

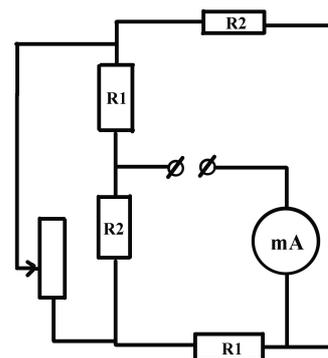
### §3 Заключительный этап: индивидуальная часть

Заключительный этап олимпиады состоит из двух частей: индивидуальное решение задач по предметам (физика, информатика) и командное решение инженерной задачи. На индивидуальное решение задач дается по 2 часа на один предмет. Решение каждой задачи дает определенное количество баллов (см. критерии оценки). По физике за каждую задачу можно получить от 0 до указанного количества баллов. Баллы по информатике зачисляются в полном объеме за правильное решение задачи. Решение задачи по информатике подразумевает написание программы на языке Python или C++. Участники получают оценку за решение задач в совокупности по всем предметам данного профиля (физика и информатика) — суммарно от 0 до 22 баллов.

#### 3.1. Задачи по физике (9 класс)

##### Задача 3.1.1 (3 балла)

В цепи, показанной на рисунке сопротивление реостата может меняться от 0 (нижнее положение ручки) до 100 кОм (верхнее положение). Когда ручка находится в верхнем положении, миллиамперметр показывает  $I_1 = 0,16$  А, а когда в нижнем -  $I_2 = 0,25$  А. Напряжение на клеммах  $U = 1,6$  В. Найдите величины  $R_1$  и  $R_2$ . Сопротивление миллиамперметра пренебрежимо мало.



##### РЕШЕНИЕ:

Можно увидеть, что эта схема представляет собой мост с реостатом посередине.

Когда реостат не имеет сопротивления, такая схема представляет собой последовательно соединенные параллельные группы. Общее сопротивление такой цепи

$R_0 = \frac{2R_1R_2}{R_1 + R_2}$ . Когда же реостат имеет сопротивление в 100 кОм ток через него практически

не течет и цепь представляет собой две параллельные группы последовательно соединенных

сопротивлений. Общее сопротивление такой цепи  $R_{100\kappa} = \frac{R_1 + R_2}{2}$

Тогда:

$I_1 = \frac{2U}{R_1 + R_2}$ , а  $I_2 = \frac{U(R_1 + R_2)}{2R_1R_2}$ . Из этих уравнений, исключив например  $R_2$ , выразив

из первого уравнения, получим:

$$I_1 I_2 R_1^2 - 2UI_2 R_1 + U^2 = 0$$

Решив данное квадратное уравнение получим:

$$R_1 = \frac{U}{I_1 I_2} (I_2 \pm \sqrt{I_2(I_2 - I_1)}), \quad R_2 = \frac{U}{I_1 I_2} (I_2 \mp \sqrt{I_2(I_2 - I_1)})$$

В силу симметрии схемы пара знаков может быть выбрана любым образом.

Значение сопротивлений равны: 16 Ом и 4 Ома.

#### **Критерии оценки:**

- Замечено, что схема представляет собой мост - 0,5 балла
- Выражены токи через закон Ома - 0,5 балла.
- Получено квадратное уравнение - 1 балл
- Получен ответ - 1 балл
- Если решения нет, но приведены разумные рассуждения в направлении решения - 0,5 балла

#### **Задача 3.1.2 (3 балла)**

В повести А. Беляева «Подводные земледельцы» упоминался подводный автомобиль. С какой максимальной скоростью мог бы двигаться такой автомобиль, если его масса равна 500 кг, внешний объем  $V = 150$  л. Какой мощности понадобился бы двигатель? Коэффициент сопротивления  $\alpha = 0,6$  Н · с/м, коэффициент трения между колесами и дном  $k = 0,5$ . Сила сопротивления воды прямо пропорциональна скорости. Потерями на трение в осях и турбулентными эффектами пренебрегите. Предполагается, что автомобиль движется по горизонтальной поверхности.

#### **РЕШЕНИЕ:**

Вперед автомобиль двигает сила трения между колесами и дорогой, в то время как сила сопротивления направлена всегда против скорости.

Легко можно увидеть, что существует максимальная скорость, при достижении которой никакое приращение мощности не даст прироста скорости.

С другой стороны, если мощность автомобиля недостаточна, то такая скорость достигнута не будет.

Скорость будет ограничена с одной стороны мощностью, которая необходима, чтобы

преодолеть силу сопротивления:  $\frac{N}{V} = \alpha V$  или  $V = \sqrt{\frac{N}{\alpha}}$ , с другой стороны максимальной силой трения, которую могут обеспечить колеса до проскальзывания:  $kmg = \alpha V$ ,  $V = \frac{k(mg - F_{apx})}{\alpha}$ . Минимальное из этих двух значений и определяет максимальное значение скорости.

