ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

**ОТЧЕТ**

**О ВЫПЛОНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

**«АНИМАЦИЯ ТОЧКИ»**

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

**ВАРИАНТ ЗАДАНИЯ №20**

Выполнил(а) студент группы М8О-206Б-22

Постнов Л. А.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Проверил и принял

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

подпись, дата

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

Москва, 2023

*Задание:* построить заданную траекторию, запустить анимацию движения точки, построить стрелки радиус-вектора, вектора скорости, вектора ускорения и радиуса кривизны.

*Закон движения:*

*Текст программы:*

import math

import sympy as s

import matplotlib.pyplot as plt

import numpy as n

from matplotlib.animation import FuncAnimation

step = 1000

t = s.Symbol('t')

T = n.linspace(1, 30, step)

r = 1 + s.cos(t)

phi = 1.25 \* t

x = r \* s.cos(phi)

y = r \* s.sin(phi)

Vx = s.diff(x)

Vy = s.diff(y)

Ax = s.diff(Vx)

Ay = s.diff(Vy)

V = s.sqrt(Vx\*\*2 + Vy\*\*2)

X = n.zeros\_like(T)

Y = n.zeros\_like(T)

VX = n.zeros\_like(T)

VY = n.zeros\_like(T)

AX = n.zeros\_like(T)

AY = n.zeros\_like(T)

for i in n.arange(len(T)):

X[i] = s.Subs(x, t, T[i])

Y[i] = s.Subs(y, t, T[i])

VX[i] = s.Subs(Vx, t, T[i])

VY[i] = s.Subs(Vy, t, T[i])

AX[i] = s.Subs(Ax, t, T[i])

AY[i] = s.Subs(Ay, t, T[i])

fig = plt.figure()

axis = fig.add\_subplot(1, 1, 1)

axis.axis('equal')

axis.set(xlim = [-3, 3], ylim = [-3, 3])

axis.plot(X, Y)

Pnt = axis.plot(X[0], Y[0], marker = 'o')[0]

Vp = axis.plot([X[0], X[0] + VX[0]], [Y[0], Y[0] + VY[0]], 'r')[0]

Ap = axis.plot([X[0], X[0] + AX[0]], [Y[0], Y[0] + AY[0]], 'g') [0]

Rp = axis.plot([0, X[0]], [0, Y[0]], 'b')[0]

def Rot2D(X, Y, Alpha): #матрица поворота для стрелок

RX = X \* n.cos(Alpha) - Y \* n.sin(Alpha)

RY = X \* n.sin(Alpha) + Y \* n.cos(Alpha)

return RX, RY

# массивы для стрелок

arrow\_size = 1

ArrowX = n.array([-0.1 \* arrow\_size, 0, -0.1 \* arrow\_size])

ArrowY = n.array([0.05 \* arrow\_size, 0, -0.05 \* arrow\_size])

ArrowAX = n.array([-0.1 \* arrow\_size, 0, -0.1 \* arrow\_size])

ArrowAY = n.array([0.05 \* arrow\_size, 0, -0.05 \* arrow\_size])

ArrowRX = n.array([-0.1 \* arrow\_size, 0, -0.1 \* arrow\_size])

ArrowRY = n.array([0.05 \* arrow\_size, 0, -0.05 \* arrow\_size])

RArrowX, RArrowY = Rot2D(ArrowX, ArrowY, math.atan2(VY[0], VX[0]))

RArrowAX, RArrowAY = Rot2D(ArrowAX, ArrowAY, math.atan2(AY[0], AX[0]))

RArrowRX, RArrowRY = Rot2D(ArrowRX, ArrowRY, math.atan2(X[0], Y[0]))

VArrow, = axis.plot(RArrowX + X[0] + VX[0], RArrowY + Y[0] + VY[0], 'r')

AArrow, = axis.plot(RArrowAX + X[0] + AX[0], RArrowAY + Y[0] + AY[0], 'g')

RArrow, = axis.plot(ArrowRX + X[0], ArrowRY + Y[0], 'b')

def anim(i):

Pnt.set\_data(X[i], Y[i])

Vp.set\_data([X[i], X[i] + VX[i]], [Y[i], Y[i] + VY[i]])

