

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

Профиль «АВТОНОМНЫЕ ТРАНСПОРТНЫЕ СИСТЕМЫ»

Ястребов Вячеслав Дмитриевич

Класс: 11

Город: Уфа

Школа: МБОУ Лицей № 153

Регион: Республика
Башкортостан

Уникальный номер участника:644

**Команда на заключительном
этапе:** Void song

Результаты заключительного этапа:

№	Индивидуальный этап												Командный этап		ИТОГ
	Физика				Информатика				За задачи	Коэффициент 40%	баллы	Коэффициент 60%			
644	10	0	10	0	0	0	0	0	0	0	20	5,333,333	58.5	14.04	19.373

Индивидуальная часть

Персональный лист участника с номером 644:



Олимпиада НТИ

ФИО Ястребов Вячеслав Викторович

Город Уфа

Школа № 153

Физика

Командная инженерная олимпиада «Олимпиада НТИ»

Направление АТЕ

Предмет Физика

Номер участника 644

1	2	3	4	Σ
10	0	10	0	20

1. Дано:

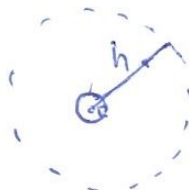
$$L = 20 \text{ км}$$

$$v_p = 1,77 \text{ км/с}$$

$$h = 35786 \text{ км}$$

$g = ?$

Решение:



$$G = 6,67 \cdot 10^{-11} \frac{\text{Н} \cdot \text{м}^2}{\text{кг}^2}$$

$$M_3 = 5,97 \cdot 10^{24} \text{ кг}$$

$$R_3 = 6371 \text{ км}$$

$$1. a_3 = \frac{v^2}{R_3 + h} = G \frac{M_3}{(R_3 + h)^2} \Rightarrow v_e = \sqrt{G \frac{M_3}{R_3 + h}} - \text{скорость спутника}$$

$$v_e \approx 3,1 \text{ км/с} \quad +4$$

2. Время, необходимое ракете для достижения точки, в которой находится в начальный момент времени спутник.

$$t = \frac{L}{v_p} \approx 11,3 \text{ с.}$$

10

3. Скорее в данной узкой траектории полета спутника можно считать прямой (так как $v_e t = 35 \text{ км} \ll 2\pi(R_3 + h)$) и эта прямая перпендикулярна прямой, по которой движется ракета (скорость спутника - касательная к орбите) $\Rightarrow$ минимальное расстояние

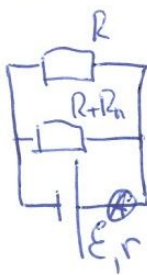
$$v_e t = 35 \text{ км}$$

+2.

Ответ. 35 км.

3. Dano,
 $R = 100 \Omega$
 $A_1 = 18\%$
 $A_2 = 14,9\%$
 $R_n = ?$

Решение,



(10)

$$1. \quad I_1 = \frac{E}{R+r} \quad I_2 = \frac{E}{R+r+R_n} \Rightarrow 0,82 I_1 < I_2 < I_1$$

$$I_3 = \frac{E}{r + \frac{R^2 + R_n R}{R_n + 2R}} \quad R_{\text{общ}} = \frac{R \cdot (R + R_n)}{R_n + 2R} \quad (\text{прибор соединен параллельно})$$

$$0,82 I_1 < I_3 < I_1 \quad I_1 < I_3 < 1,149 I_1$$

$$2. \quad 0,82 \frac{1}{R+r} < \frac{1}{R+r+R_n} < \frac{1}{R+r} \quad \text{или} \quad \frac{1}{R+r} < \frac{1}{r + \frac{R^2 + R_n R}{R_n + 2R}} < \frac{1,149}{R+r}$$

$$\left\{ \begin{array}{l} R+r < R+r+R_n, \\ R+r+R_n < \frac{R+r}{0,82}, \\ r + \frac{R^2 + R_n R}{R_n + 2R} > \frac{R+r}{1,149}, \\ R+r > r + \frac{R^2 + R_n R}{R_n + 2R}, \end{array} \right.$$

$$\Rightarrow \left\{ \begin{array}{l} 0 < R_n < 0,22(R+r), \\ R_n + 2R < R + R_n, \\ 1,149(r R_n + 2R + R^2 + R_n R) > R_n R + 2R + r R_n, \\ 0,149 R_n + 0,288 R + 0,149 R_n R - 0,851 R^2 > 0 \end{array} \right. \Rightarrow$$

+2.

