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 №27**

Выполнил(а) студент группы М8О-201Б-22

Чибугаев И.А..\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Проверил и принял

Волков Е.В.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

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

Москва, 2024

*Задание:* построить анимацию движения системы с помощью Python.

*Код:*

import math

import numpy as np

import matplotlib.pyplot as plt

from matplotlib.animation import FuncAnimation

R = 5.0  *# радиус большой шестерни*

r = 1.5  *# радиус маленькой шестерни*

omega = 0.5  *# угловая скорость*

iterations = 200

T = np.linspace(0, 10, iterations)

def theta(t):

    return omega \* t

def phi(t):

    return 2.0 \* omega \* t

*# Позиции точки A:*

*# A движется по окружности радиуса (R - r) вокруг O*

*# X\_A(t) = (R - r)\*cos(phi(t))*

*# Y\_A(t) = (R - r)\*sin(phi(t))*

XA = (R - r) \* np.cos(phi(T))

YA = (R - r) \* np.sin(phi(T))

def draw\_circle(center\_x, center\_y, radius=1.0, resolution=100):

    angles = np.linspace(0, 2 \* math.pi, resolution)

    x = center\_x + radius \* np.cos(angles)

    y = center\_y + radius \* np.sin(angles)

    return x, y

fig, ax = plt.subplots()

ax.set\_aspect('equal', adjustable='box')

ax.set\_xlim(-R - 2, R + 2)

ax.set\_ylim(-R - 2, R + 2)

ax.set\_xlabel('X')

ax.set\_ylabel('Y')

ax.set\_title('Анимация системы шестерен')

*# (Шестерня 1)*

gear1\_x, gear1\_y = draw\_circle(0, 0, radius=R)

gear1\_line, = ax.plot(gear1\_x, gear1\_y, 'b')

*# (Шестерня 2)*

gear2\_x, gear2\_y = draw\_circle(XA[0], YA[0], radius=r)

gear2\_line, = ax.plot(gear2\_x, gear2\_y, 'r')

*# кривошип OA*

crank\_line, = ax.plot([0, XA[0]], [0, YA[0]], 'k')

point\_O, = ax.plot(0, 0, 'ko')

point\_A, = ax.plot(XA[0], YA[0], 'ro')

def update(i):

    gear2\_x, gear2\_y = draw\_circle(XA[i], YA[i], radius=r)

    gear2\_line.set\_data(gear2\_x, gear2\_y)

    crank\_line.set\_data([0, XA[i]], [0, YA[i]])

    point\_A.set\_data([XA[i]], [YA[i]])

    return gear2\_line, crank\_line, point\_A

anim = FuncAnimation(fig, update, frames=iterations, interval=50, blit=True)

plt.show()

*Скриншот:*

