат"

Физика

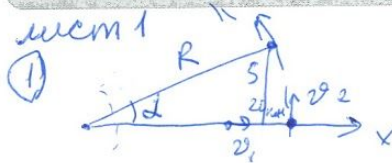
Командная инженерная олимпиада «Олимпиада НТИ»

Направление АТС

Предмет Физика

Номер участника 331

1	2	3	4	Σ
4	6	4	4	18



$$R = 6400 + 35786\pi = 42186 \text{ мм.}$$

$$\omega = 20 \text{ } \cancel{V_1} = 1,77 \text{ км/с.}$$

2) для малых α :

$$t \approx \frac{20}{V_1}$$

$$S \approx R \cdot \sin \alpha$$

$$S \approx 35 \text{ мм.}$$

$$V_2 = \omega \cdot R = \frac{2\pi R}{T} = 3 \text{ км/с.}$$

$$P = 24 \text{ м}$$

$$\alpha = \omega \cdot t$$

$$\begin{cases} x = (R - 20) + V \cdot t \\ x = \cos(\omega \cdot t) \cdot R \\ y = R \sin \omega t \\ S = \sqrt{x^2 + y^2} - \min \end{cases}$$

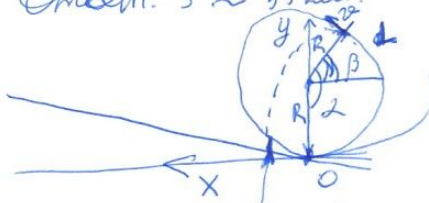
$$S'_t = 0$$

точное решение.

4 балла

$$\text{Ответ: } S \approx 35 \text{ мм.}$$

2



$$L = 11,78$$

$$R = 5 \text{ м.}$$

$$L = R \alpha$$

$$\alpha = 2,35$$

$$\beta = \alpha - 2\pi = 0,7792.$$

6 баллов

$$\frac{v^2}{R} = g \sin \beta$$

$$v = 5,927 \text{ м/с.}$$

$$v_x = v \cdot \sin \beta = 4,165 \text{ м/с}$$

$$v_y = v \cdot \cos \beta = 4,2169 \text{ м/с.}$$

$$y_0 = R + R \sin \beta = 8,5135 \text{ м}$$

$$x_0 = -R \cos \beta = -3,55 \text{ м}$$

точное

решение:

$$\begin{cases} y = y_0 + v_y \cdot t - \frac{gt^2}{2} \\ x = x_0 + v_x \cdot t \\ x^2 + (y - R)^2 = R^2 \end{cases}$$

Командная инженерная олимпиада «Олимпиада НТИ»

Направление АТС

Предмет Физика

Номер участника 331

лист 2



$R = 10 \text{ Ом.}$

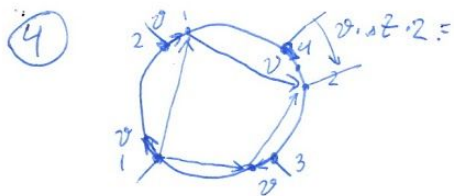
$$\begin{cases} I_P = \frac{E}{R+r} \\ I_P \cdot 0.18 = \frac{E}{R+r+R_x} \\ I_P \cdot 0.149 = \frac{E}{\frac{R^2 + R_x R}{2R + R_x} + r} \end{cases}$$

возможные R_x :

$$-R_x^3 \cdot \frac{81}{2050} + R_x^2 \cdot \frac{162}{205} - R_x \cdot 18.94218 - 9.4218 = 0.$$

$R_x > 0.$

$R_x \approx 50 \text{ Ом.}$



$$R = 42186 \text{ км}$$

$$v = c \approx 3 \cdot 10^8 \text{ м/с.}$$

$$v = \frac{2\pi}{T} \quad R = 3 \text{ км/с.}$$

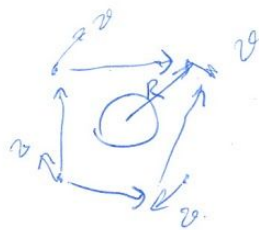
$$S = \frac{2R}{\sqrt{2}}$$

$$\Delta t \approx \frac{S}{v} = 1.9 \cdot 10^{-4} \text{ с.}$$

$$v \cdot \Delta t = 0.6 \text{ м.}$$

задержка $\approx 2 \Delta t$ в 0.4 мс для передачи данных достаточно мала, может повлиять только если сигнал используется для определения положения, за счёт скорости распространения.

4 балла



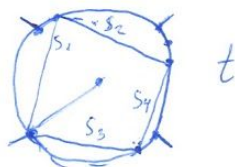
$$R = 6400 + 35786 = 42186 \text{ км.}$$

$$v = c \approx 3 \times 10^8 \text{ м/с.}$$

$$T = 24 \cdot 60 \cdot 60.$$

$$\omega = \frac{2\pi}{T} =$$

$$v = \omega \cdot R = 3 \text{ км/с.}$$



$$\sqrt{s^2 + s^2} = d^2$$

$$s = \frac{d}{\sqrt{2}} = \frac{2R}{\sqrt{2}}$$

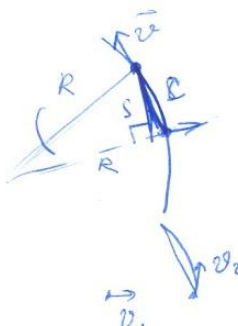
$$\Delta t \approx \frac{s}{v} = 1,9 \times 10^{-4} \text{ с.}$$