Командная инженерная олимпиада «Олимпиада НТИ»

Направление АТС

Предмет Физика

Номер участника 644

1. Дано:
 $R_3 = 6371 \text{ км}$
 $L = 20 \text{ км}$
 $V = 1,77 \text{ км/с}$
 $G = 6,67 \cdot 10^{-11} \frac{\text{Н} \cdot \text{м}^2}{\text{кг}^2}$
 $h = 35786 \text{ км}$

Решение:



$\rho = ?$
 $M_3 = 5,97 \cdot 10^{24} \text{ кг}$

1) Время, необходимое ракете для достижения точки, в которой находится в данный момент времени:

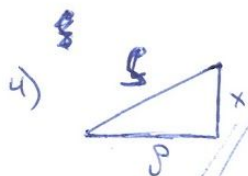
$$t = \frac{L}{V} \approx 11,3 \text{ с}$$

2) Скорость спутника:

$$V_c = \sqrt{a_{\text{центр}}(R_3 + h)} = \sqrt{G \frac{M_3}{(R_3 + h)^2} \cdot (R_3 + h)} = \sqrt{G \frac{M_3}{R_3 + h}}$$

$$= \sqrt{6,67 \cdot 10^{-11} \frac{\text{Н} \cdot \text{м}^2}{\text{кг}^2} \cdot \frac{5,97 \cdot 10^{24} \text{ кг}}{(35786 + 6371) \cdot 1000 \text{ м}}} \approx 3,13 \text{ км/с}$$

3) В данном случае траекторию полета спутника можно считать прямой, так как $3,1 \text{ км/с} \cdot 11,3 \text{ с} = 35 \text{ км} \ll$

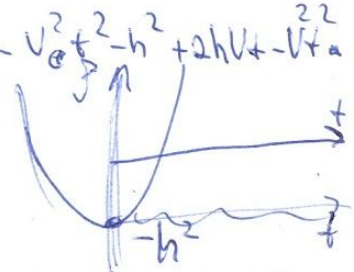


$$x = h - vt$$

$$\rho = S^2 - x^2 = (V_c t)^2 - (h - vt)^2 = V_c^2 t^2 - h^2 + 2hvt - v^2 t^2$$

$$\rho' = 2t \cdot V_c^2 + 2hV - 2t \cdot v^2 = 0 \quad t =$$

$$\rho = t^2(V_c^2 - v^2) + 2hvt - h^2$$



Командная инженерная олимпиада «Олимпиада НТИ»

Направление АТС

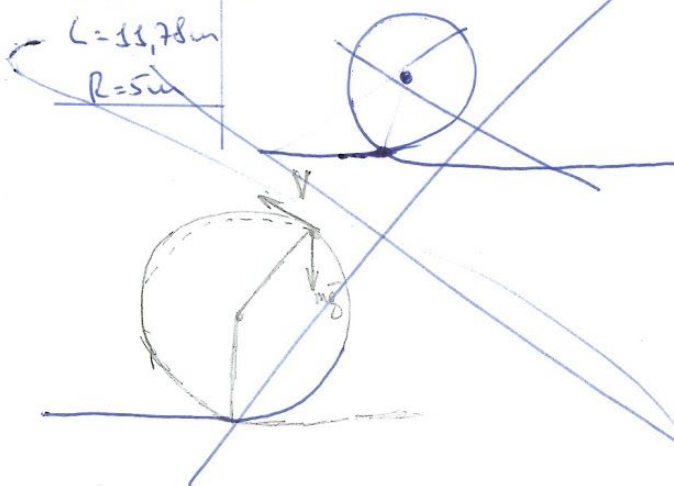
Предмет физика

Номер участника 644

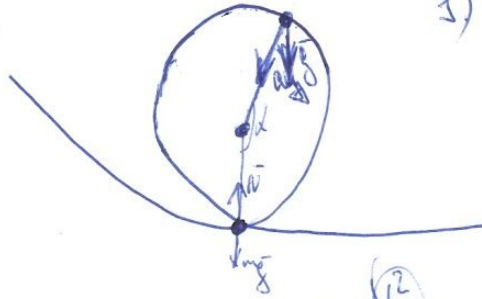
2. Дано: Решение:

$L = 11,78 \text{ м}$

$R = 5 \text{ м}$



2.

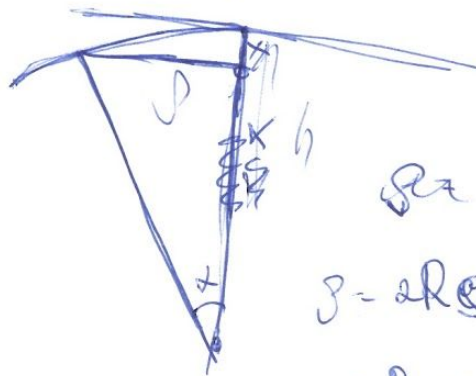


$$1) \alpha = \frac{L}{2\pi R}$$

$$\alpha = \frac{360 - 11,78}{2 \cdot 360} = \frac{360 - 11,78}{720} \approx 135^\circ$$

