



МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М.В.ЛОМОНОСОВА

Вариант 3

Место проведения Москва
город

ПИСЬМЕННАЯ РАБОТА

Олимпиада школьников "Ломоносов - 2023"
наименование олимпиады

по космонавтике
профиль олимпиады

Курбачкого Рёдора Андреевича
фамилия, имя, отчество участника (в родительном падеже)

Дата
« 4 » Март 2023 года

Подпись участника
Рёдор

09-77-86-04
(29.1)

Числовые

№ 1 85 (восемьдесят пять)
Ф.С.С. / Сагонов В.В.
Р.С. (Владислав В.Е.)
урочный материал
эта пов

$$x + \sqrt[3]{1-x^3} = \frac{2023}{2022}$$

Скажем, что $\frac{2023}{2022} = k$ где k упростили

$$x + \sqrt[3]{1-x^3} = k$$

$$\sqrt[3]{1-x^3} = k - x$$

$$1-x^3 = (k-x)^3$$

~~$$1-x^3 = k^3 - 3k^2x + 3kx^2 - x^3$$~~

$$1-x^3 = (k-x)(k^2-2kx+x^2)$$

$$1-x^3 = k^3 - 2k^2x + kx^2 - k^2x + 2kx^2 - x^3$$

$$1-x^3 = k^3 - 3k^2x + 3kx^2 - x^3$$

$$3kx^2 - 3k^2x + (k^3-1) = 0$$

это квадратное уравнение относительно x

по теореме Виета:

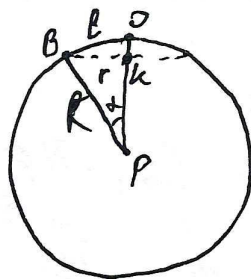
$$\text{сумма корней} = \frac{3k^2}{3k} = k$$

сумма корней = $k \Rightarrow \left(\frac{2023}{2022}\right) = \frac{2023}{2022}$

не доказано наличие корней

Ответ: $\frac{2023}{2022}$
ответ верный

№ 2



Заметим, что окружность на сфере - это окружность на плоскости, но с меньшим радиусом.

на рисунке $BO = R$ - радиус сферы, где k - радиус окружности в сечении.

а $Bk = r$ - радиус окружности в трехмерном пространстве.

$\angle BPO = \alpha$; $BP = R$ - радиус сферы.

Из прямоугольного тр-ка выр: $\sin \alpha = \frac{r}{R}$ Чисовик

$$l = 2\pi R \cdot \frac{\alpha(\text{рад})}{2\pi} = R \cdot \alpha$$

По условию: $\frac{2\pi r}{2\pi l} = 0,9999$; $2\pi r$ - реальная длина
 $2\pi l$ - длина, которая должна намотаться.

$$\Rightarrow \frac{r}{l} = 0,9999$$

$$\frac{r}{R \cdot \alpha} = 0,9999$$

$$\frac{\sin \alpha}{\alpha} = 0,9999 \quad \text{- верно}$$

Это уравнение однозначно определяет угол α , т.к. $\alpha \in [0; \frac{\pi}{2}]$
 ввиду ступенчатой калькулятора, который можно считать $\sin \alpha$,
 я предположу способ нахождения этого α .

α можно искать с помощью калькулятора бисекции левыми,
 таким образом будет получен достаточный точный результат
 Возьмем $\alpha_0 = \frac{\pi}{6}$ найдем значение функции $f(\alpha) = \frac{\sin \alpha}{\alpha}$

~~Видно, что значение функции при $\alpha_0 = \frac{\pi}{6}$ меньше 0,9999, значит, надо брать α_1 больше α_0 .~~

т.к. $\alpha_0 = \frac{\pi}{6}$; $f(\alpha) = \frac{3}{11} < 0,9999$

$$\alpha_1 = \frac{\pi}{12}$$

Далее, на каждом шаге если $f(\alpha_i) < 0,9999$,
 шаг рассматривается $\alpha_{i+1} = \alpha_i - \frac{\alpha_i}{2^i}$ - это метод, который вы я искал α
 для помощи калькулятора

если $f(\alpha_i) > 0,9999$; $\alpha_{i+1} = \alpha_i + \frac{\alpha_i}{2^i}$ Метод, почему метод бисекции

мы приближаемся к истинному α достаточного близко, и
 можно считать, что α найден таким образом.

