ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

**ОТЧЕТ**

**О ВЫПЛОНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

**«Анимация системы»**

**ПО ДИСЦИПЛИНЕ «ТЕОРЕТИЧЕСКАЯ МЕХАНИКА И ОСНОВЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ»**

**ВАРИАНТ ЗАДАНИЯ №24**

Выполнил(а) студент группы М8О-212Б-22

Мамонтов Егор Олегович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Проверил и принял

Зав. каф. 802, Бардин Б.С.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

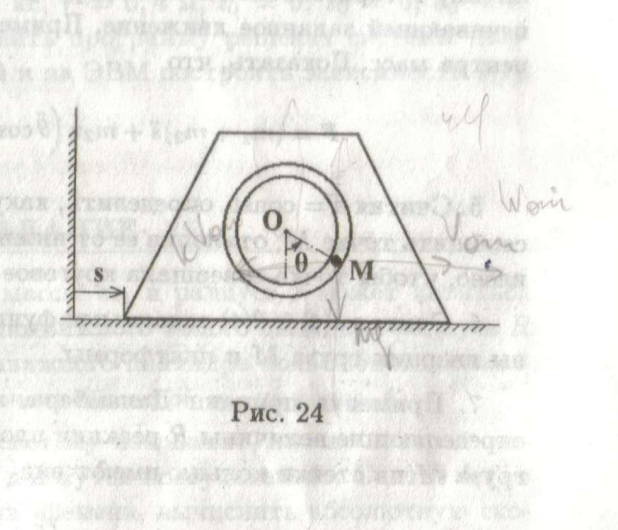
с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва, 2023

*Задание:*

Реализовать анимацию движения механической системы

*Задание варианта:*



Невесомая трубка, выгнутая в форме кругового кольца радиуса в r, закреплена на платформе, которая имеет массу m1 и может скользить без трения по горизонтальной плоскости. В трубке движется без трения точечный груз М массы m2.

*Программа:*

import numpy as np

import sympy as sp

import math

import matplotlib.pyplot as plt

from matplotlib.animation import FuncAnimation

def Trapez(x0, y0):

    PX = [x0 - 10, x0 - (2/3) \* 10, x0 + (2/3) \* 10, x0 + 10, x0 - 10]

    PY = [y0 - 7.5, y0 + 10, y0 + 10, y0 - 7.5, y0 - 7.5]

    return PX, PY

alpha = math.pi

t = sp.Symbol('t')

s = sp.sin(t)

teta = sp.cos(t)

r = 5

X\_trap = s \* sp.cos(alpha) + 0.8

Y\_trap = 7.5

VmodSignTrap = sp.diff(s, t)

VX\_trap = VmodSignTrap \* sp.cos(alpha)

VY\_trap = sp.sin(alpha) \* VmodSignTrap

xA = X\_trap - r \* sp.sin(teta)

yA = Y\_trap + r \* sp.cos(teta)

omega = sp.diff(teta, t)

VxA = VX\_trap - omega \* r \* sp.cos(teta)

VyA = VY\_trap - omega \* r \* sp.sin(teta)

F\_X\_trap = sp.lambdify(t, X\_trap)

F\_VX\_trap = sp.lambdify(t, VX\_trap)

F\_VY\_trap = sp.lambdify(t, VY\_trap)

F\_Teta = sp.lambdify(t, teta)

F\_XA = sp.lambdify(t, xA)

F\_YA = sp.lambdify(t, yA)

F\_VXA = sp.lambdify(t, VyA)

F\_VYA = sp.lambdify(t, VyA)

T = np.linspace(0, 20, 1000)

X\_trap = F\_X\_trap(T)

VX\_trap = F\_VX\_trap(T)

VY\_trap = F\_VY\_trap(T)

Teta = F\_Teta(T)

XA = F\_XA(T)

YA = F\_YA(T)

VXA = F\_VXA(T)