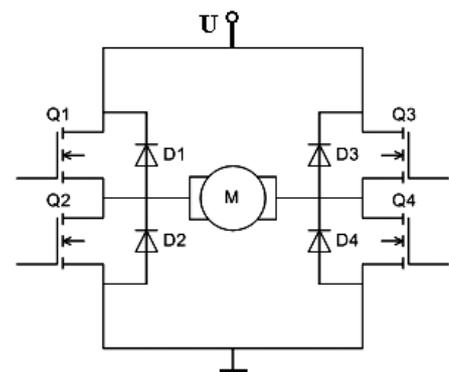
Отсюда значение ограничиваемое силой сцепления с поверхностью оказывается равным 2.8 км/с, однако понятно, что для такой огромной скорости линейная зависимость силы сопротивления от скорости уже не справедлива, кроме под водой турбулентные эффекты создадут существенно большее сопротивление. И двигатель для такой скорости нужен был бы мощностью не меньше 4,9 МВт.

#### Критерии оценки:

- Правильно записан второй закон Ньютона - 0,5 балла
- Получено выражение для максимальной скорости - 0,5 балла.
- Получено выражение для мощности - 1 балл
- Проанализирован физический смысл результата - 1 балл
- Если решения нет, но приведены разумные рассуждения в направлении решения - 0,5 балла

#### Задача 3.1.3 (3 балла)

На рисунке изображен H-мост на полевых транзисторах Q1-Q4. К данному мосту подключен однополюсной двигатель постоянного тока (на рисунке обозначен М). Если открыть транзисторы Q1 и Q4 вал двигателя будет вращаться в одну сторону, а если Q2 и Q3, то в другую. Для регулировки работы мост отключают на 10 с раз в минуту. Это означает, что транзисторы не пропускают ток. Считая, что на момент отключения двигатель работал с мощностью 200 Вт от напряжения 20 В, а электромагнитное поле в двигателе линейно падает до нуля за 4 с, найдите на какой ток должны быть рассчитаны диоды H-моста? Какая средняя тепловая мощность на них рассеется за время отключения H-моста? Сопротивление диода  $R = 0.001$  Ом. Активным сопротивлением обмотки двигателя пренебрегите.



### РЕШЕНИЕ:

Сразу после отключения внешнего напряжения ток в цепи можно считать не изменившимся и равным  $I = 10$  А. За указанное время ток снизится практически до 0. Если считать, что это происходит достаточно равномерно, можно оценить среднее количество теплоты, которое выделится на резисторах:  $Q = \frac{I^2}{4} \cdot 2r = 0,05$  Вт. Это достаточно грубая оценка.

#### Критерии оценки:

- Получен начальный ток - 0,5 балла
- Сделано разумное предположение об изменении тока - 0,5 балла.
- Получена тепловая мощность - 2 балла
- Если решения нет, но приведены разумные рассуждения в направлении решения - 0,5 балла

#### Задача 3.1.4 (3 балла)

В вакуумной лампе используется вольфрамовая нить диаметром  $d_1 = 2$  мм и длиной  $l = 16,7$  см. К лампе подводят напряжение и медленно его повышают. При напряжении  $U_1 = 625$  В нить перегорает. Какова температура горения вольфрама? Считайте, что сопротивление вольфрама прямо пропорционально абсолютной температуре  $T$ . Мощность теплового излучения с единицы площади поверхности нити можно считать равной  $\sigma T^4$  (здесь  $\sigma = 5,67 \cdot 10^{-8}$  Вт/(м<sup>2</sup>·К<sup>4</sup>) – постоянная Стефана–Больцмана). Удельное сопротивление вольфрама  $0,055$  Ом\*мм<sup>2</sup>/м при 20°C.

### РЕШЕНИЕ:

Поскольку изменения происходят медленно, можно считать, что система находится в тепловом равновесии. Тогда тепло выделяющееся на нити равно энергии излучения выделяющейся с поверхности нити.

Выделяющаяся тепловая мощность равна  $Q = \frac{U^2}{R}$ , где  $R = \frac{\rho L}{S} \cdot \frac{T}{T_0} = \frac{4\rho LT}{\pi d_1^2 T_0}$ .

Мощность излучения  $W = \sigma T^4 \pi d_1 l$

Приравнивая и выражая  $T$  получим:

$$T = \left( \frac{U^2 d_1 T_0}{4\rho \sigma l^2} \right)^{\frac{1}{5}} = 3719 \text{ К.}$$

Реальная температура плавления вольфрама 3695К

### Критерии оценки:

- Записано выражение для выделившейся энергии - 0,5 балла
- Записано выражение для лучистой энергии - 0,5 балла.
- Получен ответ - 2 балла
- Если решения нет, но приведены разумные рассуждения в направлении решения - 0,5 балла

## 3.2. Задачи по физике (10-11 класс)

### Задача 3.2.1 (3 балла)

Юный исследователь Майкл последовательно подключил к электрической сети школьного кабинета физики аккумулятор и измерил получившееся напряжение вольтметром. Вольтметр, рассчитанный на измерение как постоянного, так и переменного тока показал 44,9 В. После этого Майкл измерил тем же вольтметром напряжение отдельно в сети школьного кабинета и на аккумуляторе. В школьной сети вольтметр показал 42 В. Что показал вольтметр при подключении к аккумулятору?