$$v \cdot \Delta t = 0,6 \text{ м}$$

~~причем~~

задержка $\approx 2 \Delta t$ в 0,4 мс для передачи
данных незначительна

6400



$$a_{yc} = \frac{v^2}{R} = \omega^2 R$$

$$F = ma = G \frac{Mm}{R}$$

$$v = \omega R$$

$$\Delta = \omega \cdot t$$

$$\omega = \frac{v}{R}$$

$$\frac{v^2}{R} = G \frac{M}{R^2}$$

$$\frac{20 \text{ km}}{v_1} \cdot t = t \quad v = 3 \text{ km/s}$$

$$\frac{20 \text{ km}}{v_1} \cdot v_2 = 8$$

$$S = 2\pi R \cdot d$$

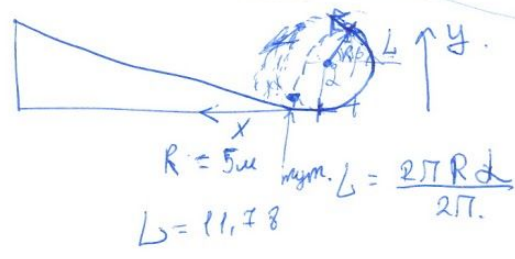
$$S^2 = 2R^2(1 - \cos \Delta)$$

при малых Δ : $C = S$

$$G = 6,67 \cdot 10^{-11} \text{ м}^3/\text{с}^2$$

$$R = 6371 \text{ км}$$

$$M = 5,97 \cdot 10^{24} \text{ кг}$$



$$L = 2\pi R \cdot \frac{\Delta}{2\pi}$$

$$L = 11,78$$

$$\Delta = 2,35 \text{ рад}$$

$$\beta = 0,7792 = \Delta - \frac{\pi}{2}$$

$$F = ma = m \frac{v^2}{R} = mg \cdot \sin \beta$$

$$v = \sqrt{35} = 5,927 \text{ м/с}$$

$$v_x = v \cdot \sin \beta = 4,1649 \text{ м/с}$$

$$v_y = v \cdot \cos \beta = 4,2169 \text{ м/с}$$

$$x_0 = -R \cdot \cos \beta = -3,557 \text{ м}$$

$$y_0 = R + R \cdot \sin \beta = 8,5135 \text{ м}$$

$$\begin{cases} x = x_0 + v_x \cdot t \\ y = y_0 + v_y \cdot t - \frac{g t^2}{2} \end{cases}$$

$$x^2 + (y - R)^2 = R^2$$

найдем время



$$x^2 + y^2 = R^2$$

$$x = (R - 20) + v \cdot t$$

$$y = \cos(\omega \cdot t) \cdot R$$

$$y = \sin(\omega \cdot t) \cdot R$$

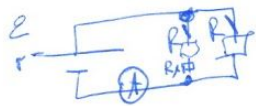
$$S = \sqrt{\Delta x^2 + y^2} =$$

$$= \sqrt{((R - 20) + v \cdot t - \cos(\omega \cdot t) \cdot R)^2 + (\sin(\omega \cdot t) \cdot R)^2}$$

= min.

$$\frac{dS}{dt} = 0 \Rightarrow \min S \text{ при } \min S^2 \Rightarrow$$

$$S^2_t = 2((R - 20) + v \cdot t - \cos(\omega t) \cdot R) \cdot (v + \sin(\omega t) \cdot R \cdot \omega) + 2(\sin(\omega t) \cdot R) \cdot R \cos(\omega t) \omega = 0$$



$$\begin{cases} I_p = \frac{E}{r+R} \\ I_p \cdot 0,18 = \frac{E}{R+r+R_x} \\ I_p \cdot 1,149 = \frac{E}{\frac{R^2+R_x R}{2R+R_x} + r} \end{cases}$$

$$\frac{0,18}{r+R} = \frac{1}{R+R_x} \quad \frac{1}{(R+R_x)} + \frac{1}{R} = \frac{1}{\left(\frac{R+R+R_x}{(R+R_x)R}\right)}$$

$$\frac{r+R}{0,18} = R+R_x+r \quad \frac{(R+R_x)R + r(2R+R_x)}{(2R+R_x)}$$

$$\frac{1,149}{r+R} = \frac{R^2 + R_x R + 2Rr + R_x r}{2R + R_x}$$

$$R = 100 \Omega$$

$$\left(\frac{r+R}{0,18} - r\right) R$$

$$r+R = 0,18R + 0,18R_x + 0,18r$$

$$0,82r + 0,82R = 0,18R_x \quad r = \frac{0,18R_x - 0,82R}{0,82} = \frac{9}{41}R_x - R$$

$$0,82(r+R) = 0,18R_x$$

$$\frac{1,149 \cdot 0,82}{0,18R_x} = \frac{R^2 + R_x R + 2R \cdot \frac{9}{41}R_x - 2R^2 + (-R^2 + R_x(R + 2R \cdot \frac{9}{41} - R))}{2R + R_x}$$

$$0,94218 \cdot 2R + 0R_x = R_x \cdot 0,18 \left(R_x \cdot \frac{180}{41} - 100 - R_x^2 \frac{9}{41} \right)$$

$$-R_x^3 \cdot \frac{81}{2050} + R_x^2 \cdot \frac{162}{205} - 18,94218R_x - 9,4218 = 0$$

$R_x > 0$.
 найдем R_x в переменной x .