наибольшая длина пути по окружности - это длина пути
 по окружности, с таким же радиусом, т.е. $l_{max} = \pi R$

$$l_{max} = \pi R = \frac{\pi l}{\alpha}$$

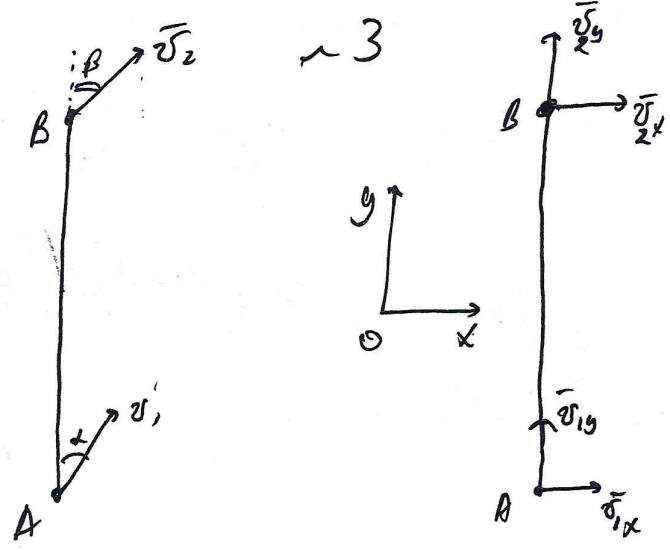
ответ: $l_{max} = \frac{\pi l}{\alpha}$

Верная формула



09-77-86-04
(29.1)

Читовик



Зная углы α и β , можно найти проекции скорости на ось Ox и Oy :

$$\vec{v}_{2x} = \vec{v}_2 \cdot \sin \beta = v_2 \cdot \frac{\sqrt{2}}{2}$$

$$\vec{v}_{2y} = v_2 \cdot \cos \beta = v_2 \cdot \frac{\sqrt{2}}{2}$$

$$\vec{v}_{1x} = v_1 \cdot \sin \alpha = v_1 \cdot \frac{1}{2}$$

$$\vec{v}_{1y} = v_1 \cdot \cos \alpha = v_1 \cdot \frac{\sqrt{3}}{2}$$

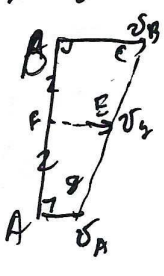
Поскольку, что палка движется и расстояние между её концами не изменяется $\Rightarrow v_{1y} = v_{2y}$

$$v_1 \cdot \frac{\sqrt{3}}{2} = v_2 \cdot \frac{\sqrt{2}}{2}$$

$$\Rightarrow v_2 = \frac{\sqrt{3}}{\sqrt{2}} v_1$$

$$\Rightarrow v_{2x} = v_2 \cdot \frac{\sqrt{2}}{2} = \frac{\sqrt{3}}{\sqrt{2}} \cdot \frac{\sqrt{2}}{2} v_1 = \frac{\sqrt{3}}{2} v_1$$

Скорость центра по оси $Oy = v_{1y} = \frac{\sqrt{3}}{2} v_1$
 найдем скорость центра по оси Ox :



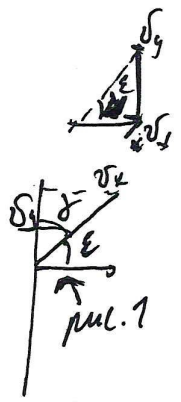
$ABCD$ - трапеция, а FE - её средняя линия
 \Rightarrow длина $FE = \frac{BC + AD}{2}$ \Rightarrow при этом FE - это скорость центра
 по оси $Ox \Rightarrow v_{3x} = \frac{v_{2x} + v_{1x}}{2} = \frac{\frac{\sqrt{3}}{2} v_1 + \frac{1}{2} v_1}{2}$

$$= \frac{\sqrt{3} + 1}{4} v_1 \approx 0,683 v_1$$

шаг 1

⇒ угол между вектором скорости челнока и вертикали

равен $\arctg \frac{v_y}{v_x} = \arctg \frac{\frac{\sqrt{3}}{2} v_1}{\frac{\sqrt{3}+1}{4}} = \arctg \frac{2\sqrt{3}}{\sqrt{3}+1}$ — это же тот угол, на который нужно (см. рис. 7)