### РЕШЕНИЕ:

Такой вольтметр показывает среднее эффективное значение.  $U_{эфф} = \sqrt{\langle U^2 \rangle}$ .

Т.к. при последовательном соединении напряжения переменного источника и постоянного аккумулятора сложатся:  $U = U_A \cos \omega t + U_{акк}$ . При этом надо иметь в виду, что напряжение на аккумуляторе  $U_{акк}$  вольтметр показывает правильно, а вот в переменной цепи гармонического тока вольтметр показывает действующее значение, отличающееся от амплитудного в  $\sqrt{2}$  раз.  $U_A = \sqrt{2}U_{пер} = \sqrt{2} \cdot 44,9$  В.

Тогда  $U^2 = 2U_{пер}^2 \cos^2 \omega t + U_{акк}^2 + 2U_{пер}U_{акк} \cos \omega t$ . При усреднении  $\cos^2$  дает за период  $\frac{1}{2}$ , в то время как среднее значение  $\cos$  равно нулю.

$$\langle U^2 \rangle = 2U_{пер}^2 \cdot \frac{1}{2} + U_{акк}^2 + 2U_{пер}U_{акк} \cdot 0 = U_{пер}^2 + U_{акк}^2, \text{ тогда}$$

$$U_{акк} = \sqrt{\langle U^2 \rangle - U_{пер}^2} = 16 \text{ В}$$

### Критерии оценки:

- Записано, что при последовательном соединении напряжения складываются - 0,5 балла

- Записано выражение для общего напряжения - 0,5 балла.
- Учтено, что вольтметр измеряет эффективное напряжение - 1 балл
- Получен ответ - 1 балл
- Если решения нет, но приведены разумные рассуждения в направлении решения - 0,5 балла

### Задача 3.2.2 (3 балла)

Автомобиль массой  $M = 800$  кг оснащен двигателем с максимальной полезной мощностью  $N = 80$  кВт. Предполагается, что автомобиль будет двигаться по дорогам, у которых коэффициент трения с колесами автомобиля может быть в диапазоне от  $k_{\min} = 0,1$  до  $k_{\max} = 0,3$ . Коэффициент лобового сопротивления автомобиля  $\alpha = 0,1$  Н\*с<sup>2</sup>/м<sup>2</sup>, а сила сопротивления пропорциональна квадрату скорости. Какова максимальная скорость движения автомобиля по горизонтальным участкам таких дорог?

#### РЕШЕНИЕ:

Вперед автомобиль двигает сила трения между колесами и дорогой, в то время как сила сопротивления направлена всегда против скорости.

Легко можно увидеть, что существует максимальная скорость, при достижении которой никакое приращение мощности не даст прироста скорости.

С другой стороны, если мощность автомобиля недостаточна, то такая скорость достигнута не будет.

Скорость будет ограничена с одной стороны мощностью, которая необходима, чтобы

преодолеть силу сопротивления:  $\frac{N}{V} = \alpha V^2$  или  $V = \sqrt[3]{\frac{N}{\alpha}}$ , с другой стороны максимальной

силой трения, которую могут обеспечить колеса до проскальзывания:  $kmg = \alpha V^2$ ,  $V = \sqrt{\frac{kmg}{\alpha}}$ .

Минимальное из этих двух значений и определяет максимальное значение скорости.

Можно увидеть, что для коэффициента  $k = 0,1$  скорость ограничена только сцеплением с поверхностью и равна 88,6 м/с, а уже для  $k = 0,11$  скорость ограничена мощностью двигателя и равна 92,8 м/с. Таким образом искомая максимальная скорость лежит между этими повышается от 88,6 до 92,8 м/с, но для большей части коэффициентов трения не может превышать 92,8 м/с.

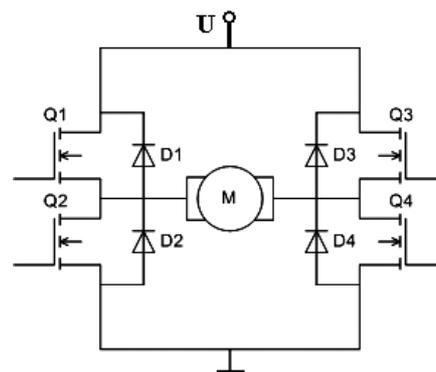
#### Критерии оценки:

- Правильно записан второй закон Ньютона - 0,5 балла

- Получено выражение для максимальной скорости - 0,5 балла.
- Получено значение для одного случая - 1 балл
- Получен диапазон значений - 1 балл (т.е. 3 балла в сумме)
- Если решения нет, но приведены разумные рассуждения в направлении решения - 0,5 балла

### Задача 3.2.3 (3 балла)

На рисунке изображен Н-мост на полевых транзисторах Q1-Q4. К данному мосту подключен однополюсной двигатель постоянного тока (на рисунке обозначен М). Если открыть транзисторы Q1 и Q4 вал двигателя будет вращаться в одну сторону, а если Q2 и Q3, то в другую. Для регулировки работы мост отключают на 10 с раз в минуту. Это означает, что транзисторы не пропускают ток.



