

Олимпиада школьников «Надежда энергетики»

КТОФ01	ВКС
--------	-----

№ группы

Место проведения

JA38-28

шифр

← Не заполнять
Заполняется
ответственным
работником

Вариант № 47701

ФАМИЛИЯ Мишин

ИМЯ Михаил

ОТЧЕСТВО Александрович

Дата рождения 08.08.2006

Класс: 10

Предмет КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ Этап: ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 4 листах

Дата выполнения работы: 25.03.2023
(число, месяц, год)

Подпись участника олимпиады: 

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, общее количество листов, на которых выполнена работа и дату выполнения работы.



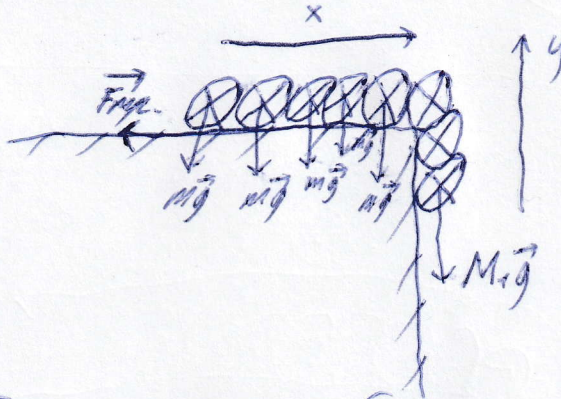
ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Дано:
 $m = 2 \text{ кг}$
 $D = 30 \text{ см}$
 $N = 50 \text{ шт}$
 $\mu = 0,11$
 $g = 9,81 \text{ м/с}^2$

N - число шариков.

1) Найдите K_0 .

Рассмотрим силы, действующие на шарик.



обозначим (для удобства) за M_1 массу всех свисавших шариков, а за M_0 массу всех шариков на горизонтальной поверхности.

Рассмотрим проекции сил по осям x и y :

$$\sum F_x = -F_{\text{тр}} = ma$$

$$\sum F_y = -M_1 g = -ma$$

$$ma = -F_{\text{тр}}$$

$$ma = M_1 g$$

$$-F_{\text{тр}} = M_1 g$$

$$-\mu \cdot M_0 g = M_1 g$$

но $F_{\text{тр}} + M_1 g = ma$ (так как система находится в движении).

$m g$ шариков горизонтальной поверхности учитывается при расчете $F_{\text{тр}}$.

неподвиж. сист.

Видим, что для поддержания системы

$F_{\text{тр}}$ ~~должна~~ $(\mu M_0 g)$ должна быть равна μ раз массе, действующей на свисавшие шары $(M_1 g)$

~~$\mu M_0 g = M_1 g$~~ Найдём, при каком наименьшем числе шаров система останется неподвижной:

Обозначим кол-во свисавших шаров за x .

$$\mu \cdot (N - x) m g = x m g$$

$$0,11 \cdot (50 - x) 2g = x 2g$$

$$(0,11 \cdot 50 - 0,11x) 2g = 1g x$$



ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

$$15,5 - \frac{11x}{100} + 19,62 = 19,62x$$

$$107,91 - 2,1582x = 19,62x$$

$$21,7782x = 107,91$$

$$x \approx 4,9549 \approx 4,95$$

Для движения шаров нужно, чтобы ^{ценности} весилась 4 шара, а ещё один весилась на \approx на 0,95 (т.е. 0,95 его массы) оказалась в вертикальной плоскости.

Если пренебречь этим, то можно сказать, что система шаров в движении при 5 висюльщика шаров.

~~$$K_0 = 5 \text{ (} 4,95 \text{ - точное число, округлённое до сотых)}$$~~

$K_0 = 5$ (если требуется целое кол-во шариков)

$K = 4,96$ (точное число, округлённое до сотых)

2) Для решения этого пункта составил алгоритм на языке Python: Полученная программа позволяет определять время, через которое висюльщик с первой шарик и с первой висюльщик по заданному кол-ву висюльщика шаров. Минимальное кол-во висюльщика шаров 4,96, т.к. при меньшем значении система не начнёт двигаться. Максимальное значение 48, т.к. в горизонтальной плоскости должны висеть 48 шаров, для вычисления времени их раскачивания.