$\angle \epsilon = \arctg \frac{2\sqrt{3}}{\sqrt{3}+1}$

$\angle \delta = \arctg \frac{\sqrt{3}+1}{2\sqrt{3}} = \arccos \frac{v_x}{v_y}$

Ответ: $\angle \delta = \arccos \frac{\sqrt{3}+1}{2\sqrt{3}}$

Список ответов

Программа на языке Python.

```
1. M, N = map(int, input().split())
```

```
2. for i in range(M):  
for j in range(N):
```

```
3. a = [] ; b = [] ; c = []
```

```
4. for i in range(N):
```

```
    a.append(list(map(int, input().split())))
```

```
5. for i in range(M):
```

```
    for j in range(N):
```

```
        if a[i][j] == 1:
```

```
            while
```

```
                k = 1
```

```
                while a[i+k][j] == 1:
```

```
                    k += 1
```

```
                p = 1
```

```
                while a[i][j+p] == 1:
```

```
                    p += 1
```

```
                    count = 0
```

```
                    for u in range(k):
```

```
                        for q in range(p):
```

```
                            if a[i+u][j+p+q] == 1:
```

```
                                count += 1
```

```
                    c.append(count)
```

```
                b.append(max(c))
```

```
                c = []  
                count = 0
```

```
print(max(b))
```

каждый раз, что вычисляется
идея: ну вот ну вот ну вот

09-77-86-04
(29.1)

л 5 Штотваи

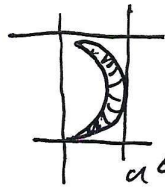
а) по условию, звезда засвечивает только 1 пиксель, а 2 звезды не засвечивают 2 соседних.

Значит можно рассмотреть все засвеченные пиксели и, если они не с какими группами засвеченных пикселей не граничат, считать их незасвеченными. Т.е. такой пиксель имеет значение с 1 на 0

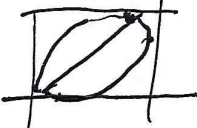
в) засвеченные пиксели останутся только от Луны.

Будем считать, что 90% пикселей, которые попали на луну, были засвечеными.

Круги крайние тогда, которые засвечены со всех сторон. Другими, что Луна лежит в прямоугольнике, ограниченном этими, проходящим через крайние



тогда. Таким образом мы получим 4 точки. Заметим, что независимо от наклона камеры, диаметр Луны на фото будет равен диаметру круга



самыми дальними из этих четырех точек тогда угол между самыми дальними точками = диаметру Луны на фото = угол = 30' почему?

Опишем изображение углового размера как "ширину" Луны при съемке, тогда Луна сфотографирована вертикально координаты дальних точек = (x_1, y_1) и (x_2, y_2)

координаты точки O = $(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2})$



Будем идти по строке $\frac{x_1+x_2}{2}$ в сторону крайней точки пока не встретим засвеченный пиксель, или наоборот, если точка O - засвечена.

(допомини и п.7: центр лежит на отрезке между точками, при этом - на его середине)

методика

Программа на языке Python для нахождения
узора в общем виде и условия равенства для
каждого случая:

```
N = int(input())
a = [] ; b = [] ; c = [] ; d = [] ; g = []
```