$$\alpha = \frac{\sqrt{2}}{2}$$

2) Total distance traveled is zero



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

$$t = \frac{h - x}{v_p}$$

$$\beta = 2R \sin \alpha$$

$$= 2R \sin \frac{v}{R} \cdot \frac{h - x}{v_p}$$

$$(2R \sin \alpha)^2 + x^2 = L^2$$

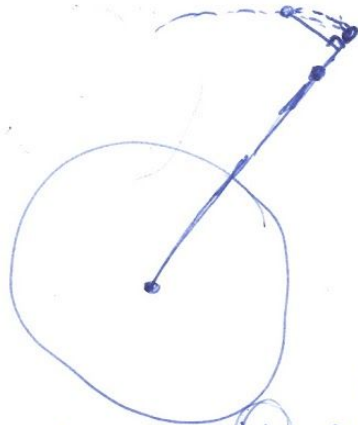
$$4R^2 \sin^2 \frac{v}{R} \cdot \frac{h - x}{v_p} + x^2 = L^2$$

$$\frac{2}{2\pi} - x = 1$$

$$+1 = 1$$

$$\gamma_{\text{Bul}} = 1 \Rightarrow \gamma_m = \frac{1}{\gamma}$$

1.



$$a_{\text{cg}} = \frac{v^2}{(R+h)} = G \frac{M_3}{(R+h)^2}$$

$$v = \sqrt{\frac{GM_3}{R+h}}$$

$$v = \omega R \rightarrow \omega = \frac{v}{R} = \frac{3.1}{35}$$

$$R_1 = R + R_A \quad R = 100 \text{ m}$$

$$\omega = \frac{L \cdot \cancel{0.025}}{2\pi R} = \frac{L \cdot \cancel{0.025}}{2\pi R t} = \frac{v \cdot \cancel{0.025}}{2\pi R} = \frac{v}{R} = \frac{1}{c}$$

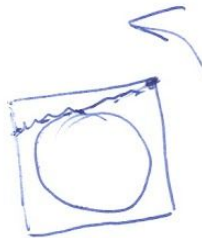
3.



$$I_P = \frac{\mathcal{E}}{r+R} = \frac{\mathcal{E}}{r+R_1}$$

11,3

11,3



$$V = a_{\text{cg}} = \frac{v^2}{R}$$

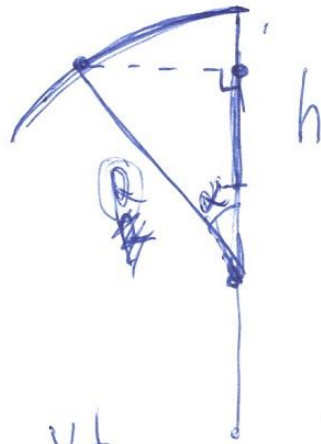
$$G = 6,67 \cdot 10^{-11}$$

$$M_3 = 5,97 \cdot 10^{24} \text{ kg}$$

$$G \frac{M_1^2}{r^2} \Leftrightarrow G \frac{M_{\text{km}}^2}{\text{km}^2}$$

$$G = 6,67$$

$$R = 6371 \text{ km}$$



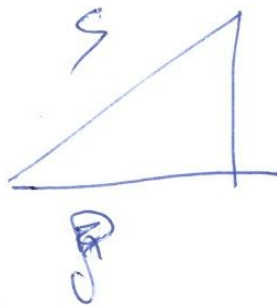
$$\omega = \frac{V}{R}$$

$$p = 2R \sin \omega t = 2R \sin \frac{Vt}{R}$$

$$p = R^2 - (R^2 - x)^2 = R^2 - (R^2 - 2Rx + x^2) = 2Rx - x^2 \quad x = h - Vt$$

$$Vt_y = h - Vt$$

$$\frac{R^2 + 2Rx - x^2 - 2R \sin \frac{Vt}{R}}{R}$$



$$h^2 - 2Vt + h + Vt^2$$

$$\frac{1}{R \cos \theta} = \frac{1}{R} + \frac{1}{R + R_n}$$

$$2 + (V_c^2 - V^2) = -2hV$$

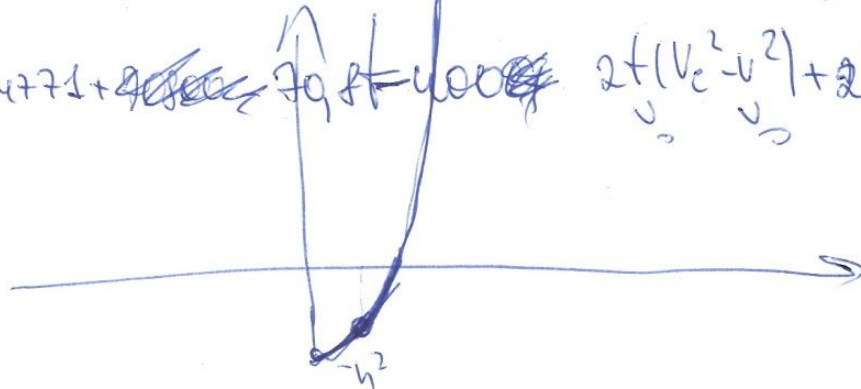
$$(Vt)^2 - (h - Vt)^2 = \frac{R + R_n + R}{R(R + R_n)}$$