При 4,96 висюльщика шаров первый шар висюльщик через: $\approx 3,3$ секунды

а первый и второй через: $\approx 3,59$ сек. после начала движения. (Программа 2)



ВНИМАНИЕ! Проверяется только то, что записано

с этой стороны листа в рамке справа

3) составим программу для нахождения T и V
 Получим результатом: $T \approx 5,1$ сек.

$$UV \approx 30,87 \text{ м/с.}$$

(программа 3) На языке программирования Python.

$$9) H = 60 D = 60 \cdot 0,3 (\text{м}) = 18 \text{ м.}$$

При этом длина всей цепочки шаров $50 \cdot 0,3 = 15$ (м.)

Значит, после перехода всех шаров в вертикальную плоскость шары подлетят ещё ⁽⁵⁾ 3 м $18 - 15 = 3$ (м.) с ускорением свободного падения ($9,81 \text{ м/с}^2$)

$V = V_0 + at$ - воспользуемся этой формулой.

$$V = V + gt - \text{имеем.}$$

V - скорость, найденная в прошлом пункте

$$V = 30,87 + 9,81 t$$

$s = 3$ м. Воспользуемся формулой

$$s = V_0 t + \frac{at^2}{2} \quad \text{Имеем: } 3 = 30,87 t + \frac{9,81 t^2}{2}$$

$$\text{Найдём } t: 6 = 61,74 t + 9,81 t^2$$

$$9,81 t^2 + 61,74 t - 6 = 0$$

$$\text{Откуда } t \approx 0,0957 \text{ с}$$

$$V = 30,87 + 9,81 \cdot 0,0957 \approx 31,81 \text{ м/с.}$$

Ответ: $V = 31,81 \text{ м/с.}$

5) ~~составим~~ составим программу на языке программирования Python (программа 5).

Программа по введённому диаметру шарика находит T (из пункта 3), а также выводит время, через которое движущийся и соскакивает с шарика, при $K_0 = 4,96$. Всегда диаметр d вводим, что $T \approx 10,2$, что не



ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

меньше, чем 2. Типичное в пункте 31)

$$10,2 = 2 \cdot 5,1$$

$$10,2 = 10,2 \cdot$$

$$\text{или } K_0 = 8,96$$

Также получим, что 1ый шар точкой системы начнется двигаться через 0,034 м.

Значит 1,2 метра - школьный футбольный.

$$1,2 \text{ м} = 120 \text{ см}.$$

Ответ: $D_2 = 120 \text{ см}.$

Все программы, представленные в решении написаны в IDLE Python 3.1

2 программа Python

```
n=50
m=2
d=0.3
u=0.11
g=9.81
a0=0
t0=0
t1=0
v=0
n=float(n)
m=float(m)
d=float(d)
u=float(u)
g=float(g)
a0=float(a0)
t0=float(t0)
v=float(v)
t1=float(t1)
print("Введите количество свешивающихся шаров(минимум 4.96;максимум 48)")
x=float(input())
a0=(x*g)-(u*(n-x)*g)
t0=(0.6/a0)**0.5
v=a0*t0
print('Первый шар соскочит через',t0,('секунд'))
x=x+1
a=(x*g)-(u*(n-x)*g)
t1=((0.6)/((2*a0)+a))**0.5
print('Второй и первый шары соскочат через',(t1+t0),('секунд'))
```


3 программа Python

```

n=50
m=2
d=0.3
u=0.11
g=9.81
a0=0
t0=0
t1=0
v=0
k=0
vs=0
n=float(n)
m=float(m)
d=float(d)
u=float(u)
g=float(g)
a0=float(a0)
t0=float(t0)
v=float(v)
t1=float(t1)
x=4.96
x=float(x)

a0=(x*g)-(u*(n-x)*g)
t0=((2*d)/a0)**0.5
v=a0*t0
k=t0
vs=v

for i in range(0,45):
    ##print(i+1,'шар соскочит через',t0,('секунд'))
    x=x+1
    a=(x*g)-(u*(n-x)*g)
    t1=((2*d)/((2*a0)+a))**0.5
    a0=a
    v=a0*t1
    ##print(i+1,'шар соскочит через',(t1),('секунд'))
    ##print(i+1,'ая пара шаров соскочи через',t0+t1,('секунд'))
    k=k+t1
    vs=vs+v
print("T =",k,"секунд")
print("U =",vs/10,"метров в секунду")