Информатика

Задача 1.1

Код программы на языке C++:

```
1    #include<iostream>
2    #include<vector>
3    #include<algorithm>
4    #include<set>
5
6    using namespace std;
7
8    int main()
9    {
10        vector<int> a;
11        int n,k,m;
12        cin >> n >> k >> m ;
13        for(int i = 0; i < n; i++)
14        {
15            int tmp;
16            cin >> tmp;
17            a.push_back(tmp);
18        }
19        a[0]=10000;
20        if(m <= k+1 || (n-k+1) <= m)
21        {
22            cout << a[m-1] << endl;
23        }
24        else
25        {
26            int b=0,c=0;
27            /*
28            sort(a.begin()+1, a.begin()+m-1);
29            sort(a.begin()+m, a.end());
30            for(int i = 1; i < (m-k-1); i++)
31            {
32                b+=a[i];
33            }
34            for(int i = ((n-k+1)-m); i > 0; i--)
35            {
36                c+=a[n-i-1];
37            }
38
39            cout << (m-k-1) << " - " << ((n-k+1)-m) <<
endl;
40            cout << b << " - " << c << endl;
41
42            for(int i = 0; i < n; i++)
43            {
```

```

44         cout << a.at(i) << " ";
45     }
46     */
47     set<int> l;
48     set<int> r;
49     for(int i = 1; i <= k+1; i++)
50     {
51         l.insert(a[i]);
52     }
53     b+=*l.begin();
54     l.erase(l.begin());
55     for(int i = 1; i < (m-k-1); i++)
56     {
57         l.insert(a[k+1+i]);
58         b+=*l.begin();
59         l.erase(l.begin());
60     }
61
62     for(int i = n-1; i >= n-k ; i--)
63     {
64         r.insert(a[i]);
65     }
66     c+=*r.begin();
67     r.erase(r.begin());
68
69     for(int i = ((n-k+1)-m)-1; i > 0; i--)
70     {
71         r.insert(a[n-k-i+2]);
72         c+=*r.begin();
73         //cout<<*r.begin()<< endl;
74         r.erase(r.begin());
75     }
76     /*
77     cout << (m-k-1) << " - " << ((n-k+1)-m) <<
endl;
78     cout << b << " - " << c << endl;
79
80     for(int i = 0; i < n; i++)
81     {
82         cout << a.at(i) << " ";
83     }
84     */
85     cout << min(b,c) + a[m-1] << endl;
86 }
87 return 0;
88 }

```

Ошибка при выполнении теста №7 (неверный ответ).

Командная часть

Результаты были получены в рамках выступления команды: Бритые ежики

Личный состав команды:

- Дунаев Алексей
- Глуговская Юлия
- Кочетова Эвелина
- Кузнецова Анна

Фото команды с собранными устройствами:



Рис.1 - Команда с четырьмя устройствами в сборке

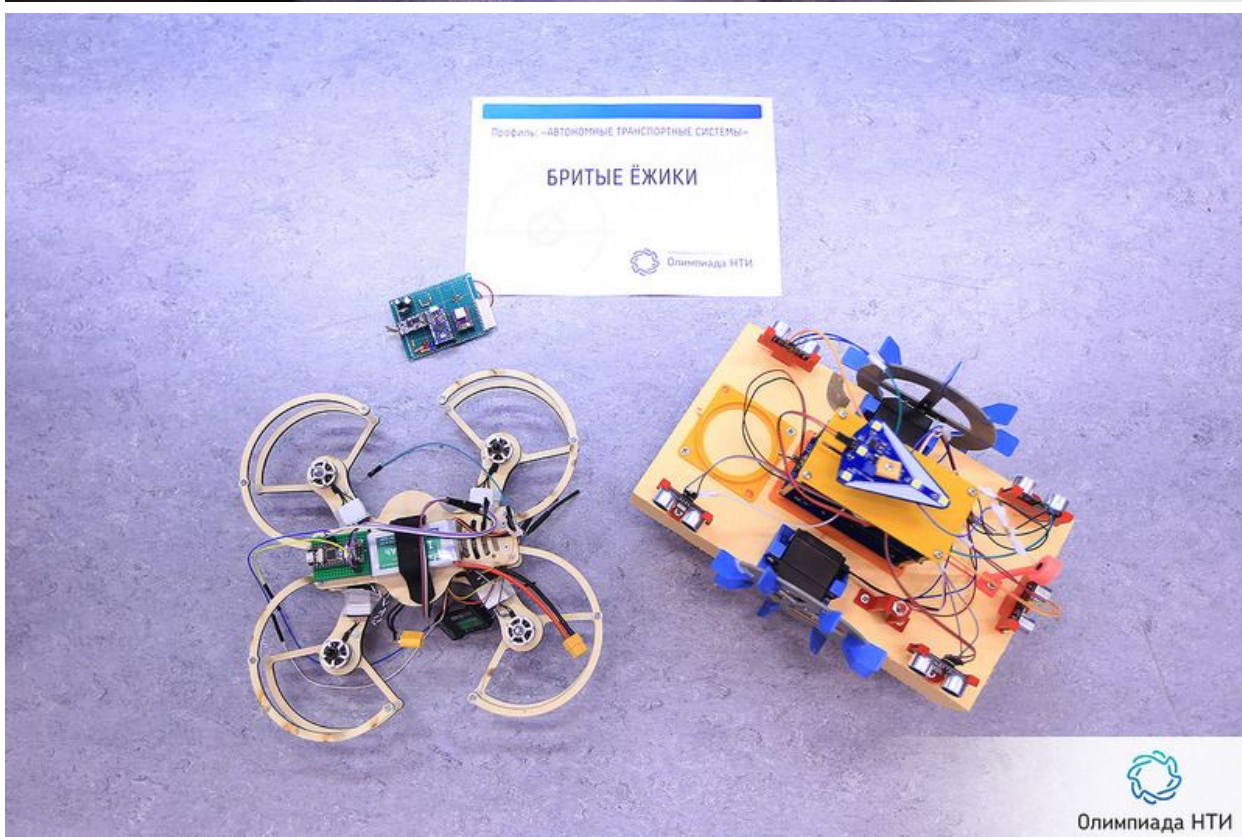
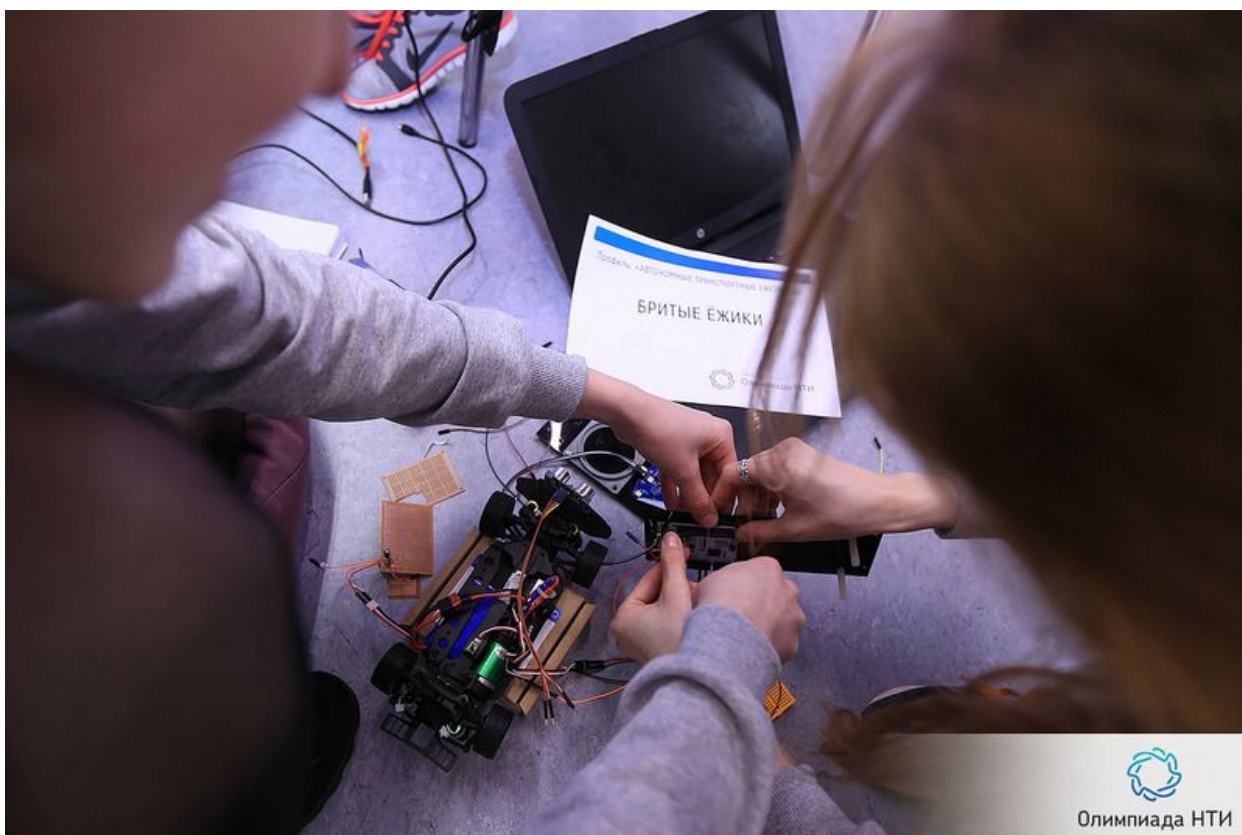


Рис.2-3 - Собранные устройства

Баллы командного этапа:

Название команды	МАРИНЕТ	АВТОНЕТ	АЭРОНЕТ	СПЕЙСНЕТ	КОМПЛЕКС	ИТОГО
Бритые ёжики	14	15,5	4	29,5		63

Маринет

Название команды	Участники		№ подзадач	База	Номер попытки										лучшая
					1	2	3	4	5	6	7	8	9	10	
Бритые ёжики	Глуговская	Юлия	1	9	9										9
	Кочетова	Эвелина	2.	5	5										5
	Кузнецова	Анна	3.	5											0
	Дунаев	Алексей	4.	17											0
			5.	5											0
			6.	9											0
РЕЗУЛЬТАТ															14

Автонет

Название команды	Участники		№ подзадач	База	Номер попытки										лучшая
					1	2	3	4	5	6	7	8	9	10	
Бритые ёжики	Глуговская	Юлия	1	10	10										10
	Кочетова	Эвелина	2.	2	0	1,5									1,5
	Кузнецова	Анна	3.	4	4										4
	Дунаев	Алексей	4.	4											0
			5.	2											0
			6.	8											0
			7.	20											0
РЕЗУЛЬТАТ															15,5