Ap.set\_data([X[i], X[i] + AX[i]], [Y[i], Y[i] + Y[i]])

Rp.set\_data([0, X[i]], [0, Y[i]])

RArrowX, RArrowY = Rot2D(ArrowX, ArrowY, math.atan2(VY[i], VX[i]))

VArrow.set\_data(RArrowX + X[i] + VX[i], RArrowY + Y[i] + VY[i])

RArrowAX, RArrowAY = Rot2D(ArrowAX, ArrowAY, math.atan2(Y[i], AX[i]))

AArrow.set\_data(RArrowAX + X[i] + AX[i], RArrowAY + Y[i] + Y[i])

RArrowRX, RArrowRY = Rot2D(ArrowRX, ArrowRY, math.atan2(Y[i], X[i]))

RArrow.set\_data(RArrowRX + X[i], RArrowRY + Y[i])

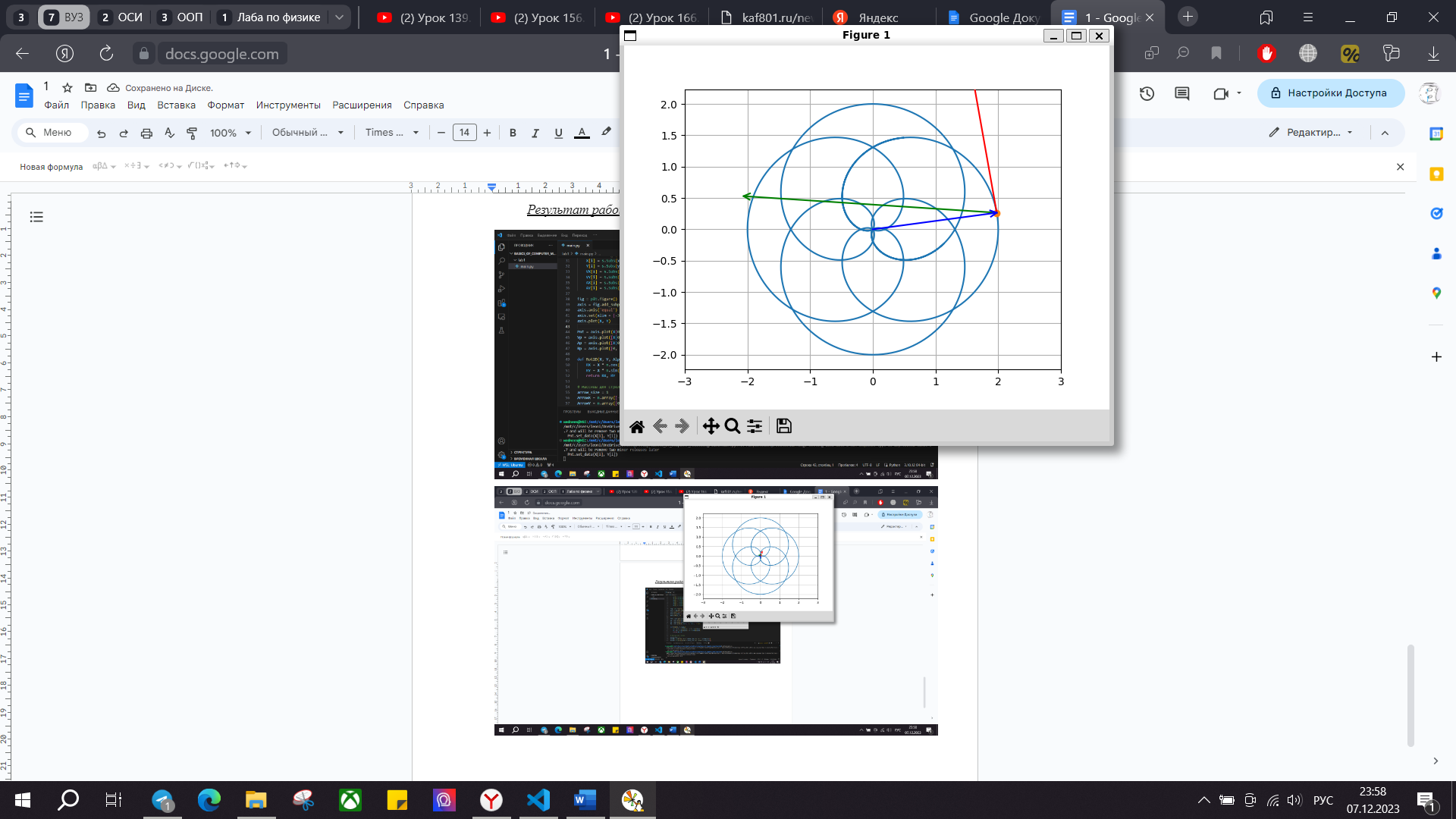
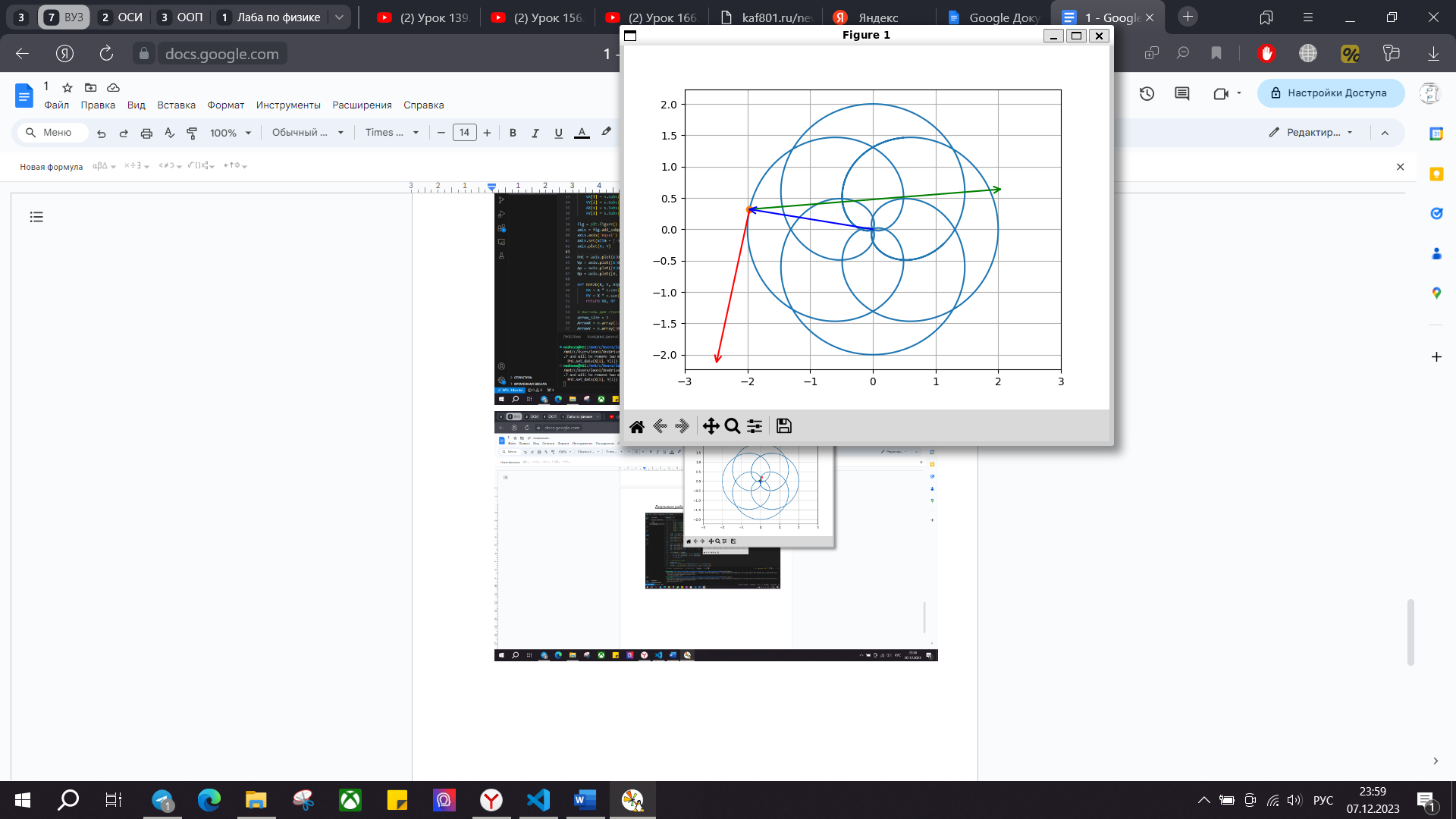
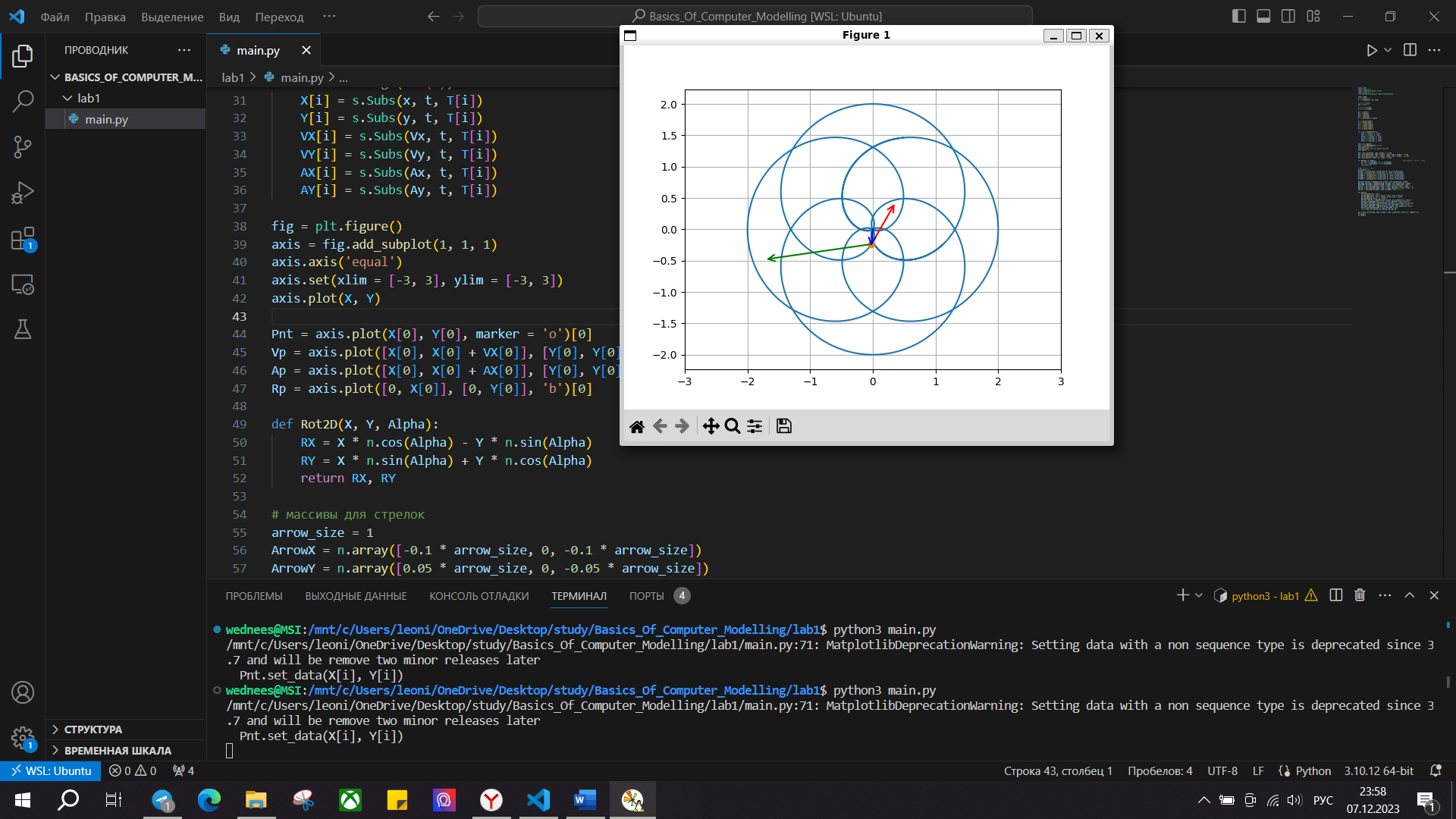
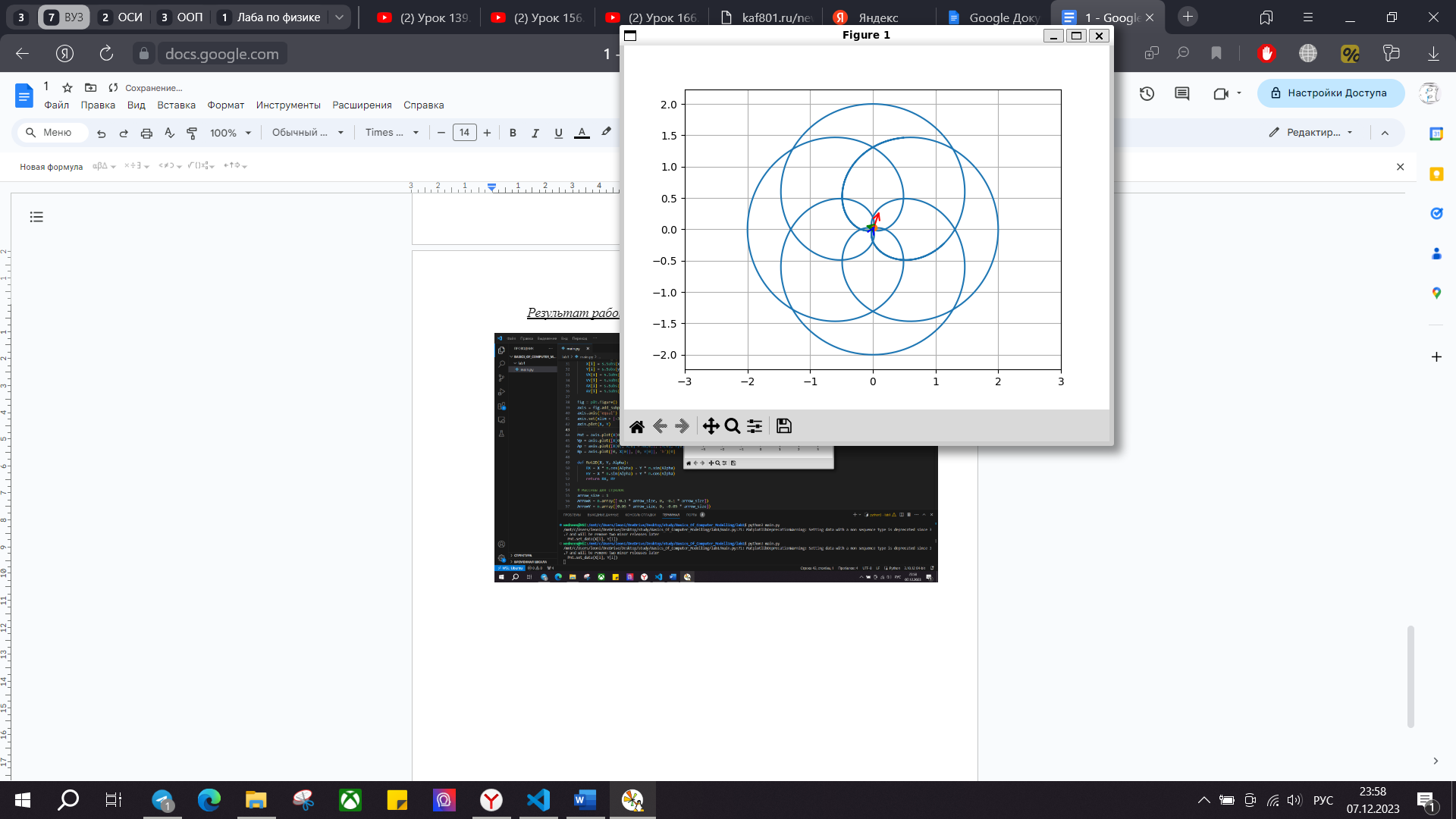
return Pnt, Vp, VArrow, Ap, AArrow, Rp, RArrow

an = FuncAnimation(fig, anim, frames = step, interval=1, blit=True, repeat=True)

plt.grid()

plt.show()

*Результат работы программы:*



*Вывод:* в результате проделанной работы, я научился анимировать выводить траекторию движения точки и векторы, описывающие ее движение на языке Python.