```


5 программа Python

```
n=50
```

```
m=2
```

```
d=float(input())
```

```
u=0.11
```

```
g=9.81
```

```
a0=0
```

```
t0=0
```

```
t1=0
```

```
v=0
```

```
k=0
```

```
vs=0
```

```
n=float(n)
```

```
m=float(m)
```

```
d=float(d)
```

```
u=float(u)
```

```
g=float(g)
```

```
a0=float(a0)
```

```
t0=float(t0)
```

```
v=float(v)
```

```
t1=float(t1)
```

```
x=4.96
```

```
x=float(x)
```

```
a0=(x*g)-(u*(n-x)*g)
```

```
t0=((2*d)/a0)**0.5
```

```
v=a0*t0
```



```
k=t0
```

```
vs=v
```

```
for i in range(0,45):
```

```
    ##print(i+1,'шар соскочит через',t0,('секунд'))
```

```
    x=x+1
```

```
    a=(x*g)-(u*(n-x)*g)
```

```
    t1=((2*d)/((2*a0)+a))**0.5
```

```
    a0=a
```

```
    v=a0*t1
```

```
    ##print(i+1,'шар соскочит через',(t1),('секунд'))
```

```
    ##print(i+1,'ая пара шаров соскочи через',t0+t1,('секунд'))
```

```
    k=k+t1
```

```
    vs=vs+v
```

```
print("T =",k,"секунд")
```

```
a0=(x*g)-(u*(n-x)*g)
```

```
t0=(0.6/a0)**0.5
```

```
v=a0*t0
```

```
print('Первый шар соскочит через',t0,('секунд'))
```


Олимпиада школьников «Надежда энергетики»

К11F01 Дистанционно,
с использованием ВКС

№ группы

Место проведения

CA88-96

шифр

← Не заполнять
Заполняется
ответственным
работником

Вариант № 47111

ФАМИЛИЯ МУХИНА

ИМЯ МАРИЯ

ОТЧЕСТВО ОЛЕГОВНА

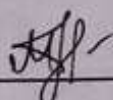
Дата рождения 13.11.2005

Класс: 11

Предмет КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ Этап: ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 4 листах

Дата выполнения работы: 25.03.2023
(число, месяц, год)

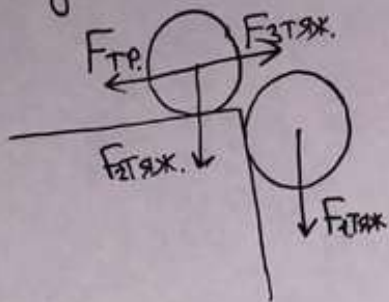
Подпись участника олимпиады: 

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, общее количество листов, на которых выполнена работа и дату выполнения работы.



ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задача 1.



Рассмотрим небольшую часть цепи, состоящую из одного свешивающегося и одного горизонтально расположенного шаров. На первый шар действует по условию задачи только сила тяжести ($F_{тяж}$), а на 2-ой шар действует

$F_{тр}$ — сила трения, и две силы тяжести $F_{тяж}$ и $F_{зтяж}$. Но, по сути, $F_{тяж} = F_{зтяж}$, так как сила $F_{зтяж}$, под воздействием которой движется шар, является $F_{тяж}$ шара, который тянет цепочку шаров вниз.

По условию задачи известно, что m одного шара равна 2 кг , μ коэффициент трения равен $0,11$; $g = 9,81 \frac{\text{м}}{\text{с}^2}$.

Обозначим за k кол-во лежащих шаров, а за n — кол-во висящих. $F_{тр} = \mu N = \mu * m * g$. $F_{тяж} = m * g$.

Чтобы цепочка соскользнула в воду, необходимо, чтобы $F_z > F_{тр} \Rightarrow m * g * k * n > \mu * m * g * k$. П.к. m и g есть в обеих частях неравенства, можно их не учитывать.

В приведенной программе есть изначальные данные: $n = 1$, $\mu = 0,11$; $k = 50,0$, то есть наибольшее кол-во шаров; $n = 0,0$. Программа создает цикл с условием, что k не равно 0. Вводится переменная $l = n * k$.

Далее проверяется условие, что l меньше n , если оно выполняется, то программа выводит n и k и прекращает работу цикла, а иначе она уменьшает кол-во лежащих шаров на $1,0$ и увеличивает кол-во висящих на то же число. Все переменные представлены типа $float$ для удобства подбора.

Ответ: наименьшее кол-во свешивающихся шаров
5.