Считая, что на момент отключения двигатель работал с мощностью 200Вт от напряжения 20 В, а частота вращения ротора  $\nu = 100$ Гц найдите, на какой ток должны быть рассчитаны диоды Н-моста? Какая средняя тепловая мощность на них рассеется за время отключения Н-моста? Сопротивление диода  $R = 0.001$  Ом, индуктивность двигателя  $L = 15$  мГн. Активным сопротивлением обмотки двигателя пренебрегите.

### РЕШЕНИЕ:

Сразу после отключения внешнего напряжения ток в цепи можно считать не изменившимся и равным  $I = 10$  А. За указанное время ток снизится практически до 0. Если считать, что это происходит достаточно равномерно, можно оценить среднее количество

теплоты, которое выделится на резисторах:  $Q = \frac{I^2}{4} \cdot 2r = 0,05$  Вт. Это достаточно грубая оценка.

### Критерии оценки:

- Получен начальный ток - 0,5 балла
- Сделано разумное предположение об изменении тока - 0,5 балла.
- Получена тепловая мощность - 2 балла
- Если решения нет, но приведены разумные рассуждения в направлении решения - 0,5 балла

### Задача 3.2.4 (3 балла)

Использованный юным техником Николой в своей схеме провод имеет диаметр  $d_1 = 0.4$  мм. длиной  $L = 1$  м с оплеткой ПВХ толщиной  $d_2 = 0.15$  мм. В процессе использования установки напряжение на проводе постепенно повышалось. При напряжении 13 В Никола измерил температуру оплетки провода и обнаружил, что она равна  $60^\circ\text{C}$ . Можно ли продолжать повышать напряжение? Удельное сопротивление меди равно  $1,72 \cdot 10^{-8}$  Ом·м, температура плавления меди  $1084^\circ\text{C}$ , температура возгорания оплетки  $400^\circ\text{C}$ , коэффициент теплопроводности материала оплетки  $0,19$  Вт/м· $^\circ\text{C}$ . Считайте, что сопротивление меди зависит линейно от абсолютной температуры.

#### РЕШЕНИЕ:

Выделяющаяся тепловая мощность равна  $Q = \frac{U^2}{R}$ , где  $R = \frac{\rho L}{S} \cdot \frac{T}{T_0} = \frac{4\rho LT}{\pi d_1^2 T_0}$ . С другой стороны, тепловой поток обеспечивающий теплоотвод равен:

$$\Phi = \eta \frac{S_{\text{нов}}}{d_2} (T - T_1) = \eta \frac{\pi d_1 L}{d_2} (T - T_1).$$

Приравняв эти два значения получим:

$$T^2 + TT_1 + \frac{U^2 d_1 d_2 T_0}{4\rho L^2 \eta} = 0, \text{ откуда } T = 671.5 \text{ К} = 398.5^\circ\text{C}$$

Видно, что повышать напряжение нельзя, т.к. обмотка проводов уже почти загорается.

#### Критерии оценки:

- Получено выражение для выделяющегося тепла - 0,5 балла
- Получено выражение для теплового потока - 0,5 балла.
- Записано квадратное уравнение - 1 балл
- Получена температура внутри оплетки - 0,5 баллов
- Сделан правильный вывод - 0,5 баллов.
- Если решения нет, но приведены разумные рассуждения в направлении решения - 0,5 балла

### 3.3. Задачи по информатике

#### Задача 3.3.1 (3 балла)

Датчики линии выдают значение от 0 до 1023, значение на выходе датчика  $e$  описывается дробно-кусочной функцией:

$$e(y) = \begin{cases} 1023, & y \leq 0 \\ 1023 \left(1 - \frac{y}{R}\right), & 0 < y \leq R \\ 0, & y > R \end{cases}$$

где  $y$  — расстояние от края пятна датчика до границы чёрного,  $R$  — радиус пятна датчика.

Иначе говоря, если датчик целиком над белым, он покажет 0; если датчик целиком над чёрным, то он покажет 1023. Чем больше датчик фиксирует чёрного фона, тем ближе к 1023 будет выдаваемое им значение. Данная выше функция корректна только при правильном выборе системы координат.

К машинке прикреплен линейка из восьми датчиков разметки. Машинка перемещается вдоль линии разметки так, что линейка перпендикулярна линии. Расстояние между центрами пятен датчиков 1,5 см.  $R = 0,5$  см. Ширина линии 2 см. На стандартный вход выдаются показания линейки датчиков — строка, содержащая восемь значений типа `int` от 0 до 1023, разделенных пробелами.

Вы должны получить число  $E$ , соответствующие расстоянию от центра линии до центра линейки. Выведите его в стандартный вывод как число с плавающей точкой, округлив до одного знака после запятой.