~~for i in range(200):~~
~~a.append(0)~~
~~g.append(0)~~
~~for i in range(200):~~
~~b.append(a)~~
~~d.append(g)~~
~~for i in range(N):~~
~~x, y = map(int, input().split())~~
~~b[y][x] = 1~~
~~d[x][y] = 1~~
~~for i in range(200):~~
~~for j in range(200):~~

```
for i in range(200):
```

```
    a.append(0)
```

```
    g.append(0)
```

```
for i in range(200)
```

```
    b.append(a)
```

```
    d.append(g)
```

```
for i in range(N):
```

```
    x, y = map(int, input().split())
```

```
    b[y][x] = 1
```

```
    d[x][y] = 1
```

```
for i in range(200):
```

```
for j in range(200):
```

```
for i in range(1; 199)
```

```
    for j in range(1; 199):
```

```
        if b[i][j+1] + b[i][j-1] + b[i-1][j] + b[i+1][j] +
```

```
+ b[i+1][j+1] + b[i-1][j-1] + b[i-1][j+1] + b[i+1][j-1] == 0:
```

```
            b[i][j] = 0
```

```
for i in range(200):
```

```
    if b[i] != [0]*200:
```

```
for i in range(200):
```

```
    k = b.index(1)
```

```
    c.append([1]*k)
```

```
    break
```

```
for i in range(200, 0, -1) цистовий
    if b[i] < [0] - 200:
        k = b.index(i)
        c.append([i, k])
        break
```

```
for i in range(200):
    if d[i] < [0] - 200:
        k = d.index(i)
        c.append([i, k])
        break
```

```
for i in range(200; 0; -1)
for
    if d[i] < [0] - 200:
        k = d.index(i)
        c.append([i, k])
        break
```

```
for i in range(200):
    p = append(c, [i])
    q = append(c, [i])
    m = max([c[0][0], c[1][0]])
    m = max([c[0][1], c[1][1]])
```

~~m = 0~~
 $x_1 = 0$
 $x_2 = 0$
 $y_1 = 0$
 $y_2 = 0$

```
for i in range(200):
for for i in range(200):
for for i in range(200):
```

$$if (c[i][0] - c[j][0])^2 + (c[j][1] - c[i][1])^2 > m$$

$$m = (c[i][0] - c[j][0])^2 + (c[j][1] - c[i][1])^2$$

$$x_1 = [c[j][0]]$$

$$y_1 = [c[i][1]]$$

$$x_2 = [c[j][0]]$$

$$y_2 = [c[j][1]]$$

$$W = \left(\frac{x_1 + x_2}{2}; \frac{y_1 + y_2}{2} \right)$$

print(W)

Титов

16

В невесомости у человека неправильно работает вестибулярный аппарат. Задержка происходит по той же причине, почему после быстрого вращения, человек не может удержать взгляд в одной точке, а он «плывёт».

Задерживание взгляда прямо связано с работой вестибулярного аппарата. ⇒ В невесомости, где у космонавтов проблема (увлечения (земной) работой вестибулярного аппарата, при поворачивании головы, взгляд не остаётся в той же точке, методы устранения последствий:

1. Можно вращать станцию с некоторой скоростью, таким образом космонавты будут чувствовать себя как на земле, здесь возникает центробежная сила, но таким способом слишком сложен в реализации.

2. Если космонавт зафиксировать голову и будет только поворачивать взгляд, то эта проблема станет минимальной. Для этого нужно сделать приборную панель и рычаги управления близко друг к другу, чтобы можно было не вращать головой, а только поворачивать взгляд. Этот способ наиболее прост и достаточно прост в реализации.

Есть верхнее направление и вернее жб

впрямую

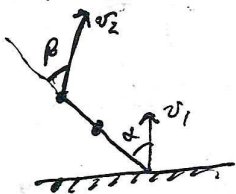
/ Космонавты
очень поперекКосмонавты
поперек

$$v_1 + v_2 = \frac{1}{2} + \frac{\sqrt{3}}{2} = \dots$$

$$(k-x)(k^2-2kx-x^2) = k^3 - 2k^2x - kx^2 - k^2x - 2kx^2 + x^3 = k^3 - 3k^2x + 3kx^2 - x^3$$

