ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

**ОТЧЕТ**

**О ВЫПЛОНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

**«ДИНАМИКА СИСТЕМЫ»**

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

**ВАРИАНТ ЗАДАНИЯ № 5**

Выполнил студент группы M8O-206Б-22

Волков А.Д.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Проверил и принял

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

подпись, дата

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

Москва, 2023

*Задание:*

*Изобразить необходимую систему, и провести по ней точку, с указанием векторов скорости, ускорения и радиуса кривизны.*

*Вариант* 5:



*Программа:*

import math

import sympy as s

import matplotlib.pyplot as plt

import numpy as n

from matplotlib.animation import FuncAnimation

t = s.Symbol('t')

x = (2 + s.sin(8 \* t)) \* s.cos(t + 0.5 \* s.sin(4 \* t))

y = (2 + s.sin(8 \* t)) \* s.sin(t + 0.5 \* s.sin(4 \* t))

Vx = s.diff(x)

Vy = s.diff(y)

v = (Vx \*\* 2 + Vy \*\* 2) \*\* 0.5

Ax = s.diff(Vx)

Ay = s.diff(Vy)

a = (Ax \*\* 2 + Ay \*\* 2) \*\* 0.5

Atan = s.diff(v)

Acs = (a \*\* 2 - Atan \*\* 2) \*\* 0.5

ATanx = Vx / v \* Atan

ATany = Vy / v \* Atan

ANorx = Ax - ATanx

ANory = Ay - ATany

ANor = (ANorx \*\* 2 + ANory \*\* 2) \*\* 0.5

CRX = (ANorx / ANor) \* v \*\* 2 / Acs

CRY = (ANory / ANor) \* v \*\* 2 / Acs

step = 2500

T = n.linspace(0, 10, step)

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)

CRx = n.zeros\_like(T)

CRy = 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])

CRx[i] = s.Subs(CRX, t, T[i])

CRy[i] = s.Subs(CRY, t, T[i])

fig = plt.figure()

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

axis.axis('equal')

axis.set(xlim = [-5, 5], ylim = [-5, 5])

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]

Cr = axis.plot([X[0], X[0] + CRx[0]], [Y[0], Y[0] + CRy[0]], 'black')[0]

def Vect\_arrow(X, Y, ValX, ValY):

a = 0.2

b = 0.3

Arx = n.array([-b, 0, -b])

Ary = n.array([a, 0, -a])

alp = math.atan2(ValY, ValX)

RotArx = Arx \* n.cos(alp) - Ary \* n.sin(alp)

RotAry = Arx \* n.sin(alp) + Ary \* n.cos(alp)

Arx = X + ValX + RotArx

Ary = Y + ValY + RotAry

return Arx, Ary

AVx, AVy = Vect\_arrow(X[0], Y[0], VX[0], VY[0])

AAx, AAy = Vect\_arrow(X[0], Y[0], AX[0], AY[0])

ACRx, ACRy = Vect\_arrow(X[0], Y[0], CRx[0], CRy[0])

Varrow = axis.plot(AVx, AVy, 'red')[0]

Aarrow = axis.plot(AAx, AAy, 'green')[0]

CRarrow = axis.plot(ACRx, ACRy, 'black')[0]

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] + AY[i]])

Cr.set\_data([X[i], X[i] + CRx[i]], [Y[i], Y[i] + CRy[i]])

AVx, AVy = Vect\_arrow(X[i], Y[i], VX[i], VY[i])

AAx, AAy = Vect\_arrow(X[i], Y[i], AX[i], AY[i])

ACRx, ACRy = Vect\_arrow(X[i], Y[i], CRx[i], CRy[i])