$$\frac{hV}{V_c^2 - V^2} =$$

$$= V_c^2 t^2 - (h^2 - 2Vt + h + Vt^2) = t^2(V_c^2 - V^2) + 2Vt + h - h^2$$

$$V_c^2 t^2 - h^2 + 2Vt - V^2 t^2 = t^2(V_c^2 - V^2) + 2hVt - h^2$$

$$y = t^2 \left(\frac{V_c^2 - V^2}{2} + \frac{2hV}{2} - \frac{h^2}{2} \right)$$



$$0,82 \frac{\varepsilon}{R+r} < \frac{\varepsilon}{R+r+R_n} < \frac{\varepsilon}{R+r}$$

0,82 ~~1~~

$$x+y$$

$$0,82 \frac{1}{x} < \frac{1}{x+y} < \frac{1}{x}$$

~~$$x < x+y < 0,82x$$~~

$$x < x+y$$

$$x+y < \frac{1}{0,82} x$$

$$R < \frac{R^2 + R_n R}{R_n + 2R} \Rightarrow 1 < \frac{R + R_n}{R_n + 2R}$$

~~$$R < \frac{R^2 + R_n R}{R_n + 2R}$$~~

$$R_n + 2R < R + R_n$$

$$R < 0$$

~~$$1 > \frac{R + R_n}{R_n + 2R}$$~~

$$R > 0$$

$$R_n + 2R$$

$$(R_n + 2R + R^2 + R_n R) > R_n R + R_n + 2R^2 + 2R$$

Информатика

Задача 1.1

Код программы на языке C:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int n, k, m, q=0, p=0, res, i, f=0;
6      scanf("%d %d %d", &n, &k, &m);
7      int mas[n];
8      for(i=1; i<=n; i++)
9      {
10         scanf("%d", &mas[i]);
11     }
12     i=1+k;
13     if(i>=m)
14     {
15         i=m;
16         q=mas[m];
17         f=1;
18     }
19     while(f!=1)
20     {
21         q=q+mas[i];
22         i=i+k;
23         if(i>=m)
24         {
25             i=m;
26             q=q+mas[m];
27             f=1;
28         }
29     }
30     f=0;
31     i=n-k+1;
32     if(i<=m)
33     {
34         i=m;
35         p=p+mas[i];
36         f=1;
37     }
38     while(f!=1)
39     {
40         p=p+mas[i];
41         i=i-k;
42         if(i<=m)
43         {
44             i=m;
```