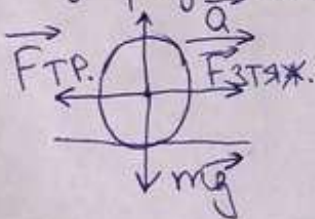


ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задание 2.

Все данные и в условии переменные описаны в решении задачи 1 и их обозначение в программе не отменяется. Во время анализа вопроса задачи 2 я поняла, что возможны две трактовки: 1) первый соскользнувший шар, то есть находящийся на схеме справа; 2) первый шар слева, то есть тот, который упадет последним. Так как трактовка 2 означает тот же вопрос, что и в задании 3, здесь будет решение задачи 2 с трактовкой 1, то есть необходимо узнать, за какое время соскочит первый шар справа, а затем

Из физических понятий известно, что:



$$m\vec{a} = \vec{F}_{тяг} + \vec{N} + \vec{F}_{тяж} + \vec{F}_{тр}$$

$$kma = mg + mg_n + \mu mg_k$$

$$ka = g + g_n + \mu g_k$$

$$a = \frac{g + g_n + \mu g_k}{k}$$

В первом вопросе примем за $S=30$ - расстояние, которое пройдет шар перед тем, как упасть, око ребро диаметру одной шара.

В начале программы я использовала библиотеку math, которая далее вычисляла корень. (sqrt). Программа вычисляла ~~корень~~ ускорение a , как и в приведенном выше уравнении. Далее время t находилось по формуле $S = \frac{at^2}{2} \Rightarrow at^2 = 2S \Rightarrow t = \sqrt{\frac{2S}{a}}$.

В итоге программа выводит время t , ответ на первый вопрос



ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Чтобы получить ответ на второй вопрос задачи, я скепировала уже имеющиеся в программе переменные и формулы, но уменьшила k на 1, а n увеличила на 1, S я ужижала на 2, так как второй шаг прыжок не 30 см, а 60 см.

Важное уточнение. При решении задачи я искала зовала данные, полученные в задаче 1 и начальное k взяла за 45.0, а n за 5.0.

Ответ: ~~1) 5.01; 2) 11.76 с. 9,78 с.~~ 1) 0,5 с.; 2) 0,98 с.

Задача 3.

Решение аналогично решению задачи 2.

Изменение: введен цикл, который на входе проверяет условие k не равно 0, далее каждый раз находит новое ускорение a и получает время t путем складывания ранее полученной времени с новым, зависящим от нового a . k и n изменяются как и ранее. Далее программа выводит итоговое время t . Для получения ответа на второй вопрос я использовала формулу $S = \frac{v^2 - v_0^2}{2a}$, $v_0 = 0 \Rightarrow S = \frac{v^2}{2a} \Rightarrow v^2 = S \cdot 2a \Rightarrow$

$\Rightarrow v = \sqrt{S \cdot 2 \cdot a}$, скорость в программе обозначена v . Через второй `print` выведется скорость v (v).

Ответ: ~~1) 104,36 с.; 2) 171,74~~ 1) 10,44 с.; 2) 17,17 с.



ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задача 5.

При увеличении размера шариков, то есть D , будет увеличиваться путь S , так как по решению $S = D$. Значит, будет увеличиваться и время, за которое все шарики сплывут в воду.

Изменив начальное значение S в программе к задаче 3, я получила, что $2T = 20,88$ с. Можно получить, если S будет больше или равно 1.20084.

Ответ: 1,20084 м.

Все программы представлены на языке программирования python

#решение задачи 1

```
nu = 0.11
```

```
k = 50.0
```

```
n = 0.0
```

```
while k != 0:
```

```
    l = nu * k
```

```
    if l < n:
```

```
        print(n)
```

```
        break
```

```
    k -= 1.0
```

```
    n += 1.0
```


#решение задачи 2

```
from math import sqrt
```

```
k = 45.0
```

```
n = 5.0
```

```
nu = 0.11
```

```
g = 9.81
```

```
s = 0.3
```

```
a = (g + g * n + nu * g * k) / k
```

```
t = sqrt(2 * s/a)
```

```
print(t)
```

```
k = 44.0
```

```
n = 6.0
```

```
a = (g + g * n + nu * g * k) / k
```

```
t += sqrt(2 * s/a)
```

```
print(t)
```

```
#решение задачи 3
from math import sqrt
k = 45.0
n = 5.0
nu = 0.11
g = 9.81
s = 0.3
t = 0
a = 0
while k != 0:
    a = (g + g * n + nu * g * k) / k
    t += sqrt(2 * s/a)
    k -= 1.0
    n += 1.0
print(t)
v = sqrt(s * 2 * a)
print(v)
```