VYA = F\_VYA(T)

fig = plt.figure(figsize=(17, 10))

ax1 = fig.add\_subplot(1, 2, 1)

ax1.axis('equal')

ax1.set(xlim=[X\_trap.min() - 20, X\_trap.max() + 20], ylim=[Y\_trap - 20, Y\_trap + 20])

ax1.plot([X\_trap.min() - 10, X\_trap.max() + 10], [-(X\_trap.min() - 10) \* sp.tan(alpha), -(X\_trap.max() + 10) \* sp.tan(alpha)], 'black')

PrX, PrY = Trapez(X\_trap[0], Y\_trap)

Trap = ax1.plot(PrX, PrY, 'black')[0]

radius = ax1.plot([X\_trap[0], XA[0]], [Y\_trap, YA[0]], 'black')[0]

varphi = np.linspace(0, 2 \* math.pi, 40)

R1 = r - 5 / 15

R2 = r + 5 / 15

Point = ax1.plot(XA[0], YA[0], marker='o', markersize=10, color='green')[0]

circle\_1 = ax1.plot(X\_trap[0] + R1 \* np.cos(varphi), Y\_trap + R1 \* np.sin(varphi), 'black')[0]

circle\_2 = ax1.plot(X\_trap[0] + R2 \* np.cos(varphi), Y\_trap + R2 \* np.sin(varphi), 'black')[0]

ax2 = fig.add\_subplot(3, 2, 2)

ax2.plot(T, VX\_trap)

ax2.set\_ylabel('VX Trapeze')

ax3 = fig.add\_subplot(3, 2, 4)

ax3.plot(T, VXA)

ax3.set\_ylabel('VX Point')

ax4 = fig.add\_subplot(3, 2, 6)

ax4.plot(T, VYA)

ax4.set\_ylabel('VY Point')

plt.subplots\_adjust(wspace=0.3, hspace=0.7)

def foo(i):

    PrX, PrY = Trapez(X\_trap[i], Y\_trap)

    Trap.set\_data(PrX, PrY)

    radius.set\_data([X\_trap[i], XA[i]], [Y\_trap, YA[i]])

    Point.set\_data(XA[i], YA[i])

    circle\_1.set\_data(X\_trap[i] + R1 \* np.cos(varphi), Y\_trap + R1 \* np.sin(varphi))

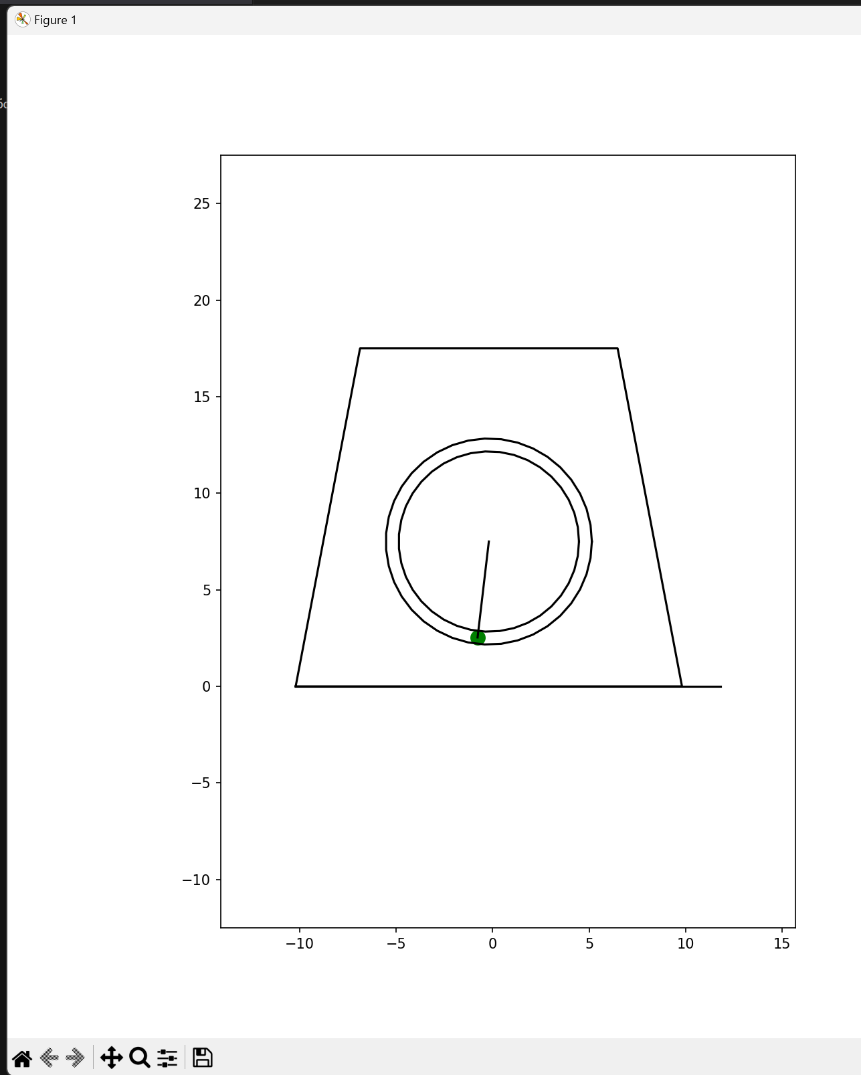
    circle\_2.set\_data(X\_trap[i] + R2 \* np.cos(varphi), Y\_trap + R2 \* np.sin(varphi))