#### Пример ввода:

1023 1023 154 358 1023 1023 1023 1023

#### Пример вывода:

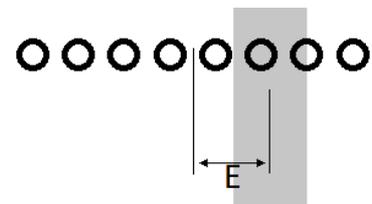
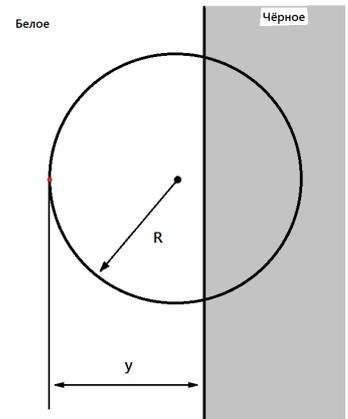
-1.6

#### СПОСОБ ОЦЕНКИ РАБОТЫ:

Для генерации тестов и проверки результата используется следующий код на языке Python:

```
import random as rd

class Task():
```



```

def __init__(self):

    self.line = (0.5 - rd.random()) * 10
    self.left = self.line - 1
    self.right = self.line + 1
    self.num = 8

    def _side(x):
        return [x-0.5, x+0.5]

    def _sens(self):
        sens = []
        init_pos = -(4.5 + 0.75)
        for i in range(self.num):
            if _side(init_pos + i*1.5)[1] < self.left:
                sens.append(1023)
                continue
            elif _side(init_pos + i*1.5)[0] > self.left and _side(init_pos +
i*1.5)[1] < self.right:
                sens.append(0)
                continue
            elif _side(init_pos + i*1.5)[0] < self.left and _side(init_pos +
i*1.5)[1] > self.left:
                sens.append(int(round(1023*(1 - (_side(init_pos + i*1.5)[1]-
self.left)),0)))
                continue
            elif _side(init_pos + i*1.5)[0] < self.right and _side(init_pos
+ i*1.5)[1] > self.right:
                sens.append(int(round(1023*(1 - (self.right-_side(init_pos +
i*1.5)[0])),0)))
                continue
            else:
                sens.append(1023)
                continue
        return sens
    self.sens = _sens(self)

    def __str__(self):
        return " ".join(str(i) for i in self.sens) + '\n'

class Solve():

```

```

def __init__(self, dots):
    self.dots = [int(i) for i in dots.split()]
    state = 0
    for idx, i in enumerate(self.dots):
        if i == 0:
            state = 1
            continue
        elif i != 1023:
            if state == 1:
                dist = (0.5 - i/1023.)
                self.line = -(4.5+0.75) + idx*1.5 + dist - 1
                break
            else:
                dist = (0.5 - i/1023.)
                self.line = -(4.5+0.75) + idx*1.5 - dist + 1
                break

def generate():
    tests = []
    for test in range(10):
        tsk = Task()
        tests.append(tsk.__str__())
    return tests

def check(reply, clue):
    return reply == clue

```

### **РЕШЕНИЕ:**

Программа на языке C++, решающая данную задачу:

```

1. #include <iostream>
2. using namespace std;
3. int main () {
4.     int sens[8];
5.     int state = 0;
6.     float dist = 0;
7.     float line = 0;
8.     for (int i=0; i < 8; i++) cin >> sens[i];
9.     for (int i=0; i < 8; i++) {
10.         if (sens[i] == 0) {
11.             state = 1;
12.             continue;
13.         }

```

```

14.         if (sens[i] != 1023) {
15.             if (state == 1) {
16.                 dist = (0.5 - sens[i]/1023.);
17.                 line = -(4.5+0.75) + i * 1.5 + dist - 1;
18.                 break;
19.             }
20.         else {
21.             dist = (0.5 - sens[i]/1023.);
22.             line = -(4.5+0.75) + i * 1.5 - dist + 1;
23.             break;
24.         }
25.     }
26. }
27.     cout << line;
28. }

```

### Задача 3.3.2 (4 балла)

Машинка движется по прямой. Из-за динамического сопротивления среды, зависимость скорости машинки от значения ШИМа на двигателе, описывается функцией

$$V(P) = aP^2 - bP,$$

где  $a$  и  $b$  — неизвестные параметры,  $V$  — скорость,  $P$  — значение ШИМа от 0 до 255.

Машинка ведёт журнал перемещений, запись журнала состоит из пары чисел: первое — координата, второе — значение ШИМа на двигателе. Запись происходит каждую 0,1 секунды. В промежутках между записями скорость машинки постоянная.

В некоторый момент после старта программа дала сбой и в лог записывается только координата. Имея в распоряжении оставшуюся часть лога, вам необходимо восстановить недостающие значения ШИМа.

