Московский авиационный институт

(национальный исследовательский университет)

Институт № 8 «Информационные технологии и прикладная математика»

**Лабораторная работа №1**

**по курсу «Теоретическая механика»**

**Анимация точки**

Выполнил студент группы М8О-204Б-21

Рябов Серафим Михайлович

Преподаватель: Беличенко Михаил Валериевич

Оценка:

Дата:

Москва, 2022

**Вариант №12**

**Задание:**

Построить заданную траекторию и анимацию движения точки, а также отобразить стрелки скорости и ускорения. Построить радиус кривизны траектории.

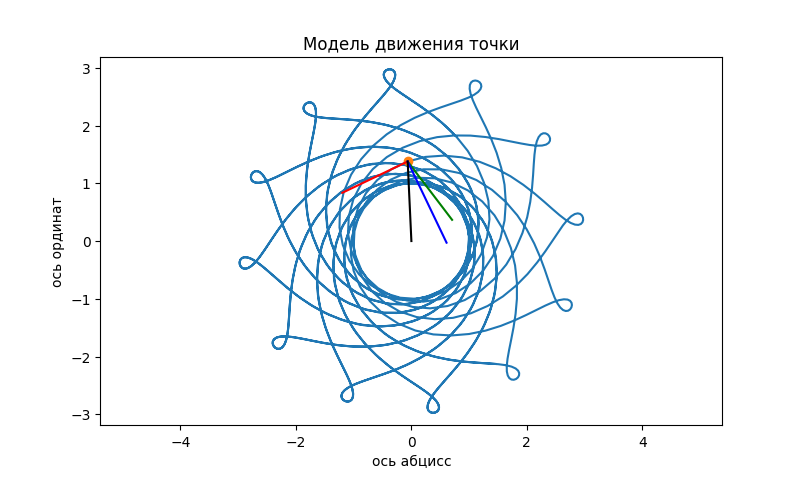
Закон движения точки:

Текст программы

import numpy as np  
import matplotlib.pyplot as plt  
from matplotlib.animation import FuncAnimation  
import math  
  
t = np.linspace(0, 20, 1001)  
  
r = 2 + np.sin(6 \* t)  
phi = 6.5 \* t + 1.2 \* np.cos(6 \* t)  
  
x = r \* np.cos(phi)  
y = r \* np.sin(phi)  
  
xSpd = np.diff(x) # speed  
ySpd = np.diff(y)  
spd = np.sqrt(xSpd \* xSpd + ySpd \* ySpd)  
  
xAcl = np.diff(xSpd) # acceleration  
yAcl = np.diff(ySpd)  
acl = np.sqrt(xAcl \* xAcl + yAcl \* yAcl)  
tanAcl = np.diff(spd)  
  
tanAcl\_calc = np.zeros\_like(spd)  
tanAcl\_calc[:len(tanAcl)] = tanAcl  
acl\_calc = np.zeros\_like(spd)  
acl\_calc[:len(acl)] = acl  
curvRadius = spd \* spd / np.sqrt(acl\_calc \* acl\_calc - tanAcl\_calc \* tanAcl\_calc) # radius of curvature  
  
fig = plt.figure(figsize=[10, 10])  
ax = fig.add\_subplot(1, 1, 1)  
ax.axis('equal')  
ax.set\_title("Модель движения точки")  
ax.set\_xlabel('ось абцисс')  
ax.set\_ylabel('ось ординат')  
  
  
ax.plot(x, y)  
  
P = ax.plot(x[0], y[0], marker='o')[0]  
  
aclLine = ax.plot([x[0], x[0] + xAcl[0]], [y[0], y[0] + yAcl[0]], 'g', label = 'Вектор ускорения')[0]  
spdLine = ax.plot([x[0], x[0] + xSpd[0]], [y[0], x[0] + xSpd[0]], 'r', label = 'Вектор скорости')[0]  
curvRadiusLine = ax.plot([x[0], x[0] - ySpd[0] \* curvRadius[0] / math.sqrt(xSpd[0]\*\*2 + ySpd[0]\*\*2)],  
 [y[0], y[0] + xSpd[0] \* curvRadius[0] / math.sqrt(xSpd[0]\*\*2 + ySpd[0]\*\*2)], 'b', label = 'Вектор кривизны')[0]  
radiusLine, = ax.plot([0, x[0]], [0, y[0]], 'black', label = 'Радиус-вектор')  
  
R = math.sqrt(math.pow(x[0], 2) + math.pow(y[0], 2))  
  
def Movement(i):  
 P.set\_data(x[i], y[i])  
 curvRadX = x[i] - ySpd[i] \* curvRadius[i] / math.sqrt(xSpd[i]\*\*2 + ySpd[i]\*\*2)  
 curvRadY = y[i] + xSpd[i] \* curvRadius[i] / math.sqrt(xSpd[i]\*\*2 + ySpd[i]\*\*2)  
  
 spdLine.set\_data([x[i], x[i] + xSpd[i] \* 4], [y[i], y[i] + ySpd[i] \* 4])  
 curvRadiusLine.set\_data([x[i], curvRadX], [y[i], curvRadY])  
 aclLine.set\_data([x[i], x[i] + (xAcl[i] \* 20)], [y[i], y[i] + (yAcl[i] \* 20)])

radiusLine.set\_data([0, x[i]], [0, y[i]])  
  
 return [P]  
  
anim = FuncAnimation(fig, Movement, frames=len(t), interval=10)  
  
plt.show()

Результат работы программы:



где красный отрезок – вектор скорости

черный отрезок – радиус-вектор

синий отрезок – вектор кривизны

зеленый отрезок – вектор ускорения