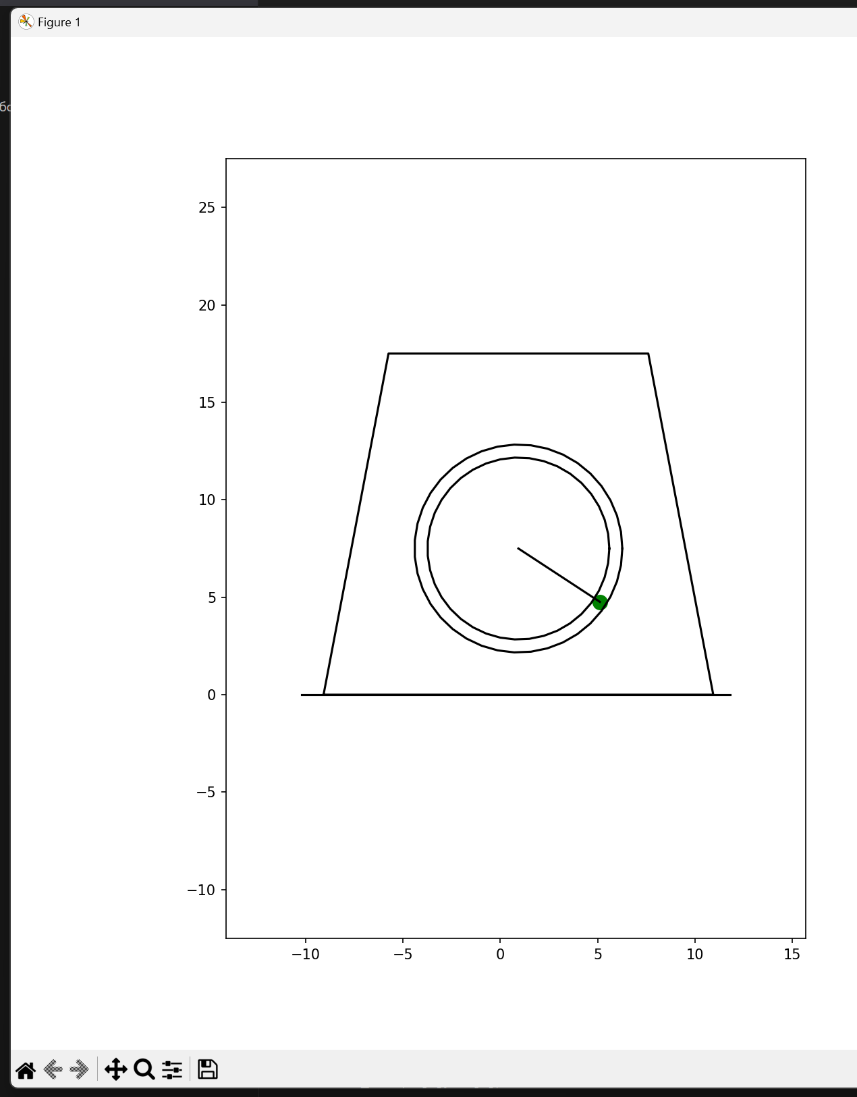
    return Trap, Point, radius, circle\_1, circle\_2