```
45         p=p+mas[i];
46         f=1;
47     }
48 }
49 if(q>p)
50     res=p;
51 else
52     res=q;
53 //printf("%d %d\n", q, p);
54 printf("%d", res);
55 }
```

Ошибка при выполнении теста №7 (неверный ответ) (0 баллов).

Командная часть

Результаты были получены в рамках выступления команды: Void song

Личный состав команды:

- Васильченко Богдан
- Опякин Роман
- Сатаев Эмиль
- Тихомирова Екатерина
- Ястребов Вячеслав

Фото команды с собранными устройствами:



Рис.1 - Команда с четырьмя устройствами в сборке

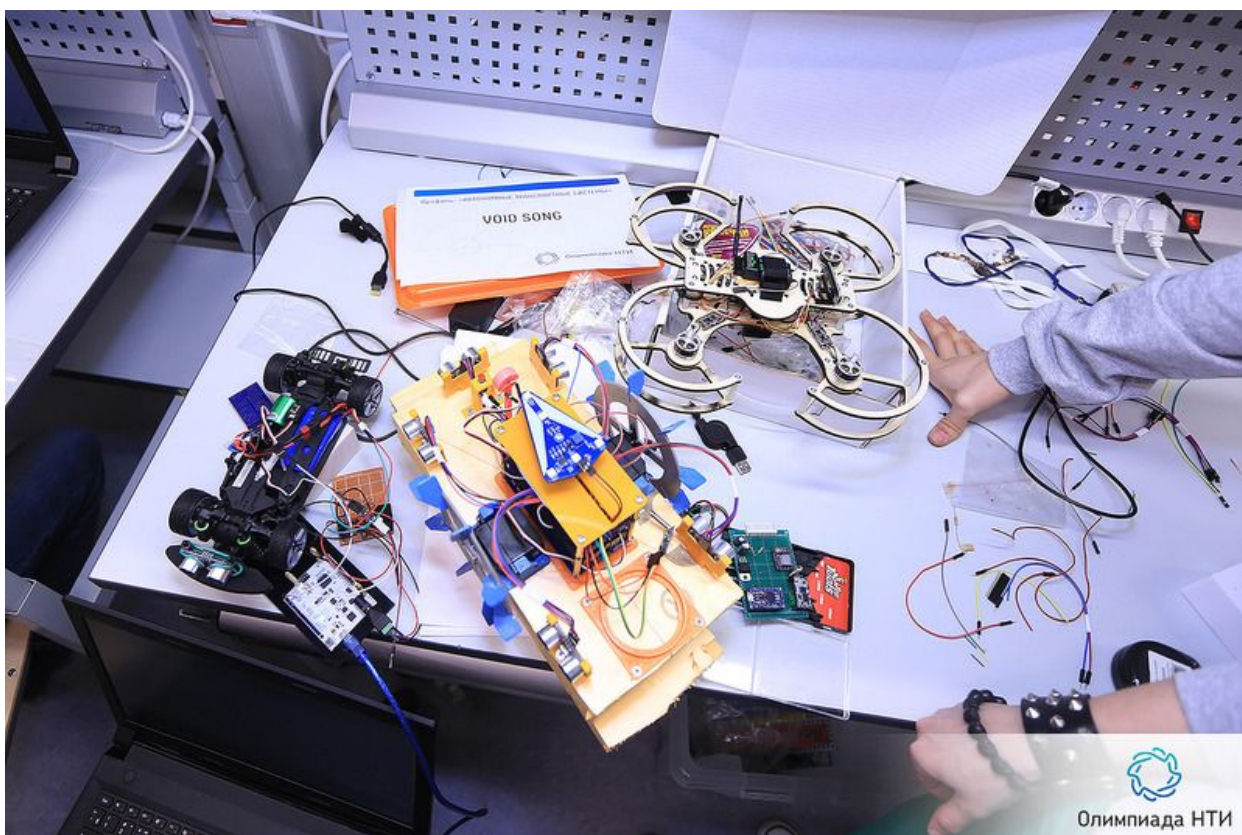


Рис.2 - Четыре устройства, собираемые командой, по состоянию на конец второго дня Олимпиады

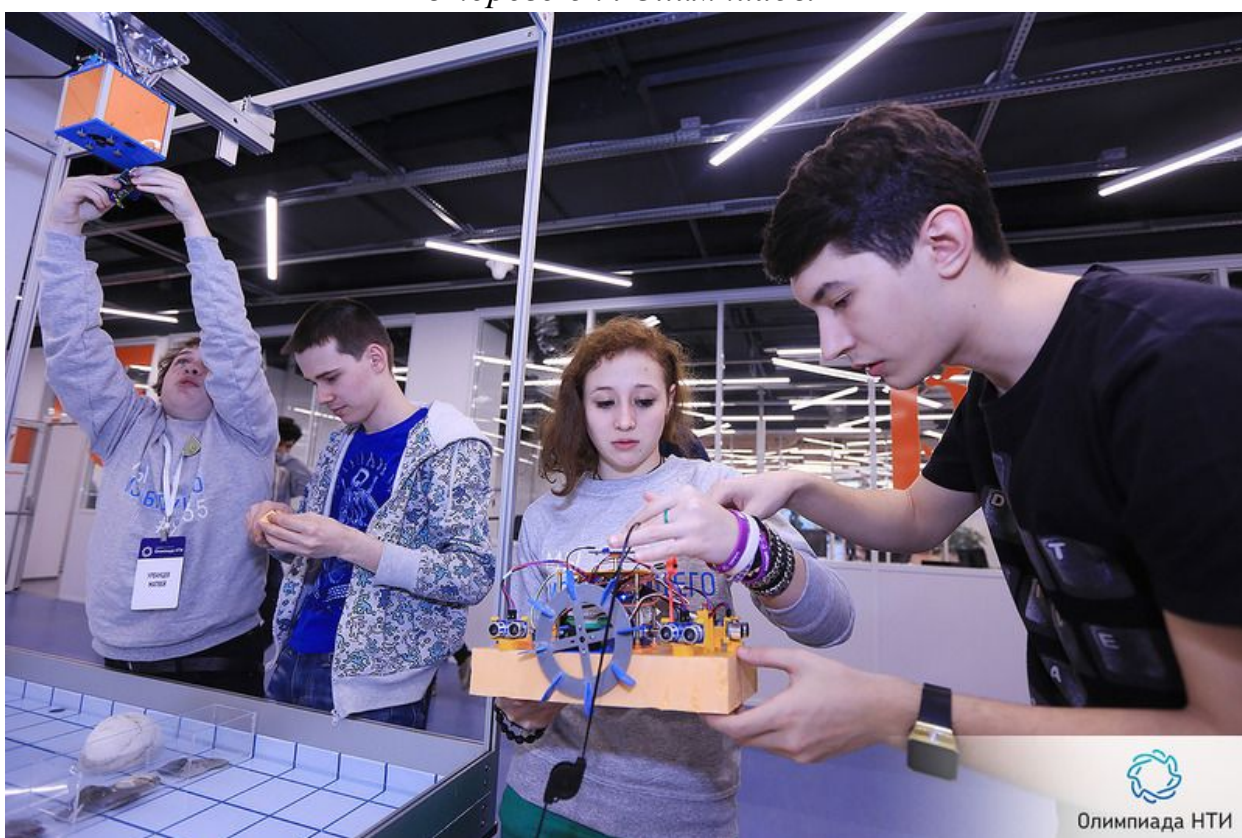


Рис.3 - Команда отлаживает корабль на полигоне подтрека “Маринет”

Баллы командного этапа:

Название команды	МАРИНЕТ	АВТОНЕТ	АЭРОНЕТ	СПЕЙСНЕТ	КОМПЛЕКС	ИТОГО
Void song	8,5	20	0	30		58,5

Маринет

Название команды	Участники		№ подзадач	База	Номер попытки										лучшая
					1	2	3	4	5	6	7	8	9	10	
Void song	Васильченко	Богдан	1	9	0	8,5									8,5
	Опякин	Роман	2.	5											0
	Сатаев	Эмиль	3.	5											0
	Тихомирова	Екатерина	4.	17											0
	Ястребов	Вячеслав	5.	5											0
			6.	9											0
РЕЗУЛЬТАТ															8,5

Автонет

Название команды	Участники		№ подзадач	База	Номер попытки										лучшая
					1	2	3	4	5	6	7	8	9	10	
Void song	Васильченко	Богдан	1	10	10										10
	Опякин	Роман	2.	2	2										2
	Сатаев	Эмиль	3.	4	4										4
	Тихомирова	Екатерина	4.	4	4										4
	Ястребов	Вячеслав	5.	2											0
			6.	8											0
			7.	20											0
РЕЗУЛЬТАТ															20

Аэронет

Название команды	Участники		№ подзадач	База	Номер попытки										лучшая
					1	2	3	4	5	6	7	8	9	10	
Void song	Васильченко	Богдан	1	10											0
	Опякин	Роман	2.	8											0
	Сатаев	Эмиль	3.	12											0
	Тихомирова	Екатерина	4.	15											0
	Ястребов	Вячеслав	5.	10											0
			6.	5											0
РЕЗУЛЬТАТ															0

Спейснет

Название команды	Участники		№ подзадач	База	Номер попытки										лучшая
					1	2	3	4	5	6	7	8	9	10	
Void song	Васильченко	Богдан	1	10	10										10
	Опякин	Роман	2.	5	5										5
	Сатаев	Эмиль	3.	15	15										15
	Тихомирова	Екатерина													0
	Ястребов	Вячеслав													0
															0
РЕЗУЛЬТАТ															30

Комплексный проезд

Не приступали.

Решения командного этапа:

Маринет

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

Zadaniye1.ino