Аэронет

Название команды	Участники		№ подзадач	База	Номер попытки										лучшая
					1	2	3	4	5	6	7	8	9	10	
Бритые ёжики	Глуговская	Юлия	1	10	4										4
	Кочетова	Эвелина	2.	8											0
	Кузнецова	Анна	3.	12											0
	Дунаев	Алексей	4.	15											0
			5.	10											0
			6.	5											0
РЕЗУЛЬТАТ															4

Спейснет

Название команды	Участники		№ подзадач	База	Номер попытки										лучшая
					1	2	3	4	5	6	7	8	9	10	
Бритые ёжики	Глуговская	Юлия	1	10	10										10
	Кочетова	Эвелина	2.	5	0	4,5									4,5
	Кузнецова	Анна	3.	15	15										15
	Дунаев Алексей														0
															0
															0
РЕЗУЛЬТАТ															29,5

Комплексный проезд

Не приступали.

Решения командного этапа:

Маринет

Код программы на языке C++, обеспечивающий решение задач подтрека:

Zadaniye12.ino

```
#include <Servo.h>
#include <NewPing.h>

#define MAX_DISTANCE 200
#define HC12_SetupPin 4
#define GEO_CHANEL "AT+C001"
#define SAT_CHANEL "AT+C010"

Servo servoLeft, servoRight;

struct GEO {
    int x;
    int y;
    int grad;
};

GEO geo_data;

/*порты для оборудования*/
// Кнопка старта
int Button = 0;
const int ButtonPin = A5;

//передний дальномер
const int FFtrig = 6;
const int FFecho = 6;
NewPing sonarFF(FFtrig, FFecho, MAX_DISTANCE);

//передний правый и задний правый дальнометры
const int FRtrig = 12;
const int FRecho = 12;
const int RRtrig = 8;
const int RRecho = 8;
NewPing sonarFR(FRtrig, FRecho, MAX_DISTANCE);
NewPing sonarRR(RRtrig, RRecho, MAX_DISTANCE);

//передний правый и задний правый дальнометры
const int FLtrig = 11;
const int FLecho = 11;
const int RLtrig = 7;
const int RLecho = 7;
```

```
NewPing sonarFL(FLtrig,FLecho, MAX_DISTANCE);
NewPing sonarRL(RLtrig,RLecho, MAX_DISTANCE);
```

```
float distFF;
float distFL;
float distFR;
float distRL;
float distRR;
float Light;
float Voltage;
float vLeft;
float vRight;
```

```
//светодатчики левый и правый
const int ir_pin = A4;
const int volt_pin = A0;
const int led_pin = 5;
```

```
const int L_wheel = 9;
const int R_wheel = 10;
```

```
const int v = 10; //минимальна скорость
```

```
String str;
```

```
void setup()
{
```

```
    //сервопривод
    servo_attach();
```

```
    pinMode(ir_pin, INPUT);
    pinMode(volt_pin, INPUT);
    pinMode(led_pin, OUTPUT);
    pinMode(ButtonPin, INPUT);
```

```
    Serial.begin(9600);
    Serial1.begin(9600);
    Serial1.flush();
```

```
    pinMode(HC12_SetupPin, OUTPUT);
    digitalWrite(HC12_SetupPin, HIGH);
```

```
    delay(3000);
    digitalWrite(led_pin,HIGH);
    delay(3000);
    set_transmitter_chanel(GEO_CHANEL);
```

```

}

void loop()

{
    while (Button < 500) {
        Button = analogRead(ButtonPin);
        delay(50);
    }

    int flag=0;

    SensorScan();
    if(distFF<13 && distFF!=0)
    {
        if(distFL<13 && distFL!=0) setServo(50, v);
        if(distFR<13 && distFR!=0) setServo(v, 50);
        if((distFL>13 || distFL==0 ) && (distFR>13 || distFR==0 ))
setServo(50, v);
        delay(3000);
    }
    //read_geo_data(); - ждет наличия конкретных данных
/* delay(3000);
    setServo(50, v);
    delay(1000);
    setServo(v,v);*/
    digitalWrite(led_pin,LOW);
    delay(500);

}

float setServo(float vLeft, float vRight)
{
    if (vLeft > 80) { vLeft = 80; }
    if (vRight > 80) { vRight = 80;}
    servoLeft.write(90 - vLeft);
    servoRight.write(90 + vRight);

}

float SensorScan ()
{

    delay(2);
    distFF = sonarFF.ping_cm();

```

```

Serial.print("FF:");
Serial.println(distFF);

delay(2);
distFL = sonarFL.ping_cm();
Serial.print("FL:");
Serial.println(distFL);

delay(2);
distRL = sonarRL.ping_cm();
Serial.print("RL:");
Serial.println(distRL);

delay(2);
distFR = sonarFR.ping_cm();
Serial.print("FR:");
Serial.println(distFR);

delay(2);
distRR = sonarRR.ping_cm();
Serial.print("RR:");
Serial.println(distRR);

Light = analogRead(ir_pin);
Serial.print("IR:");
Serial.println(Light);

Voltage = analogRead(volt_pin);
//Voltage = analogRead(volt_pin);
Serial.print("Voltage:");
Serial.println(Voltage*10.00/1023.00);

Serial.println("---");

}

void servo_detach() {

    servoLeft.detach();
    servoRight.detach();
}

void servo_attach() {

    servoLeft.attach(L_wheel);
    servoRight.attach(R_wheel);
    servo_stop();
}