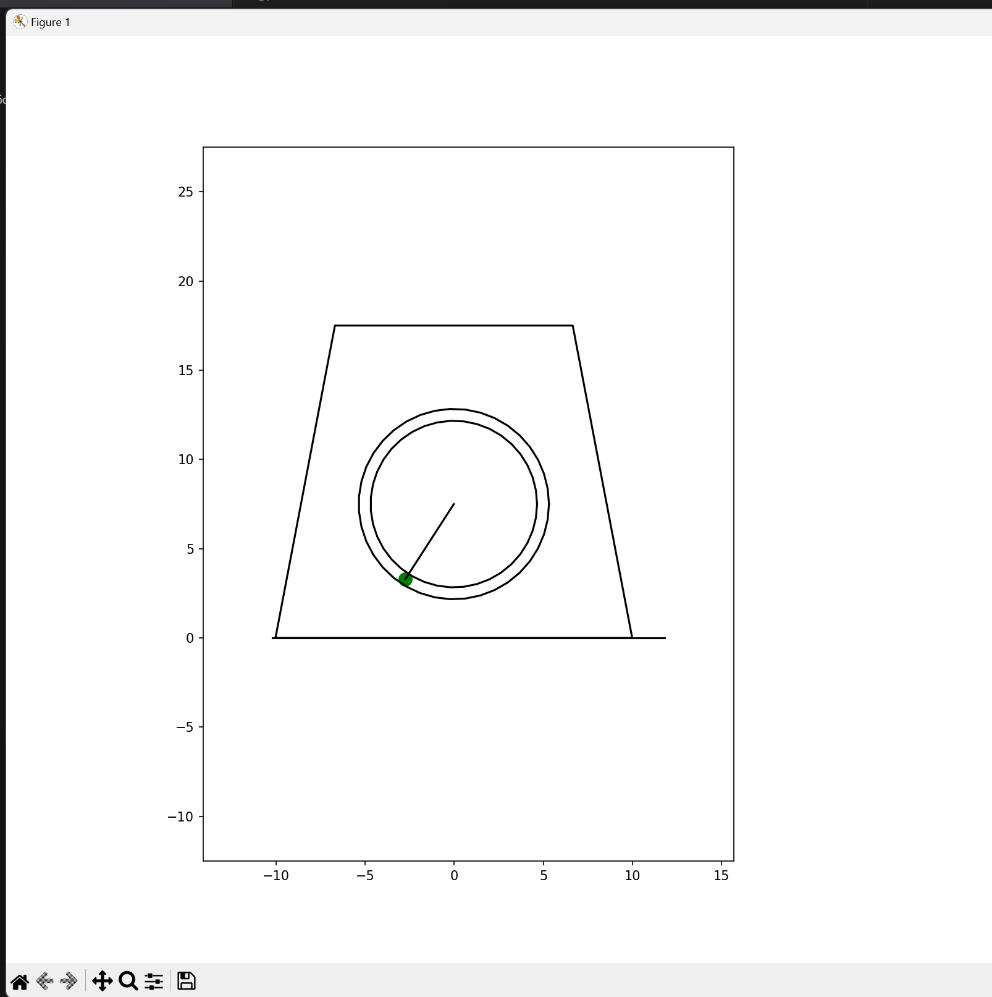
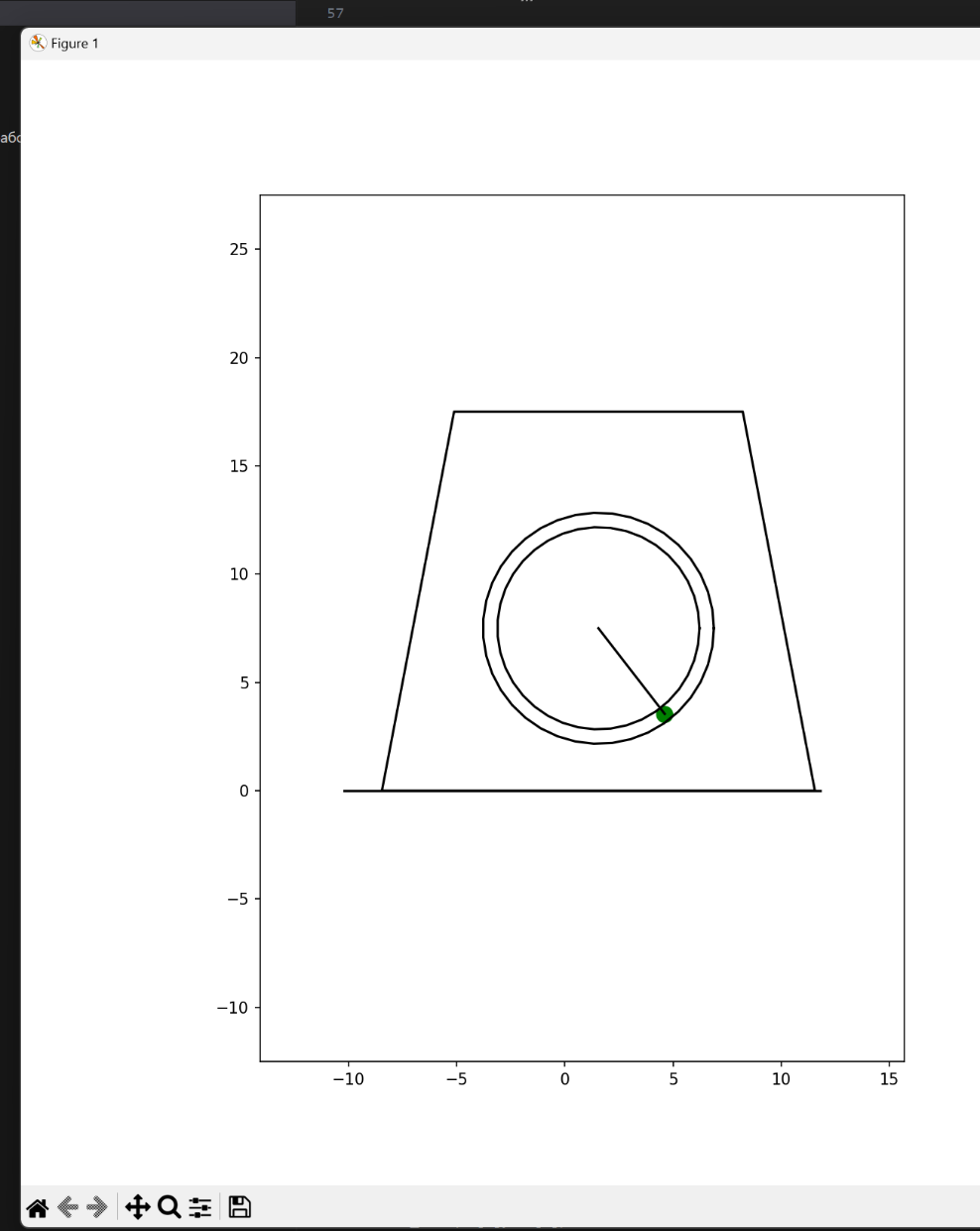
kino = FuncAnimation(fig, foo, frames=1000, interval=0.01, blit=True)

plt.show()

*Результат работы программы:*







*Вывод:*

Проделав лабораторную работу, я построил анимацию движения системы и построил графики законов её движения.