```
#include <Servo.h>

#define LEFT A1

#define RIGHT A2

Servo servo;

Servo turn;

int ct = 90;

void calibrate()
{
    int l = analogRead(LEFT);
    int r = analogRead(RIGHT);
    if (l < 200)
    {
        turn.write(ct + 4);
        ct += 4;
        delay(100);
    }
    else if (r < 200)
    {
        turn.write(ct - 4);
```

```

        ct -= 4;

        delay(100);
    }
    else if (l > 200 && r > 200)
    {
        if (ct > 90) ct -= 3;
        else if (ct < 90) ct+=3;
        turn.write(ct);
    }
}

void setup() {
    // put your setup code here, to run once:
    pinMode(8, OUTPUT);
    pinMode(A0, INPUT);
    pinMode(A1, INPUT);
    pinMode(A2, INPUT);
    Serial.begin(9600);
    servo.attach(5);
    turn.attach(3);
    delay(2000);
    turn.write(90);
}

bool flag = true;

void loop() {

```

```
servo.write(100);

Serial.println("start");

while(analogRead(A0) > 300)

{

    delay(50);

    //Serial.println(analogRead(A0));

    calibrate();

}

delay(1500);

Serial.println("stop");

while(analogRead(A0) > 300)

{

    delay(100);

    if (flag) {

        digitalWrite(12, HIGH);

    }

    else

    {

        digitalWrite(12, LOW);

    }

    flag = !flag;

    calibrate();

}

Serial.println("asdasd");

digitalWrite(12, LOW);

servo.write(90);
```

```
digitalWrite(8, HIGH);  
  
delay(1000);  
  
digitalWrite(8, LOW);  
  
delay(90000000);  
  
}
```

Автонет

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

Zadaniye12.ino

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

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