```

```

void servo_stop() {

    servoLeft.write(90);
    servoRight.write(90);
}

GEO read_geo_data() {

    while (1) {

        if (Serial1.available() > 0) {

            str = Serial1.readStringUntil('\n');
            //str = "00;1019;0609;192";
            Serial.println(str);

            geo_data.x      = str.substring(3, 7).toInt();
            geo_data.y      = str.substring(8, 12).toInt();
            geo_data.grad = str.substring(13, 16).toInt();

            return geo_data;
        }
    }
}

void set_transmitter_chanel(char* set_chanel) {

    digitalWrite(HC12_SetupPin, LOW);
    delay(50);
    Send_command(set_chanel);
    digitalWrite(HC12_SetupPin, HIGH);

}

void Send_command(char* ATcommand) {

    Serial1.println (ATcommand);
    Serial.println (ATcommand);
    delay(200);
    if (Serial1.available() > 0)
    {
        str = Serial1.readStringUntil('\n');// LF
        Serial.println (str);
    }
    delay(100);
}

```

Автонет

Код программы на языке C++, обеспечивающий решение задач подтрека:

Zadaniye12.ino

```
#include <Servo.h>
Servo myservo;
#define Trig 9
#define Echo 8

void setup() {
    myservo.attach(5);
    //myservo.write(90);

    pinMode(Trig, OUTPUT); //инициируем как выход
    pinMode(Echo, INPUT); //инициируем как вход
    Serial.begin(9600);
}
unsigned int impulseTime = 0;
unsigned int distance_sm = 0;
void loop() {
    myservo.write(90);
    digitalWrite(Trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(Trig, LOW);
    impulseTime = pulseIn(Echo, HIGH);
    distance_sm = impulseTime / 58;
    Serial.println(distance_sm);

    if (distance_sm < 30) {
        myservo.write(76);
    }

    if (distance_sm > 30) {
        myservo.write(104);
    }

    if (distance_sm == 30) {
        myservo.write(90);
    }

    delay (200);
}
```

Zadaniye3.ino

```
#include <stdint.h>
#include "irreceiver.h"
```

```

const uint8_t IR_RECEIVER_PIN = 7;

IrReceiver receiver(IR_RECEIVER_PIN);
    long tmp = a;
    int N;

const uint8_t SIGNAL_FORWARD = 0xc0;
const uint8_t SIGNAL_STOP = 0xf0;
const uint8_t SIGNAL_LEFT = 0xa0;
const uint8_t SIGNAL_RIGHT = 0x90;

void setup()
{
    Serial.begin(9600);
    receiver.init();
}

void loop()
{
    while (!receiver.isDataReceived()) {
        // wait here
    }

    uint8_t signal = receiver.getReceivedData() >> 4;

    switch (signal) {
        case SIGNAL_FORWARD:
            Serial.println("Forward!");
            break;

        case SIGNAL_LEFT:
            Serial.println("Left!");
            break;

        case SIGNAL_RIGHT:
            Serial.println("Right!");
            break;

        case SIGNAL_STOP:
            Serial.println("Stop!");
            break;

        default:
            Serial.print("Unknown signal: ");
            Serial.print(signal, HEX);
            Serial.println("!");
    }

    if (a == 0)
        N = 1;
    else
        {

```

```

N = 0;
while (tmp != 0)
{
    N++;
    tmp = tmp / 16;
}
}
bin = (char*)malloc((4 * N + 1)*sizeof(char));
for (int i = N - 1; i >= 0; i--)
{
    switch (a % 16)
    {
        case 0:
        {
            bin[4 * i] = '0';
            bin[4 * i + 1] = '0';
            bin[4 * i + 2] = '0';
            bin[4 * i + 3] = '0';
            break;
        }
        case 1:
        {
            bin[4 * i] = '0';
            bin[4 * i + 1] = '0';
            bin[4 * i + 2] = '0';
            bin[4 * i + 3] = '1';
            break;
        }
        case 2:
        {
            bin[4 * i] = '0';
            bin[4 * i + 1] = '0';
            bin[4 * i + 2] = '1';
            bin[4 * i + 3] = '0';
            break;
        }
        case 3:
        {
            bin[4 * i] = '0';
            bin[4 * i + 1] = '0';
            bin[4 * i + 2] = '1';
            bin[4 * i + 3] = '1';
            break;
        }
        case 4:
        {
            bin[4 * i] = '0';
            bin[4 * i + 1] = '1';
            bin[4 * i + 2] = '0';
            bin[4 * i + 3] = '0';
            break;
        }
    }
}

```

```

}
case 5:
{
    bin[4 * i] = '0';
    bin[4 * i + 1] = '1';
    bin[4 * i + 2] = '0';
    bin[4 * i + 3] = '1';
    break;
}
case 6:
{
    bin[4 * i] = '0';
    bin[4 * i + 1] = '1';
    bin[4 * i + 2] = '1';
    bin[4 * i + 3] = '0';
    break;
}
case 7:
{
    bin[4 * i] = '0';
    bin[4 * i + 1] = '1';
    bin[4 * i + 2] = '1';
    bin[4 * i + 3] = '1';
    break;
}
case 8:
{
    bin[4 * i] = '1';
    bin[4 * i + 1] = '0';
    bin[4 * i + 2] = '0';
    bin[4 * i + 3] = '0';
    break;
}
case 9:
{
    bin[4 * i] = '1';
    bin[4 * i + 1] = '0';
    bin[4 * i + 2] = '0';
    bin[4 * i + 3] = '1';
    break;
}
case 10:
{
    bin[4 * i] = '1';
    bin[4 * i + 1] = '0';
    bin[4 * i + 2] = '1';
    bin[4 * i + 3] = '0';
    break;
}
case 11:
{