Лог подается на стандартный ввод в виде набора строк формата:

```
<время> <расстояние> <ШИМ> '\n'
```

('n' - символ конца строки, не отображается) если значение ШИМа не утеряно, и формата:

```
<время> <расстояние> '\n'
```

если значение ШИМа утеряно. Пример данных приведен ниже. Утерянные значения ШИМа, начиная с первого утерянного и до последнего, выведите в стандартный вывод как набор чисел типа `int`, разделенных пробелами. Пример вывода так же приведен ниже.

**Пример ввода:**

0.00 0.00 0  
0.10 7.20 4  
0.20 34.40 8  
0.30 94.40 12  
0.40 200.00 16  
0.50 364.00 20  
0.60 599.20 24  
0.70 918.40 28  
0.80 1334.40 32  
0.90 1860.00 36  
1.00 2508.00 40  
1.10 3291.20 44  
1.20 4222.40 48  
1.30 5314.40 52  
1.40 6580.00 56  
1.50 8032.00 60  
1.60 9683.20 64  
1.70 11546.40 68  
1.80 13634.40 72  
1.90 15960.00 76  
2.00 18536.00 80  
2.10 21375.20 84  
2.20 24490.40 88  
2.30 27894.40 92  
2.40 31600.00 96  
2.50 35620.00 100  
2.60 39967.20 104  
2.70 44654.40 108  
2.80 49694.40 112  
2.90 55100.00 116  
3.00 60884.00 120  
3.10 67059.20 124  
3.20 73638.40 128  
3.30 80634.40 132  
3.40 88060.00 136  
3.50 95928.00 140  
3.60 104251.20 144  
3.70 113042.40 148  
3.80 122314.40 152  
3.90 132080.00 156  
4.00 142352.00 160

4.10 153143.20 164  
4.20 164466.40 168  
4.30 176334.40 172  
4.40 188760.00 176  
4.50 201756.00 180  
4.60 215335.20 184  
4.70 229510.40 188  
4.80 244294.40 192  
4.90 259700.00 196  
5.00 275740.00 200  
5.10 292427.20 204  
5.20 309774.40 208  
5.30 327794.40 212  
5.40 346500.00 216  
5.50 365904.00 220  
5.60 386019.20 224  
5.70 406858.40 228  
5.80 428434.40  
5.90 450760.00  
6.00 473848.00  
6.10 497711.20  
6.20 522362.40  
6.30 547814.40  
6.40 574080.00  
6.50 599330.60  
6.60 623586.20  
6.70 646866.80  
6.80 669192.40  
6.90 690583.00  
7.00 711058.60  
7.10 730639.20  
7.20 749344.80  
7.30 767195.40  
7.40 784211.00  
7.50 800411.60  
7.60 815817.20  
7.70 830447.80  
7.80 844323.40  
7.90 857464.00  
8.00 869889.60  
8.10 881620.20

8.20 892675.80  
8.30 903076.40  
8.40 912842.00  
8.50 921992.60  
8.60 930548.20  
8.70 938528.80  
8.80 945954.40  
8.90 952845.00  
9.00 959220.60  
9.10 965101.20  
9.20 970506.80  
9.30 975457.40  
9.40 979973.00  
9.50 984073.60  
9.60 987779.20  
9.70 991109.80  
9.80 994085.40  
9.90 996726.00  
10.00 999051.60  
10.10 1001082.20  
10.20 1002837.80  
10.30 1004338.40  
10.40 1005604.00  
10.50 1006654.60  
10.60 1007510.20  
10.70 1008190.80  
10.80 1008716.40  
10.90 1009107.00  
11.00 1009382.60  
11.10 1009563.20  
11.20 1009668.80  
11.30 1009719.40  
11.40 1009735.00  
11.50 1009735.60

**Пример вывода:**

232 236 240 244 248 252 256 251 246 241 236 231 226 221 216 211 206  
201 196 191 186 181 176 171 166 161 156 151 146 141 136 131 126 121 116  
111 106 101 96 91 86 81 76 71 66 61 56 51 46 41 36 31 26 21 16 11 6 1