Черновики

$$x + \sqrt[3]{1-x^3} = \frac{2023}{2022}$$



$$1-x^3$$

$$x=1$$

$$1-x^3 | x-1$$

$$\begin{array}{r} -x^3+1 | x-1 \\ \hline -x^3+x^2-x-1 \\ \hline -x^2+x+1 \\ \hline -x^2+x+1 \\ \hline -x+1 \end{array}$$

$$\frac{2023}{2022} - x$$

$$x + \sqrt[3]{(x-1)(-x^2-x-1)} = x - \sqrt[3]{(x-1)(x^2+x+1)}$$

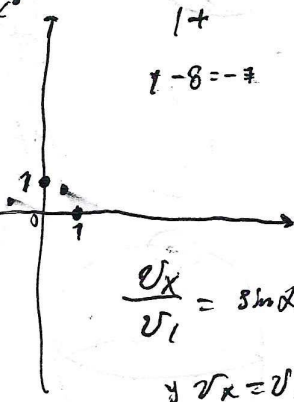
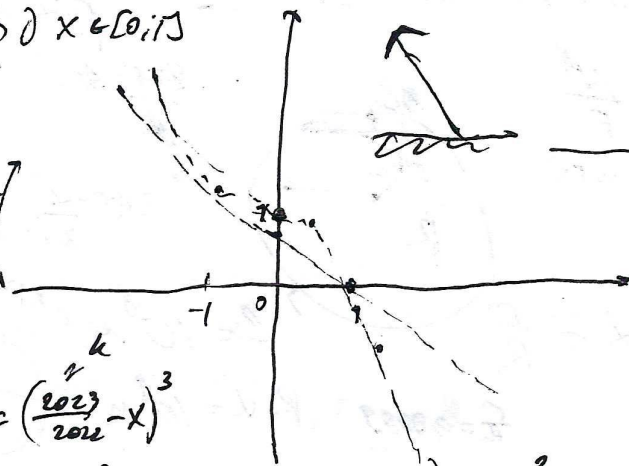
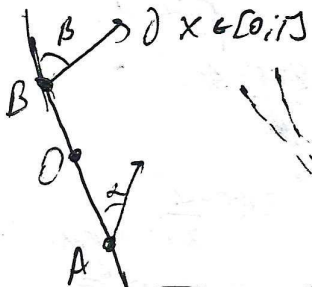


$$\sqrt[3]{\frac{1}{8}}$$

$$1 + 7-8 = -7$$

$$2022x + 2022\sqrt[3]{1-x^3} = 2023$$

$$\sqrt[3]{1-x^3}$$



$$\frac{v_x}{v_1} = 8 \sin \alpha$$

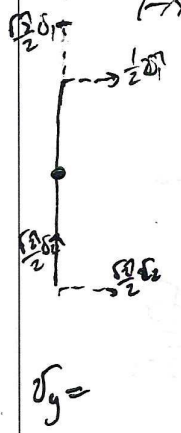
$$v_x = v_1$$

$$1-x^3 = \left(\frac{2023}{2022} - x\right)^3$$

$$1-x^3 = (k-x)^3 = (k-x)(k^2-2kx+x^2) = k^3 - 3k^2x + 3kx^2 - x^3 = k^3 - 3k^2x + 3kx^2 - x^3$$