```

```

        bin[4 * i] = '1';
        bin[4 * i + 1] = '0';
        bin[4 * i + 2] = '1';
        bin[4 * i + 3] = '1';
        break;
    }
    case 12:
    {
        bin[4 * i] = '1';
        bin[4 * i + 1] = '1';
        bin[4 * i + 2] = '0';
        bin[4 * i + 3] = '0';
        break;
    }
    case 13:
    {
        bin[4 * i] = '1';
        bin[4 * i + 1] = '1';
        bin[4 * i + 2] = '0';
        bin[4 * i + 3] = '1';
        break;
    }
    case 14:
    {
        bin[4 * i] = '1';
        bin[4 * i + 1] = '1';
        bin[4 * i + 2] = '1';
        bin[4 * i + 3] = '0';
        break;
    }
    case 15:
    {
        bin[4 * i] = '1';
        bin[4 * i + 1] = '1';
        bin[4 * i + 2] = '1';
        bin[4 * i + 3] = '1';
        break;
    }
    }
    a = a / 16;
}
bin[4 * N] = '\0';
return bin;
}
}

```

Irreceiver.cpp

```
#include "irreceiver.h"
```

```
#include <Arduino.h>
```

```

const unsigned int START_TIMEOUT = 2500;
const unsigned int DATA_STAGE_1_TIMEOUT = 1200;
const unsigned int DATA_STAGE_2_TIMEOUT = 900;
const unsigned int CHECKING_TIMEOUT = 10000;

static IrReceiver *active_receiver = NULL;

static void initTimer()
{
    TCCR4A = 0;
    TCCR4B = 0;
    TCCR4C = 0;
    TCCR4D = 0;
    TCCR4E = 0;
}

static void startTimer(unsigned int us)
{
    TC4H = 0;
    TCNT4 = 0;
    OCR4C = microsecondsToClockCycles(us) / 1024;
    TCCR4B = (1 << PSR4) |
              (1 << CS43) | (0 << CS42) |
              (1 << CS41) | (1 << CS40);
    TIMSK4 = (1 << TOIE4);
}

static void stopTimer()
{
    TIMSK4 = 0;
    TCCR4B = 0;
}

static void edgeHandler()
{
    if (active_receiver != NULL) {
        active_receiver->handleEdge();
    }
}

ISR(TIMER4_OVF_vect)
{
    if (active_receiver != NULL) {
        active_receiver->handleTimeout();
    }
}

void IrReceiver::init()
{
    _state = STATE_BEGIN;
}

```

```

    initTimer();

    active_receiver = this;

    pinMode(_pin, INPUT);

    attachInterrupt(digitalPinToInterrupt(_pin),
                    edgeHandler,
                    FALLING);
}

void IrReceiver::handleEdge()
{
    stopTimer();

    if (_state == STATE_BEGIN) {
        startTimer(START_TIMEOUT);

        _state = STATE_START;
        return;
    }

    if (_state == STATE_START) {
        _state = STATE_BEGIN;
        return;
    }

    if (_state == STATE_DATA_STAGE_1) {
        startTimer(DATA_STAGE_2_TIMEOUT);

        _state = STATE_DATA_STAGE_2;
        return;
    }

    if (_state == STATE_DATA_STAGE_2) {
        _state = STATE_BEGIN;
        return;
    }

    if (_state == STATE_CHECKING) {
        _state = STATE_BEGIN;
        return;
    }
}

void IrReceiver::handleTimeout()
{
    stopTimer();

    if (_state == STATE_START) {
        if (digitalRead(_pin) == HIGH) {

```

```

        startTimer(DATA_STAGE_1_TIMEOUT);
        _data = 0;
        _bit_counter = 12;

        _state = STATE_DATA_STAGE_1;
    } else {
        _state = STATE_BEGIN;
    }
    return;
}

if (_state == STATE_DATA_STAGE_1) {
    _state = STATE_BEGIN;
    return;
}

if (_state == STATE_DATA_STAGE_2) {
    if (digitalRead(_pin) == HIGH) {
        _data <<= 1;
    } else {
        _data <<= 1;
        _data |= 1;
    }

    _bit_counter--;

    if (_bit_counter > 0) {
        startTimer(DATA_STAGE_1_TIMEOUT);

        _state = STATE_DATA_STAGE_1;
    } else {
        startTimer(CHECKING_TIMEOUT);

        _state = STATE_CHECKING;
    }
    return;
}

if (_state == STATE_CHECKING) {
    _state = STATE_RECEIVED;
    return;
}
}

bool IrReceiver::isDataReceived()
{
    cli();
    State state = _state;
    sei();
    return (state == STATE_RECEIVED);
}

```

```

uint16_t IrReceiver::getReceivedData()
{
    if (isDataReceived()) {
        cli();
        uint16_t data = _data;
        _state = STATE_BEGIN;
        sei();
        return data;
    } else {
        return 0;
    }
}

```

Irreceiver.h

```

#pragma once

#include <stdint.h>

class IrReceiver
{
public:
    IrReceiver(uint8_t pin) :
        _pin(pin),
        _state(STATE_BEGIN),
        _data(0),
        _bit_counter(0)
    {
    }

    void init();

    void handleEdge();

    void handleTimeout();

    bool isDataReceived();

    uint16_t getReceivedData();

private:
    enum State
    {
        STATE_BEGIN,
        STATE_START,
        STATE_DATA_STAGE_1,
        STATE_DATA_STAGE_2,
        STATE_CHECKING,
        STATE_RECEIVED
    };
};

```

```

private:
    uint8_t _pin;
    volatile State _state;
    volatile uint16_t _data;
    volatile int _bit_counter;
};

```

Аэронет

Код программы отсутствует; команда собрала и получила зачет только за собранную конструкцию коптера.