**СПОСОБ ОЦЕНКИ РАБОТЫ:**

Для генерации тестов и проверки результата используется следующий код на языке

## Python:

```
from functools import reduce
import random as rd
import numpy as np

class Track():
    def __init__(self, a, b, t):
        self.a, self.b = a, b
        self.P = 0
        self.t = float(t)
        self.dist = []
        self.vel = []
        self.sh = []
        self.timet = []

    def speed (self):
        return self.a * self.P**2 + self.b * self.P

    def accel(self):
        step = rd.randint(3,6)
        while self.P < 256:
            try:
                self.dist += [self.dist[-1] + self.speed()*self.t]
            except IndexError:
                self.dist += [self.speed()*self.t]
            try:
                self.timet += [self.timet[-1] + self.t]
            except IndexError:
                self.timet += [0]
            self.vel += [self.speed()]
            self.sh += [self.P]
            self.P += step
        self.dist[0] = 0.0

    def stop(self):
        step = -1 * rd.randint(3, 6)
        while self.P >= 0:
            try:
                self.dist += [self.dist[-1] + self.speed()*self.t]
                self.timet += [self.timet[-1] + self.t]
            except IndexError:
```

```

        self.dist += [self.speed()*self.t]
        self.timet += [self.t]
        self.vel += [self.speed()]
        self.sh += [self.P]
        self.P += step
self.dist[0] = 0.0

def _str(self, nline):
    self.__l = int(len(self.dist)/2)
    if nline > self.__l-1:
        return ('{0:.2f}'.format(self.timet[nline]) + ' ' +
'{0:.2f}'.format(self.dist[nline]) + '\n')
    if nline <= self.__l-1:
        return ('{0:.2f}'.format(self.timet[nline]) + ' ' +
'{0:.2f}'.format(self.dist[nline]) + ' ' + '{}'.format(self.sh[nline]) + '\n')

    def __str__(self):
        return reduce(lambda res, x: res+x, map(lambda x: self._str(x),
range(len(self.dist))), "")

class Solver():
    def __init__(self, tm, dist, shm):
        self.dist = dist
        self.shm = shm
        self.l = int(len(shm)/2)
        # vel init
        self.vel = []
        self._t = tm[1]-tm[0]
        for idx, i in enumerate(self.dist):
            if i > 0:
                self.vel += [(self.dist[idx] - self.dist[idx-1])/self._t]
            else:
                self.vel += [self.dist[idx]/self._t]

        self.s1, self.s2 = self.shm[self.l-1], self.shm[int(self.l/2)]
        self.v1, self.v2 = self.vel[self.l-1], self.vel[int(self.l/2)]
        self.s1sq, self.s2sq = self.s1**2, self.s2**2

        self.mat = np.linalg.det(np.array([[self.s1sq, self.s1], [self.s2sq,
self.s2]]))

```

```

        self.ymt = np.linalg.det(np.array([[self.s1sq, self.v1], [self.s2sq,
self.v2]]))
        self.xmt = np.linalg.det(np.array([[self.v1, self.s1], [self.v2,
self.s2]]))

def value(self, mat1, mat2):
    return int(round(mat1/mat2, 0))

def output(self):
    a = self.value(self.xmt, self.mat)
    b = self.value(self.ymt, self.mat)
    self.ret_shm = []
    for i in range(len(self.shm), len(self.dist)):
        val = int(round((-b + (b**2 + 4*a*self.vel[i])**0.5)/(float(2*a)),
0))
        self.ret_shm += [val]

    return self.ret_shm

def generate():
    tests = []
    for i in range(8):
        tr = Track(rd.randint(2, 4), rd.randint(2, 10), 0.1)
        tr.accel()
        tr.stop()
        tests.append(tr.__str__())
    return tests

def check(reply, clue):
    return reply == clue

```

### **РЕШЕНИЕ:**

Пример программы на языке C++, решающей данную задачу:

```

1. #include <iostream>
2. #include <sstream>
3. #include <vector>
4. #include <string>
5. #include <math.h>
6.
7. using namespace std;
8.
9. int get_info(vector<float> &time, vector<float> &dist, vector<int> shim)

```

```

10. {
11.     vector<float>::iterator t_it = time.begin();
12.     vector<float>::iterator d_it = dist.begin();
13.     vector<int>::iterator s_it = shim.begin();
14.
15.     string str;
16.     while (getline (cin, str))
17.         {
18.             istringstream iss(str);
19.             float buf;
20.             vector <float> v;
21.             for (vector<float>::iterator it = v.begin(); iss; it++)
22.                 {
23.                     iss >> buf;
24.                     v.push_back(buf);
25.                 }
26.             if (v.size() == 4) {
27.                 time.push_back(v[0]);
28.                 dist.push_back(v[1]);
29.                 shim.push_back(v[2]);
30.                 t_it++, d_it++, s_it++;
31.             }
32.             if (v.size() == 3) {
33.                 time.push_back(v[0]);
34.                 dist.push_back(v[1]);
35.                 t_it++, s_it++;
36.             }
37.         }
38.     return 0;
39. }
40.
41. float square_eq (int a, int b, int vel)
42. {
43.     return (float) (-b + (float) sqrt(b*b + 4*a*vel)) / (float) (2*a) ;
44. }
45.
46. int main ()
47. {
48.     vector<float> time, dist, sped;
49.     vector<int>     shim;
50.