```
#define SONAR_PIN 12
```

```
NewPing sonar(4, 2, 300);
```

```
Servo servo;
```

```
bool flag = true;
```

```
void setup() {
```

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

```
    // put your setup code here, to run once:
```

```
    servo.attach(5);
```

```
    //delay(2000);
```

```
    servo.write(90);
```

```
    delay (2000);
```



```
}
```

```
void calc()
```

```
{
```

```
  servo.write(40);
```

```
    delay(150);
```

```
    servo.write(90);
```

```
    delay(150);
```

```
}
```

```
void loop() {
```

```
  int d = sonar.ping_cm();
```

```
  Serial.println(d);
```

```
  if (d > 35)
```

```
  {
```

```
    servo.write(102);
```

```
    flag = true;
```

```
  } else if (d < 25 and d != 0) {
```

```
    if (flag){
```

```
      calc();
```

```
      flag = false;
```

```
    }
```

```
    servo.write(66);
```

```
  }
```

```
  else servo.write(90);
```

```
}
```

Zadaniye3.ino

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

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

```
#define getbit(x,pos) ( ((x) & ( 0x1 << (pos) )) !=0 )
```

```
const uint8_t IR_RECEIVER_PIN = 1;
```

```
IrReceiver receiver(IR_RECEIVER_PIN);
```

```
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()) {
```

```
        //Serial.println("looping");
```

```
}
```

```
uint8_t our_sig = receiver.getReceivedData();  
Serial.println("Принята последовательность:");  
int i;  
for (i = 0; i < 8; i++)  
    Serial.print(getbit(our_sig, i));  
Serial.println("");  
uint8_t signal = our_sig >> 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("!");

    }

}

```

Zadaniye4.ino

```

#include <Servo.h>

#include <stdint.h>

#include "irreceiver.h"

#include <string.h>

#include <NewPing.h>


#define TRIGGER_PIN 5 // Arduino pin tied to trigger pin on
the ultrasonic sensor.

#define ECHO_PIN 6 // Arduino pin tied to echo pin on the
ultrasonic sensor.

#define MAX_DISTANCE 200 // Maximum distance we want to ping for
(in centimeters). Maximum sensor distance is rated at 400-500cm.

NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE); // NewPing
setup of pins and maximum distance.


#define SetPin 2


const uint8_t IR_RECEIVER_PIN = 1;

IrReceiver receiver(IR_RECEIVER_PIN);

const uint8_t SIGNAL_STOP = 0xf0;

const uint8_t SIGNAL_FORWARD = 0xc0;

```

```
const uint8_t SIGNAL_LEFT = 0xa0;
const uint8_t SIGNAL_RIGHT = 0x90;
```

```
String str;

int x, y, angle;

int StartX, StartY;

int SatLoc;

int spd = 90;

Servo myservo;

Servo Motor;
```

```
#define Left 125

#define Mid 90

#define Right 55

// 90 центр

// 125 лево

// 55 право
```

```
void setup()

{ pinMode(SetPin, OUTPUT);

  digitalWrite(SetPin, LOW);

  Serial.begin(9600);

  Serial1.begin(9600);

  myservo.attach(9);

  myservo.write(90);

  Motor.attach(8);
```

```

Motor.write(90);

receiver.init();

ChannelDebag();


byte flag = 1;

while (flag)

{ if (Serial1.available())

    { spd = Serial1.parseInt();

      Serial.print("NewSpeed");

      str = Serial1.readStringUntil('\n');// LF

      if (spd == -3)

      {

          flag = 0;

      }

    }

    Serial.println(spd);

    Motor.write(spd);////////////////////////////////////////

    //myservo.write(spd);

}

Motor.write(90);

ChannelCoor();

MoveSat(-300);


MoveSat(78);

GetCoordinate();

StartX = x;

StartY = y;

```



```

    spd=97;
}

uint8_t signal = SIGNAL_FORWARD;
int distance = 0;
void loop()
{ GetCoordinate();

    if (receiver.isDataReceived())
    {
        signal = receiver.getReceivedData() >> 4;
        Serial.print("Incoming signal: ");
        Serial.print(signal, HEX);
        Serial.println("!");
    }

    distance = sonar.ping_cm();
    Serial.print("Distance: ");
    Serial.println(distance);
    if ((distance < 30) and (distance > 8))
    { Serial.println("Barrier");
        Motor.write(90);
        digitalWrite(13, HIGH);
    }
    else
    {

```

```

if (signal == SIGNAL_FORWARD)
{
  Serial.print("No problem -<< ");
  Serial.print(sp);
  Serial.println(" >>");
  if (y < 154)
  {
    Motor.write(90);
    if (SatLoc > 140)
    {
      while (1)
      {
        ///////////END END END END//////////
        digitalWrite(13, HIGH);
        delay(500);
        digitalWrite(13, LOW);
        delay(500);
      }
    }
    MoveSat(18);
  }
}
else
{
  if (x > StartX)
  {
    myservo.write(Mid + 10);
    Serial.println("NaLevo");
  }
  else
  {
    myservo.write(Mid - 10);
    Serial.println("NaPravo");
  }
  Motor.write(sp);
}

```

