

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

ОТЧЕТ
О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ
«ДИНАМИКА СИСТЕМЫ»
ПО ДИСЦИПЛИНЕ «ТЕОРЕТИЧЕСКАЯ МЕХАНИКА И
ОСНОВЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ»
ВАРИАНТ ЗАДАНИЯ 24

Выполнил(а) студент группы М8О-208Б-20

Попов Матвей Романович _____
подпись, дата

Проверил и принял

Доцент каф. 802, Чекина Е.А. _____
подпись, дата

с оценкой _____

Москва, 2021

Задание (ЛР №3): построить анимацию движения системы, а также графики законов движения системы (поэкспериментировать с параметрами системы). Исследовать на устойчивость. Показать правильность работы своей механической системы.

Код программы:

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.animation import FuncAnimation
import math
from scipy.integrate import odeint

t = np.linspace(0, 10, 1000)

print('Enter m1 of the figure, m2 of the point, radius and angle\n')

m1 = int(input())#20
m2 = int(input())#5
r = float(input())#0.4
g = 9.81

#коэффициенты для уравнения Лагранжа
def odesys(y, t, m1, m2, r, g):
    dy = np.zeros(4)
    dy[0] = y[2]
    dy[1] = y[3]

    a11 = m1 + m2
    a12 = m2 * r * np.cos(y[1])
    a21 = np.cos(y[1])
    a22 = r

    b1 = m2 * r * np.sin(y[1]) * (y[3] ** 2)
    b2 = -g * np.sin(y[1])

    dy[2] = (b1 * a22 - b2 * a12)/(a11 * a22 - a12 * a21)
    dy[3] = (b2 * a11 - b1 * a21)/(a11 * a22 - a12 * a21)
    return dy

s0 = 0
phi0 = float(input()) #0
ds0 = 0
dphi0 = 2
y0 = [s0, phi0, ds0, dphi0]

Y = odeint(odesys, y0, t, (m1, m2, r, g))

x = Y[:, 0]
phi = Y[:, 1]

O = x
#определяем точки трапеции
XA = 2.5 + x
YA = 1.5
#определяем точку
XB = 2.5 + x + r * np.sin(phi)
```

```

YB = 1.5 - r * np.cos(phi)
Xtr = np.array([1, 1.5, 3.5, 4, 1])
Ytr = np.array([0, 3, 3, 0, 0])

fig = plt.figure(figsize = [1, 1])
ax = fig.add_subplot(1, 2, 1)
ax.set(xlim = [0, 10], ylim = [0, 5])
ax.set_aspect('equal')

TRAP = ax.plot(0[0] + Xtr, Ytr)[0]
LAB = ax.plot([XA[0], XB[0]], [YA, YB[0]])[0]
PA = ax.plot(XA[0], YA, marker = 'o')[0]
PB = ax.plot(XB[0], YB[0], marker = 'o', markersize = 5)[0]

#строим графики
ax1 = fig.add_subplot(4, 2, 2)
ax1.plot(x)
plt.ylabel('Vx values')

ax1 = fig.add_subplot(4, 2, 4)
ax1.plot(phi)
plt.ylabel('Vy values')

def anime(i):
    PA.set_data(XA[i], YA)
    PB.set_data(XB[i], YB[i])
    LAB.set_data([XA[i], XB[i]], [YA, YB[i]])
    TRAP.set_data(0[i] + Xtr, Ytr)
    return [PA, PB, TRAP]

anima = FuncAnimation(fig, anime, frames = 1000, interval = 10)