```

```

51.     get_info(time, dist, shim);
52.
53.     int sh_end = shim.size()/2;
54.     int di_end = dist.size();
55.     int ti_end = time.size();
56.
57.     for (vector<float>::iterator d=dist.begin(); d != dist.end(); ++d)
58.         {
59.             float dtime = (float) (time[1] - time[0]);
60.             if (d != dist.begin()) sped.push_back((float) ((*d-
*(d-))/dtime));
61.             else sped.push_back((float) (*d/dtime));
62.         }
63.
64.     float m1 = shim[sh_end/2];
65.     float m2 = shim[sh_end-1];
66.
67.     float s1 = sped[sh_end/2];
68.     float s2 = sped[sh_end-1];
69.
70.     float mm1 = m1*m1;
71.     float mm2 = m2*m2;
72.
73.     float main_det = -(mm1*m2 - mm2*m1);
74.         float det_b = -(mm1*s2 - mm2*s1);
75.     float det_a = -(s1*m2 - s2*m1);
76.
77.     int a = (int) round (det_a/main_det);
78.     int b = (int) round (det_b/main_det);
79.
80.     for (int i=shim.size(); i < di_end; i++) cout << (int)
round(square_eq(a, b, sped[i])) << " ";
81. }

```

### **Задача 3.3.3 (3 балла)**

На вход программы подается строка из символов '0' и '1', соответствующая сообщению в формате Sony SIRC 20, используемым в передаче инфракрасных сигналов. Строка начинается и заканчивается случайным количеством нулей, которое необходимо пропустить. Начало сообщения отмечено последовательностью '11110'. Далее идет набор символов, где последовательностью '110' зашифрован бит со значением '1', а



```

        if rd.random() > 1/128.: return dev
        else: return "{0:012b}".format(rd.randint(0, 4095))

def __init__(self, message, case = 'ok'):
    self.command = rd.randint(0, 127)
    self.dev_n = rd.randint(0, 31)
    self.dev = "{0:07b}{1:05b}".format(self.command, self.dev_n)

    self.msg = message
    self.leng = len(message)
    self.bitarr = [self.cod(self.dev)+"{0:08b}".format(ord(i)) for i in
self.msg]

    # if case == 'nochk':
    #     self.csum = rd.randint(0, 255)
    # else:
    #     self.csum = self.leng

    # if case == 'nolen':
    #     self.leng = rd.randint(0, self.leng)
    # else:
    #     pass

# self.l_bit = self.cod(self.dev)+"{0:08b}".format(self.leng)

# for i in self.msg:
#     self.csum ^= ord(i)
#     self.bitarr = self.bitarr

    self.sirc = '0'*rd.randint(8, 54) + reduce (lambda res, x: res + '11110'
+ x, [self.deconv(i) for i in self.bitarr], "")

    self.corr = ''.join([self.crr(i) for i in self.sirc])

def __str__(self):
    return self.sirc + '\n'

def generate():
    tests = []
    for test in range(10):
        msg = Msg(chr(rd.randint(0x41, 0x7A)))

```

```

        tests.append(msg.__str__())
    return tests

def solve(dataset):
    par = Parse(dataset.strip())
    par.tobit()
    return par.__str__() + '\n'

def check(reply, clue):
    reply = reply.lower()
    clue = clue.lower()
    r = reply.split()

    c = clue.split()

    return c == r

```

## **РЕШЕНИЕ:**

Пример программы на языке C++, решающей данную задачу:

```

1. #include <iostream>
2. #include <vector>
3. #include <string>
4. #include <cstdio>
5. #include <cmath>
6.
7. #define ERROR 1
8. #define OK 0
9.
10. using namespace std;
11.
12. int getint () {
13.     char c;
14.     if ((c = getchar()) != '\n') return (int) (c - '0');
15.     else return -1;
16. }
17.
18. int decimal(vector<int>&data,vector<int>::iterator &it,int len) {
19.     int num = 0;
20.     for (vector<int>::iterator i=it; (int) (it-i)<len; i--) {
21.         if (i == it) num += *i;

```

```

22.         else num += *i * pow(2, (int) (it-i));
23.     }
24.     it -= len;
25.     return num;
26. }
27.
28. int main ()
29. {
30.     int buf;
31.     //save raw && output
32.     vector<int> rawdata, data;
33.     while ((buf=getint()) != -1) rawdata.push_back(buf);
34.     vector<int>::iterator it = rawdata.begin();
35.     vector<int>::iterator ed = rawdata.end();
36.     while (it < rawdata.end()-4) {
37.         if (*it==1 && *(it+1)==1 && *(it+2)==1 && *(it+3)==1 &&
*(it+4)==0) {
38.             it += 5;
39.             break;
40.         }
41.         else it++;
42.     }
43.     // interrupts if out of vector after
44.     if (it >= rawdata.end()-4) {
45.         return ERROR;
46.     }
47.     // convert 10 -> 0; 110 -> 1
48.     while (it != rawdata.end()-3) {
49.         if (*it==1 && *(it+1)==0) {
50.             data.push_back(0);
51.             it += 2;
52.             continue;
53.         }
54.         if (*it==1 && *(it+1)==1 && *(it+2)==0) {
55.             data.push_back(1);
56.             it += 3;
57.             continue;
58.         }
59.         else it++;
60.     }
61.     vector<int>::iterator dt = data.end()-1;

```

```
62. //parsing from the end of vector<>data; look at decimal()
63. int inf = decimal (data, dt, 8);
64. int dev = decimal (data, dt, 5);
65.     int com = decimal (data, dt, 7);
66. printf ("0x%02X 0x%02X 0x%02X", com, dev, inf);
67. cout << endl;
68. return OK;
69. }
```