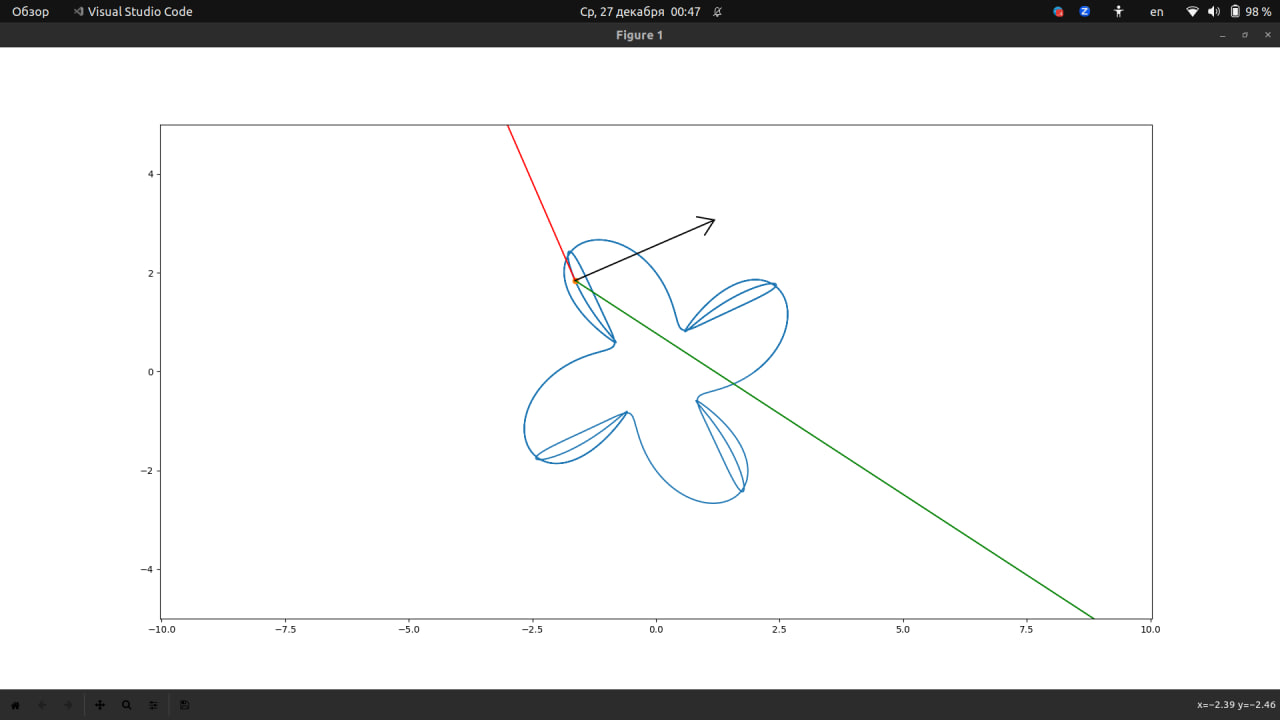
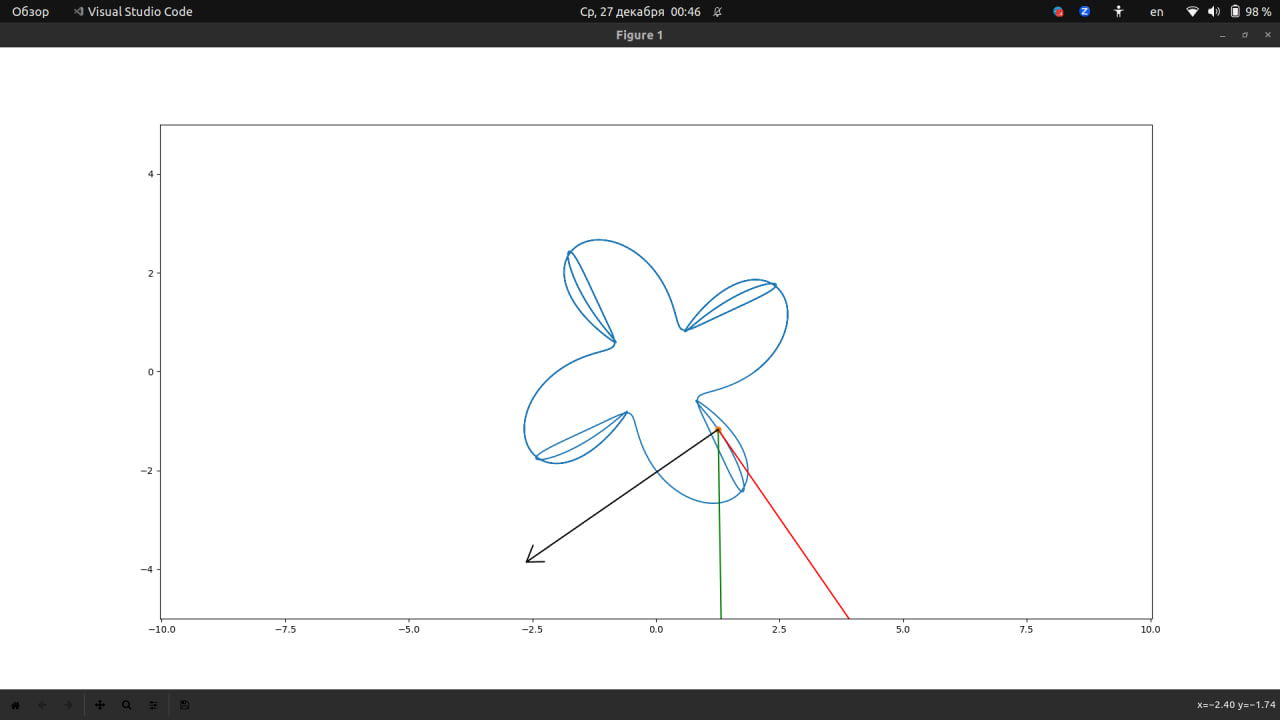
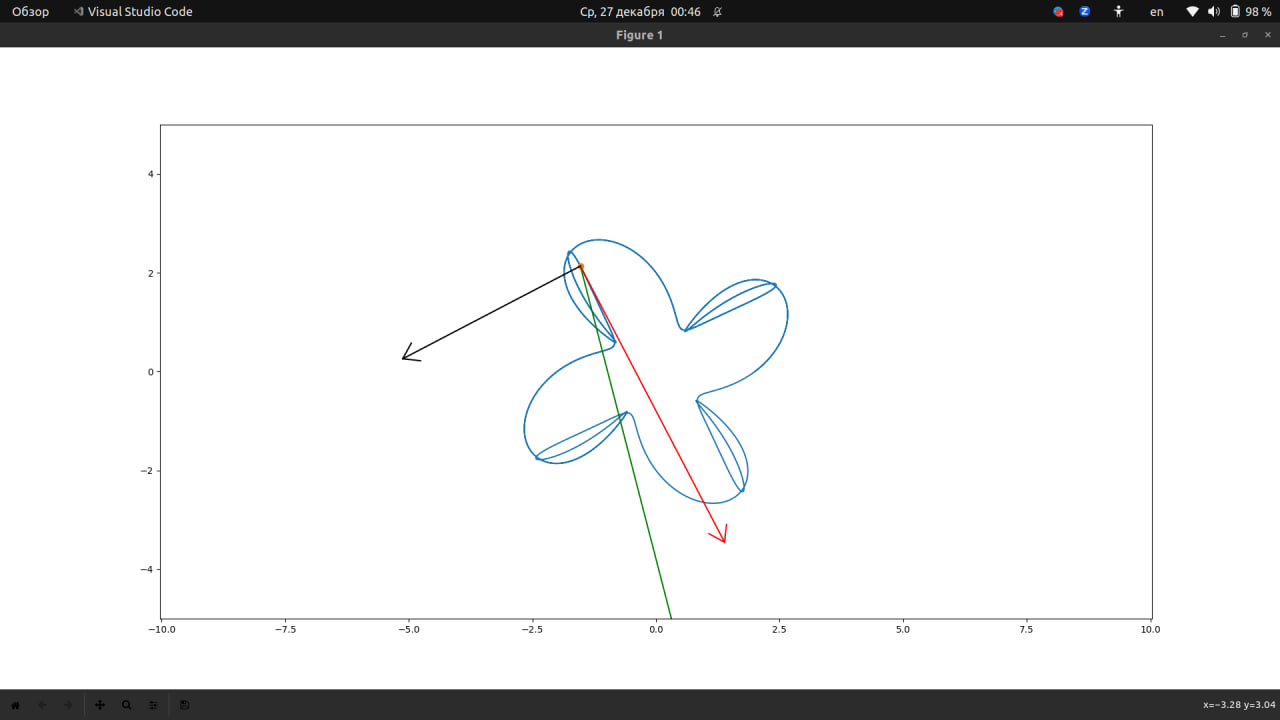
Varrow.set\_data(AVx, AVy)

Aarrow.set\_data(AAx, AAy)

CRarrow.set\_data(ACRx, ACRy)

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

plt.show()



*Выводы:*

Я успешно выполнил данную лабораторную работу по теоретической механике. В ходе ее выполнения я изучил основные принципы работы с библиотеками numpy, scipy и matplotlib. Также я изучил некоторые новые формулы и применил их в саоей программе.