Олимпиада школьников «Надежда энергетики»

	ВКС
--	-----

№ группы

Место проведения

СА88-42

шифр

← Не заполнять
Заполняется
ответственным
работником

Вариант № 47 III

ФАМИЛИЯ Назарюхи

ИМЯ Магвей

ОТЧЕСТВО Евгеньевич

Дата рождения 04.03.2005

Класс: 11

Предмет Компьютерное моделирование Этап: Заключительный

Работа выполнена на 3 листах

Дата выполнения работы: 25.03.2023
(число, месяц, год)

Подпись участника олимпиады:



Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, общее количество листов, на которых выполнена работа и дату выполнения работы.



ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

У меня реализованы 4 пункта задачи.

Разберем их по порядку.

Пункт №1: Ф-ция $P_1(m, g, U, R)$

Она получает на вход вес m шара,

$g = 9.81 \frac{m}{c^2}$, μ , R - кол-во штрихов.

Путем перебора, начиная с 1 штриха,

она проверяет будет ли

$$F_{тр} < F_T = mgk \rightarrow k \text{ - кол-во штрихов. Если это}$$

не так, то добавляем k к 1 , а если это так, то возвращаем текущее k .

Пункт №2: Ф-ция $P_2(k_0, g, U, R, J)$.

Тут мною было принято решение, что

вот в такой ситуации

т.к. шар перестает касаться

опоры и в этот момент расстояние

до точки касания опоры

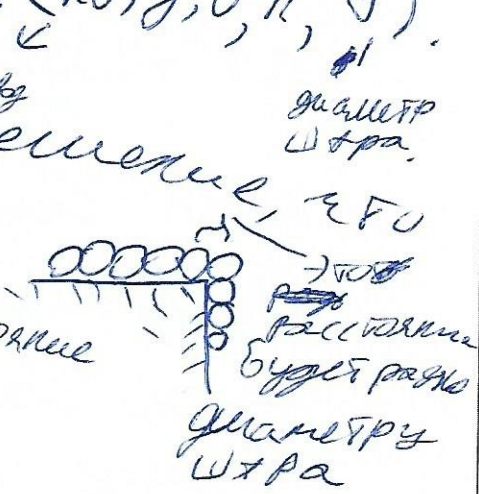
следующим шаром $= J$.

Вернемся к Ф-ции. Она находит a_1

учитывая, что начальная скорость нет у тел.

t_1 - время забр. на это и V_1 - скорость

теда получившуюся.





ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

После удара тоже самое но для шара следующий, но скорость уже есть (V_1) поэтому учитываем ее в расчетах.

возвращает ф-ия t_1 и $t_1 + t_2$.

Пункт N3; ф-ия ~~ВВ~~ P3 (k, g, r, t)

ф-ия дает тоже самое, что и P2, но ~~иде~~ только для 1 и 2 шара, а для всех. Для 1 шара, считается также как и в P2, а для следующих ~~ис~~ используется ~~и~~ ~~и~~.

чтобы найти время в ф-иях P2 и P3 решается кв. ур. вида:

$$at^2 + 2V_0t - S = 0$$

~~и~~ ~~и~~ скорость тела после падения шара ^{продолжено}
корни считаем с помощью дискриминанта.

Пункт N4: ф-ия P4 ($V_{изг}, g, l$)