```

    }

    digitalWrite(13, LOW);
}

if (signal == SIGNAL_STOP)
{
    Serial.println("Stop");

    Motor.write(90);

    digitalWrite(13, HIGH);
}
}
}

```

```

void MoveSat(int steps)
{
    ChannelSatelite();

    Serial.print("MoveSat| ");

    Serial.println(steps);

    Serial1.println(steps);

    while (1)
    {
        if (Serial1.available() > 0)
        {
            str = Serial1.readStringUntil('\n'); //have /r/n sting

            Serial.println(str);

            if (str.substring(0,4) == "STOP")
            {
                SatLoc += steps;

                ChannelCoor();

                return ;
            }
        }
    }
}

```

```

        if (str.substring(0,4) == "END_")
        {
            SatLoc = 0;

            ChannelCoor();

            return ;
        }
    }
}

```

```

void GetCoordinate()
{
    while (1)
    {
        if (Serial1.available() > 0) {
            str = Serial1.readStringUntil('\n');
            //str = "00;1019;0609;192";

            Serial.println(str);

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

            return ;
        }
    }
}

```

```
void ChannelSatelite()
{ digitalWrite(SetPin, LOW);
  delay(200);
  Serial.println("Start HC-12 Config");
  Serial1.println ("AT+C010");
  delay(200);
  while (Serial1.available() > 0)
  { str = Serial1.readStringUntil('\n');// LF
    Serial.print ("Back: ");
    Serial.println (str);
  }
  digitalWrite(SetPin, HIGH);
  delay(100);
  Serial.println("Save & Exit");
}
```

```
void ChannelCoor()
{ digitalWrite(SetPin, LOW);
  delay(200);
  Serial.println("Start HC-12 Config");
  Serial1.println ("AT+C001");
  delay(200);
  while (Serial1.available() > 0)
  { str = Serial1.readStringUntil('\n');// LF
    Serial.print ("Back: ");
    Serial.println (str);
  }
}
```

```

    digitalWrite(SetPin, HIGH);

    delay(100);

    Serial.println("Save & Exit");
}

void ChannelDebag()
{ digitalWrite(SetPin, LOW);

  delay(200);

  Serial.println("Start HC-12 Config");

  Serial1.println ("AT+C003");

  delay(200);

  while (Serial1.available() > 0)

  { str = Serial1.readStringUntil('\n');// LF

    Serial.print ("Back: ");

    Serial.println (str);

  }

  digitalWrite(SetPin, HIGH);

  delay(100);

  Serial.println("Save & Exit");
}

```

Аэронет

Не приступали к программированию устройства и сдаче задания.

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 0
#define RIGHT_DIR 1

#define LED1 8
#define LED2 9
String str;
int state;
bool flag;

#define STEP_IN_ROTATION 200

const int volt_pin = A0;
float Voltage;
int dir, rotation;
long steps, i;
struct GEO {
    int x;
    int y;
    int grad;
};

GEO geo_data;

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

void loop() {
    flag = 0;
    // put your main code here, to run repeatedly:
    // if (Serial1.available() > 0) {

```



```

// max int 31072
// rotation = Serial1.readStringUntil('\n').toInt();

// if (rotation > 0) {
//   move(RIGHT_DIR, rotation);
//}

//if (rotation < 0) {
//move(LEFT_DIR, abs(rotation));
//}
read_geo_data();
Serial.println(geo_data.x);
Serial.println(geo_data.y);
Serial.println(geo_data.grad);
if ((geo_data.y <= 300) && (geo_data.y > 0)) {
  rotation = -2;
}
if ((geo_data.y > 300) && (geo_data.y < 700)) {
  rotation = 0;
}
if ((geo_data.y >= 700)) {
  rotation = 2;
}
/*if (flag == 0)
{
  if (state == -1) rotation = -1;
  if (state == 1) rotation = 1;
}
*/
if (rotation > 0) {
  move(RIGHT_DIR, rotation);
}

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

//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()) {
            rotation = 0;

```

```

        return;
    }

    if (isStopInSerial()) {
        return;
    }

    doStep();

}

Serial1.println("STOP");

}

GEO read_geo_data() {

    while (1) {

        if (Serial1.available() > 0) {

            str = Serial1.readStringUntil('\n');
            //str = "00;1019;0609;192";
            Serial.println(str);
            if (str == "STOP") return geo_data;
            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 doStep() {

    digitalWrite(STEP_PIN, HIGH);
    delayMicroseconds(800);
    digitalWrite(STEP_PIN, LOW);
    delayMicroseconds(800);

}

```