Спейснет

Код программы на языке C++, обеспечивающий решение задач подтрека:

```

#define STEP_PIN 10
#define DIR_PIN 16
#define POWER_ENABLE_PIN 14
#define END_SWITCH1 3
#define END_SWITCH2 2

#define LEFT_DIR 1
#define RIGHT_DIR 0
#define SIZE_K 0.02

#define LED1 8
#define LED2 9

#define STEP_IN_ROTATION 200

#define GEO_CHANEL "AT+C056"
#define SAT_CHANEL "AT+C010"

struct GEO {
    int x;
    int y;
    int grad;
};

int position = 0;
int Xmax = 1000;
int Ymax = 1000;
int gest = 50;

GEO geo_data;
String str;
int stop_flag = 0;

```

```

const int volt_pin = A0;
float Voltage;
int dir, rotation;
long steps, i;

GEO read_geo_data() {

    while (1) {

        if (Serial1.available() > 0) {

            str = Serial1.readStringUntil('\n');
            //str = "00;1019;0609;192";
            Serial.println(str);

            geo_data.x    = str.substring(3, 7).toInt();
            geo_data.y    = str.substring(8, 12).toInt();
            geo_data.grad = str.substring(13, 16).toInt();

            return geo_data;
        }
        //delay(500);
    }
}

void set_transmitter_chanel(char* set_chanel) {

    digitalWrite(4, LOW);
    delay(100);
    Send_command(set_chanel);
    digitalWrite(4, HIGH);

}

void Send_command(char* ATcommand) {

    Serial1.println (ATcommand);
    Serial.println (ATcommand);
    delay(400);
    if (Serial1.available() > 0)
    {
        str = Serial1.readStringUntil('\n');// LF
        Serial.println (str);
    }
    delay(100);
}

void setup() {
    // put your setup code here, to run once:
    Serial1.begin(9600);

```

```

Serial.begin(9600);
delay(2000);
pinMode(4, OUTPUT);
pinMode(STEP_PIN, OUTPUT);
pinMode(DIR_PIN, OUTPUT);
pinMode(POWER_ENABLE_PIN, OUTPUT);
pinMode(LED1, OUTPUT);
pinMode(LED2, OUTPUT);

digitalWrite(POWER_ENABLE_PIN, HIGH);
digitalWrite(LED1, HIGH);
digitalWrite(LED2, HIGH);
delay(1000);
digitalWrite(LED1, LOW);
digitalWrite(LED2, LOW);
set_transmitter_chanel(GEO_CHANEL);
move(LEFT_DIR, 8);
delay(1000);
}

void loop() {
  Serial.println(" Stert ");
  GEO_data = read_geo_data();

  rotation = 0;
  if (data.y > Xmax / 2 + gest)
  {
    rotation = -gest * SIZE_K;
  }
  if (data.y < Xmax / 2 - gest)
  {
    rotation = gest * SIZE_K;
  }

  if (rotation > 0) {
    move(RIGHT_DIR, rotation);
  }
  if (rotation < 0) {
    move(LEFT_DIR, abs(rotation));
  }

  Serial.println(rotation);
  delay(40);
  /*
    Voltage = analogRead(volt_pin);
    Serial.print("Voltage:");
    Serial.println(Voltage*15.00/1023.00);

    Serial.println("---");
  */
}

```

```

void move(int dir, unsigned int rotation) {

    digitalWrite(POWER_ENABLE_PIN, LOW);
    digitalWrite(DIR_PIN, dir);

    steps = rotation * STEP_IN_ROTATION;

    doSteps(steps);

    digitalWrite(POWER_ENABLE_PIN, HIGH);

}

boolean isEndSwitch() {

    if (digitalRead(END_SWITCH1) == 1) {
        //Serial1.println("END_SWITCH1 (Start)");
        rollBack(RIGHT_DIR);
        return true;
    }

    if (digitalRead(END_SWITCH2) == 1) {
        //Serial1.println("END_SWITCH2 (Finish)");
        rollBack(LEFT_DIR);
        return true;
    }

    return false;
}

boolean isStopInSerial() {

    if (Serial1.available() > 0) {
        if (Serial1.readStringUntil('\n').toInt() == 0) {
            return true;
        }
    }

    return false;
}

void rollBack(int dir) {

    digitalWrite(DIR_PIN, dir);
    for (i = 0; i <= 100; i++) {
        doStep();
    }
}

```

```
void doSteps(long steps) {  
  
    for (i = 0; i <= steps; i++) {  
  
        if (isEndSwitch()) {  
            return;  
        }  
  
        if (isStopInSerial()) {  
            return;  
        }  
  
        doStep();  
  
    }  
  
    //Serial1.println("STOP");  
  
}  
  
void doStep() {  
  
    digitalWrite(STEP_PIN, HIGH);  
    delayMicroseconds(800);  
    digitalWrite(STEP_PIN, LOW);  
    delayMicroseconds(800);  
  
}
```