$$1-x^3 = k^3 - 3k^2x + 3kx^2 - x^3$$

Сумма углов = $3k^2$



$$3kx^2 - 3kx + (k^3-1) = 0$$

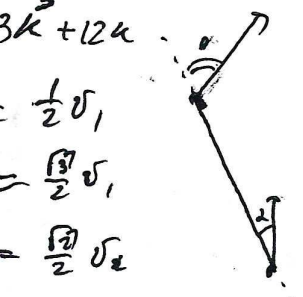
$$9k^4 - 12k(k^3-1) = -3k^3 + 12k$$

$$v_x = v_1 \cdot \sin \alpha = \frac{1}{2} v_1$$

$$v_y = v_1 \cdot \cos \alpha = \frac{\sqrt{3}}{2} v_1$$

$$v_{2x} = v_2 \cdot \sin \alpha = \frac{\sqrt{3}}{2} v_2$$

$$v_{2y} = \frac{\sqrt{3}}{2} v_2$$

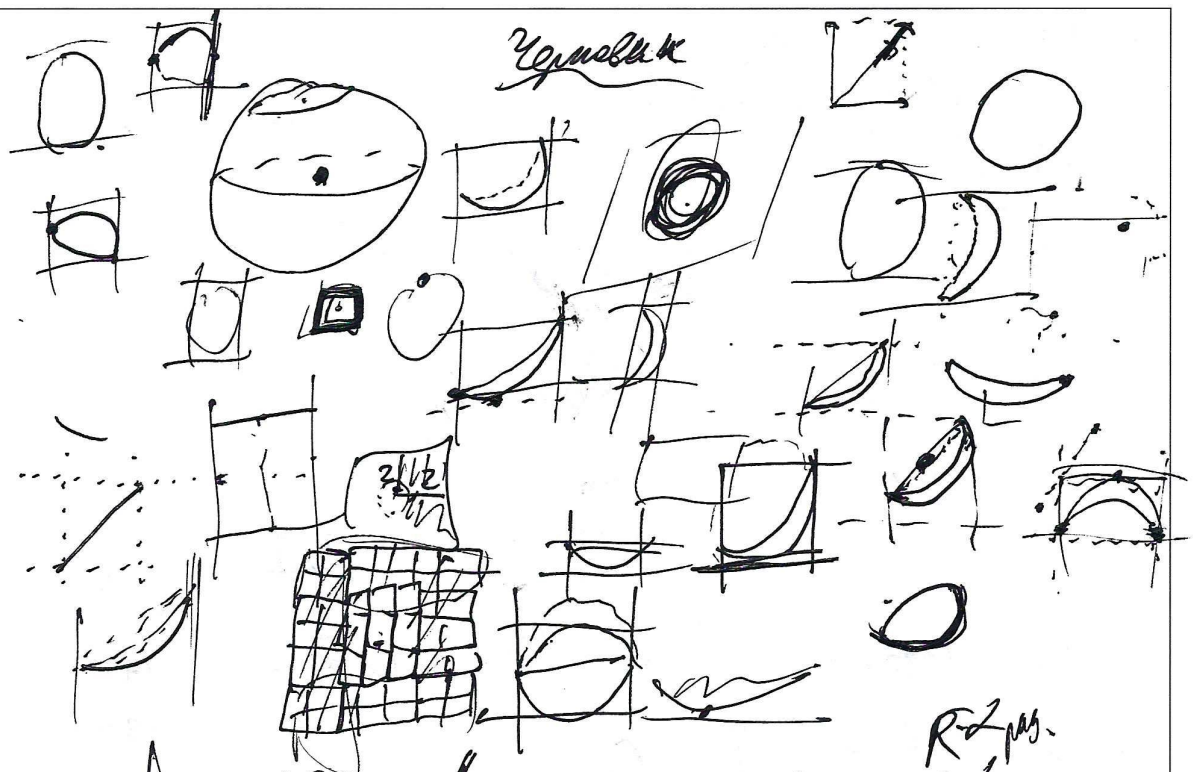


3,464
2,732

$$v_1 = \frac{1}{2} v_2$$

$$v_2 = \frac{1}{2} v_1$$

Черновик



MF $\frac{MF}{F}$

$$\frac{2\pi r \cdot d}{2\pi} = r \cdot d$$



$R \cdot d = \mu s$

$$l = \frac{2\pi R \cdot d}{2\pi}$$

$$\frac{2\pi r \cdot d}{360} = 2\pi \mu s$$

$$\sin \alpha = \frac{r}{R}$$

$l = 10^9 \text{ m}$

$R \cdot d = 10^9 \text{ m}$

$\Rightarrow \underline{\underline{\epsilon = 0.9999}}$

$$\sin \alpha = 0.9999 \cdot \alpha$$

$$\frac{r}{R \cdot d} = 0.9999$$

$$R \cdot \sin \alpha = 0.9999 \cdot l$$

$$r = 0.9999 \cdot l = 9999 \cdot 10^5 \text{ m}$$

$\frac{\sin \alpha}{\alpha} = 0.9999$

$$(x-1)(x-3) = x^2 - 3x$$

$$\frac{\frac{1}{2}}{\frac{\pi}{63}} = \frac{3}{\pi} \frac{\sqrt{3}}{2}$$

$$= \frac{\sqrt{3} + 1}{4}$$

$$= \frac{2\sqrt{3}}{2\sqrt{3} + 1}$$