просто считаем начала расстояние $= \Delta(n \cdot d; H)$
нужное найти между шаром и опорой а потом находим время за которое это совершается и считаем ~~время~~ скорость по формуле:
 $V = V_{изг} + g \cdot t_n$ → время падения


```

import math

def p1(m, g, u, n):
    for k in range(1, n):
        Ftr = m * g * (n - k) * u
        if Ftr < m * g * k:
            return k

def p2(k0, g, u, n, d):
    a1 = g * k0 / (n-k0)
    t1 = round(math.sqrt(2 * d / a1), 2)
    v1 = a1 * t1
    a2 = g * (k0 + 1) / (n-k0 - 1)
    t2 = min((v1 + math.sqrt(v1 ** 2 + a2 * d)) / a2, (-1 * v1 + math.sqrt(v1
** 2 + a2 * d)) / a2)
    if t2 <= 0:
        t2 = max((v1 + math.sqrt(v1 ** 2 + a2 * d)) / a2, (-1 * v1 +
math.sqrt(v1 ** 2 + a2 * d)) / a2)
    v2 = v1 + a2 * t2
    return t1, t2 + t1

def p3(k0, g, u, n, d):
    a1 = g * k0 / n
    t1 = round(math.sqrt(2 * d / a1), 2)
    v1 = a1 * t1
    tsum = t1
    for i in range(1, n - k0):
        a2 = g * (k0 + i) / n
        t2 = min((v1 + math.sqrt(v1 ** 2 + a2 * d)) / a2, (-1 * v1 +
math.sqrt(v1 ** 2 + a2 * d)) / a2)
        if t2 <= 0:
            t2 = max((v1 + math.sqrt(v1 ** 2 + a2 * d)) / a2, (-1 * v1 +
math.sqrt(v1 ** 2 + a2 * d)) / a2)

        v1 = v1 + a2 * t2
        tsum += t2
    return tsum, v1

def p4(vobs, g, l):
    t2 = min((vobs + math.sqrt(vobs ** 2 + g * l)) / g, (-1 * vobs +
math.sqrt(vobs ** 2 + g * l)) / g)
    if t2 <= 0:
        t2 = max((vobs + math.sqrt(vobs ** 2 + g * l)) / g, (-1 * vobs +
math.sqrt(vobs ** 2 + g * l)) / g)
    v4 = vobs + g * t2
    return v4

m = 2
d = 0.3
u = 0.11
n = 50
g = 9.81
h = 60 * d
V = 1 / 6 * math.pi * pow(d, 3)
p1 = m / V
k0 = p1(m, g, u, n)
t1, t1_2 = p2(k0, g, u, n, d)
tobs, vobs = p3(k0, g, u, n, d)
print(f'p1: k0={k0}\np2: T1={t1} T1-2={round(t1_2, 2)} \np3: T={round(tobs,

```

```

2)} V={round(vobs, 2)}')
l = h - n * d

v4 = p4(vobs, g, l)
print("p4:", "V=", round(v4, 2))
#-----
qwert = tobs
d1 = d
for i in range(1, 1000):
    i *= 0.1
    d1 += i
    m = 1 / 6 * math.pi * pow(d1, 3) * p1
    k01 = p1(m, g, u, n)
    temp = p3(k01, g, u, n, d1)
    print([round(temp[0], 2), round(temp[0] - qwert, 2)], end=" ")
    qwert = round(temp[0], 2)
print()
print("""Как нетрудно заметить изменение времени на все падение имеет почти
арифметическую прогрессию, только в
самом начале имеется меньшее изменение, а дальше дельта держится около 1.13-
1.14 (при увеличении диаметра на 10 см в шаг)""")
d1 = d
for i in range(1, 100000):
    i *= 0.001
    d1 += i
    m = 1 / 6 * math.pi * pow(d1, 3) * p1
    k01 = p1(m, g, u, n)
    temp = p3(k01, g, u, n, d1)
    if k01 == k0 and temp[0] >= 2 * tobs:
        print("при диаметре =", round(d1 * 100, 2), " сантиметров нам
подходит под условия 5 пункта")
        print("отношение времён = ", round(temp[0]/tobs, 2))
        break
    else:
        print('Нет такого значения для диаметра в первых 10 метрах с шагом
проверки в 10 см начиная от 30 см')

```

Просьба большая соблюсти табуляцию символ в символ иначе код не заработает

Олимпиада школьников «Надежда энергетики»

К11 F01	Аннотационно, с использованием ВКС
---------	------------------------------------

№ группы

Место проведения

СА88-29

шифр

← Не заполнять
Заполняется
ответственным
работником

Вариант № 47111

ФАМИЛИЯ

Сухогин

ИМЯ

Михаил

ОТЧЕСТВО

Амфириевич

Дата
рождения

28.06.2005

Класс: 11

Предмет


Компьютерное моделирование

Этап: Заключительный

Работа выполнена на 3 листах

Дата выполнения работы: 25.03.2023
(число, месяц, год)

Подпись участника олимпиады:



Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, общее количество листов, на которых выполнена работа и дату выполнения работы.



ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

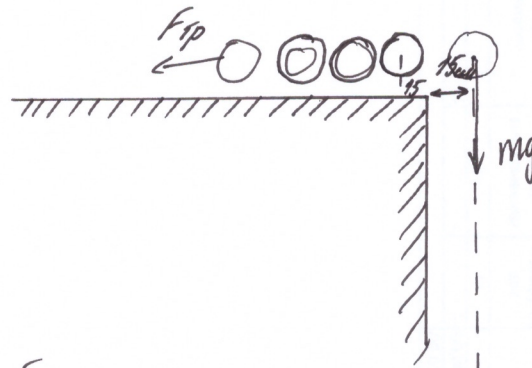
$$m = 2 \text{ кг}$$

$$D = 0,3 \text{ м}$$

$$n = 50 \text{ шар.}$$

$$\mu = 0,11$$

$$g = 9,81 \text{ м/с}^2$$



Решение задачи с точки зрения физики:

Шарики связаны, значит мы можем записать все силы действующие на них. Поскольку мы не угадываем переходные значения шарика задача упрощается до уровня:

Пусть x - кол. шариков висит.

$$F_{сп} F_{сп} - F_{сп} = F_{равн.}$$

$$m_{вис.} g - \mu \cdot m_{леж.} g = m_{вис.} a$$

$$\underline{2 \cdot (x)g - \mu \cdot 2 \cdot (n-x) \cdot g = ma}$$

1) Для движения нам надо чтобы сила трения была меньше тяжести

Объем: 5

$$F_{сп} < F_{тр} \\ m \cdot 2 \cdot (n-x) \cdot g < \mu \cdot 2 \cdot x \cdot g$$

$$m \cdot (n-x) < \mu x$$

$$m n - \mu x < \mu x$$

$$m n < x + \mu x$$

$$m n < x(1 + \mu)$$

$$\frac{m n}{1 + \mu} < x$$

$$\frac{5,5}{1,11} < x$$

$$x > 4,95$$

Значит мин.
кол. шариков
= 5



ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

2) Для решения пункта нужно модеровать ситуацию. Для упрощения (мы не считаем переход с грани на грань) можем сказать, что шар переходит из группы



Значит шару за 15 см надо пройти еще 30 см для перехода в висение. Так они и будут друг друга слышать. переходы из группы в группу. Напишем программу (там все описано)

Для висения ускорение $a = \frac{2 \cdot x \cdot g - m \cdot 2 / (n - x) \cdot g}{m \cdot 2 \cdot x}$

$$S = S_0 + v_0 t + \frac{a t^2}{2}, S_0 = 0$$

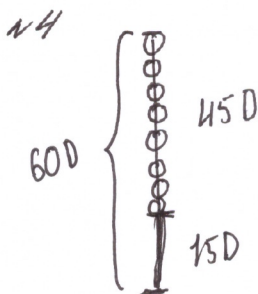
$$\frac{a t^2}{2} + v_0 t + S = 0$$

$$D = (v_0^2) + 4 \cdot 0,5 \cdot a \cdot S$$

$$x_{1,2} = \frac{-v_0 \pm \sqrt{D}}{2 \cdot 0,5 a} \quad (x_2 - \text{отриц.})$$

Из программы:
 первый шар соскочит за $t_{\text{time}} = 7,82$ секунды,
 а следующий через $\Delta t_{\text{time}} = 1,524$ сек,
 т.е. через 9,14 висеть

н3 Из программы видно, что последний шарик упадет через 13,0929 секунды после начала движе., а общая скорость $v = 14,3854019$ м/с



Шарикам надо падасть $15D = 4,5$ метра

Пишем подходящую программу, считаем

скорость: 19,7620 м/с

Время: 13; 43715 сек.



ВНИМАНИЕ! Проверяется только то, что записано
с этой стороны листа в рамке справа

5) В программе есть параметр $rust$ в функции модуляции, он отвечает за размер шариков, меняем его, и получаем, что время $26,20$ сек, т.е. 27. Будет возможно достичь при диаметре шара $1,21$ м. т.е. 1 м 21 см ($1,20$ м мало, точность 2 знака) Ответ: ~~1,20~~ $1,21$ м

Р.С. и снагана магал сгибать массу, в зависимости от диаметра, но масса не влияет на ~~то~~ время. А расстояние не влияет на мин. количество шаров, которое нужно для магала движения

Ответы: 1) 5

2) Время: $7,82$ сек.

Время 2: $9,14$ сек.

3) Время: $13,09$ сек.

Скорость: $11,39$ м/с

4) Скорость: $14,76$ м/с

Время: $13,44$ сек.

5) $1,21$ м