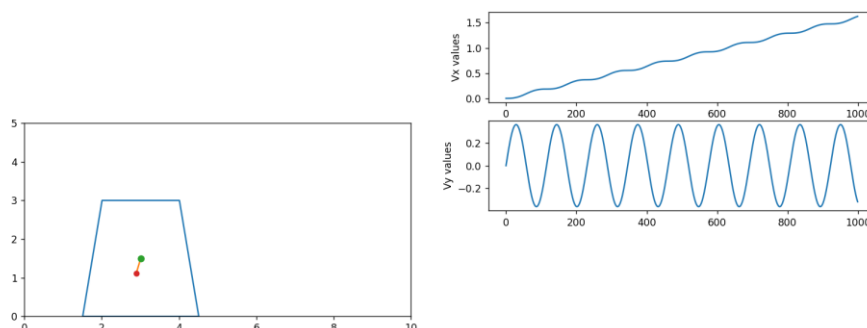
plt.show()

```

Результат работы программы (ЛР №4):

При $m_1 = 20$, $m_2 = 5$, $r = 0.4$, $\phi_0 = 0$, где m_1 — масса трапеции, m_2 — масса точки, r — радиус вращения точки, ϕ_0 — начальный угол отклонения.

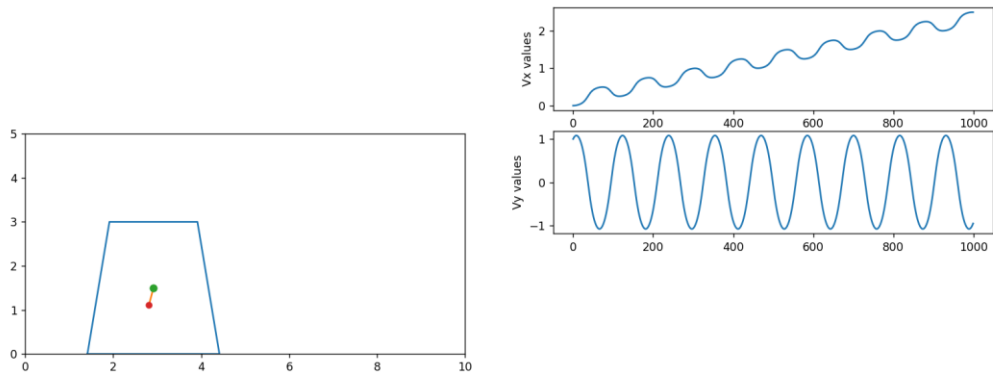
Figure 1



Результат: колебания трапеции слабо заметны, движение только вправо.

При $m_1 = 20$, $m_2 = 20$, $r = 0.4$, $\phi_0 = 1$

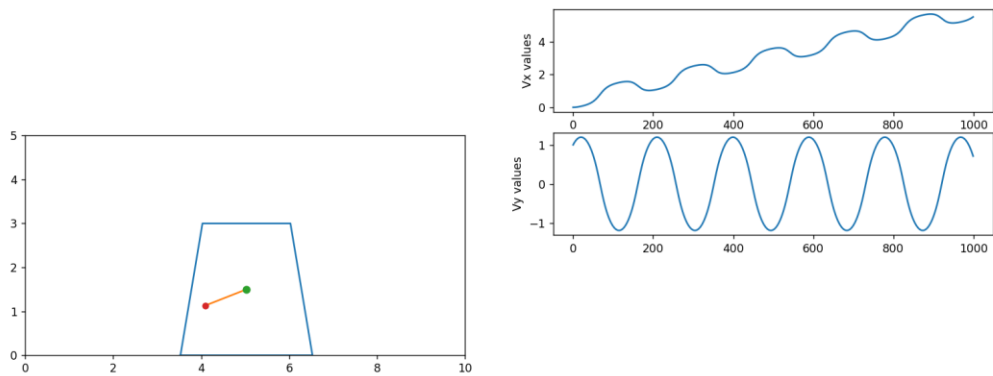
Figure 1



Результат: ярко выраженный колебательный характер, трапеция немного отодвигается влево, затем следует рывок вправо

При $m_1 = 20$, $m_2 = 20$, $r = 1$, $\phi_0 = 1$

Figure 1



Результат: колебания усилились

Вывод: проделав лабораторную работу, я построил анимацию движения системы, построил графики законов её движения и убедился в её устойчивости.