**Московский авиационный институт**

**(Национальный исследовательский университет)**

Институт: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Теоретическая механика»

**Лабораторная работа № 3**

**по курсу** **«Теоретическая механика»**

**Составление и численное решения дифференциальных уравнений движения системы и ее анимация.**

Студент: Попов И. П.

Группа: М80-206Б-20

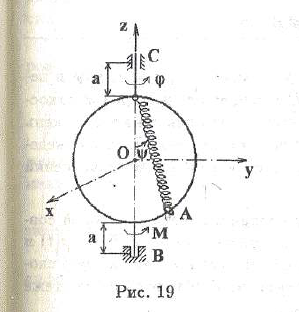
Преподаватель: Сухов Е.А.

Дата: 21.12

Оценка:

Москва, 2021

**Вариант №19**

****

**Задание:**

Необходимо составить и численно решить дифференциальные уравнения движения системы (уравнения Лагранжа второго рода), а затем реализовать анимацию движения механической системы используя язык программирования Python.

**Листинг программы:**

import numpy as np

import matplotlib.pyplot as plt

from matplotlib.animation import FuncAnimation

from scipy.integrate import odeint

import sympy as sp

import math

def formY(y, t, fV, fOm):

y1,y2,y3,y4 = y

dydt = [y3,y4,fV(y1,y2,y3,y4),fOm(y1,y2,y3,y4)]

return dydt

# defining parameters

alpha = math.pi / 6

M = 1

m = 0.1

R = 0.3

c = 20

l0 = 0.2

g = 9.81

# defining t as a symbol

t = sp.Symbol('t')

# defining functions of 't':

phi=sp.Function('phi')(t)

psi=sp.Function('psi')(t)

Vphi=sp.Function('Vphi')(t)

Vpsi=sp.Function('Vpsi')(t)

l = 2 \* R \* sp.cos(phi)

#constructing the Lagrange equations

#1 defining the kinetic energy

TT1 = M \* R\*\*2 \* Vphi\*\*2 / 4

V1 = 2\*Vpsi \* R

V2 = Vphi \* R \* sp.sin(2 \* psi)

Vr2 = V1\*\*2 + V2\*\*2

TT2 = m \* Vr2 / 2

TT = TT1+TT2

# 2 defining the potential energy

Pi1 = 2 \* R \* m \* g \* sp.sin(psi)\*\*2

Pi2 = (c \* (l - l0)\*\*2) / 2

Pi = Pi1+Pi2

# 3 Not potential force

M = alpha \* phi\*\*2;

# Lagrange function

L = TT-Pi

# equations

ur1 = sp.diff(sp.diff(L,Vphi),t)-sp.diff(L,phi) - M

ur2 = sp.diff(sp.diff(L,Vpsi),t)-sp.diff(L,psi)

# isolating second derivatives(dV/dt and dom/dt) using Kramer's method

a11 = ur1.coeff(sp.diff(Vphi,t),1)

a12 = ur1.coeff(sp.diff(Vpsi,t),1)

a21 = ur2.coeff(sp.diff(Vphi,t),1)

a22 = ur2.coeff(sp.diff(Vpsi,t),1)

b1 = -(ur1.coeff(sp.diff(Vphi,t),0)).coeff(sp.diff(Vpsi,t),0).subs([(sp.diff(phi,t),Vphi), (sp.diff(psi,t), Vpsi)])

b2 = -(ur2.coeff(sp.diff(Vphi,t),0)).coeff(sp.diff(Vpsi,t),0).subs([(sp.diff(phi,t),Vphi), (sp.diff(psi,t), Vpsi)])

detA = a11\*a22-a12\*a21

detA1 = b1\*a22-b2\*a21

detA2 = a11\*b2-b1\*a21

dVdt = detA1/detA

domdt = detA2/detA

countOfFrames = 2500

# Constructing the system of differential equations

T = np.linspace(0, 25, countOfFrames)

fVphi = sp.lambdify([phi,psi,Vphi,Vpsi], dVdt, "numpy")

fVpsi = sp.lambdify([phi,psi,Vphi,Vpsi], domdt, "numpy")

y0 = [0, np.pi/6, -0.5, 0]

sol = odeint(formY, y0, T, args = (fVphi, fVpsi))

# Plotting and graph with axis alignment

fig = plt.figure(figsize=(17, 8))

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

ax1.axis('equal')

phi = sol[:,0]

psi = sol[:,1]

Vphi = sol[:,2]

Vpsi = sol[:,3]

w = np.linspace(0, 2 \* math.pi, countOfFrames)

conline, = ax1.plot([sp.sin(2\*psi[0]) \* R \* sp.cos(phi[0]), 0], [-sp.cos(2\*psi[0]) \* R, R], 'black')

P, = ax1.plot(sp.sin(2\*psi[0]) \* R \* sp.cos(phi[0]), -sp.cos(2\*psi[0]) \* R, marker='o', color='black')

Circ, = ax1.plot(R \* sp.cos(phi[0]) \* np.cos(w), R \* np.sin(w), 'black')

#Additional subgraph

ax2 = fig.add\_subplot(4, 2, 2)

ax2.plot(T, Vphi)

ax2.set\_xlabel('T')

ax2.set\_ylabel('Vphi')

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

ax3.plot(T, Vpsi)

ax3.set\_xlabel('T')

ax3.set\_ylabel('Vpsi')

def anima(i):

P.set\_data(sp.sin(2\*psi[i]) \* R \* sp.cos(phi[i]), -sp.cos(2\*psi[i]) \* R)

conline.set\_data([sp.sin(2\*psi[i]) \* R \* sp.cos(phi[i]), 0], [-sp.cos(2\*psi[i]) \* R, R])

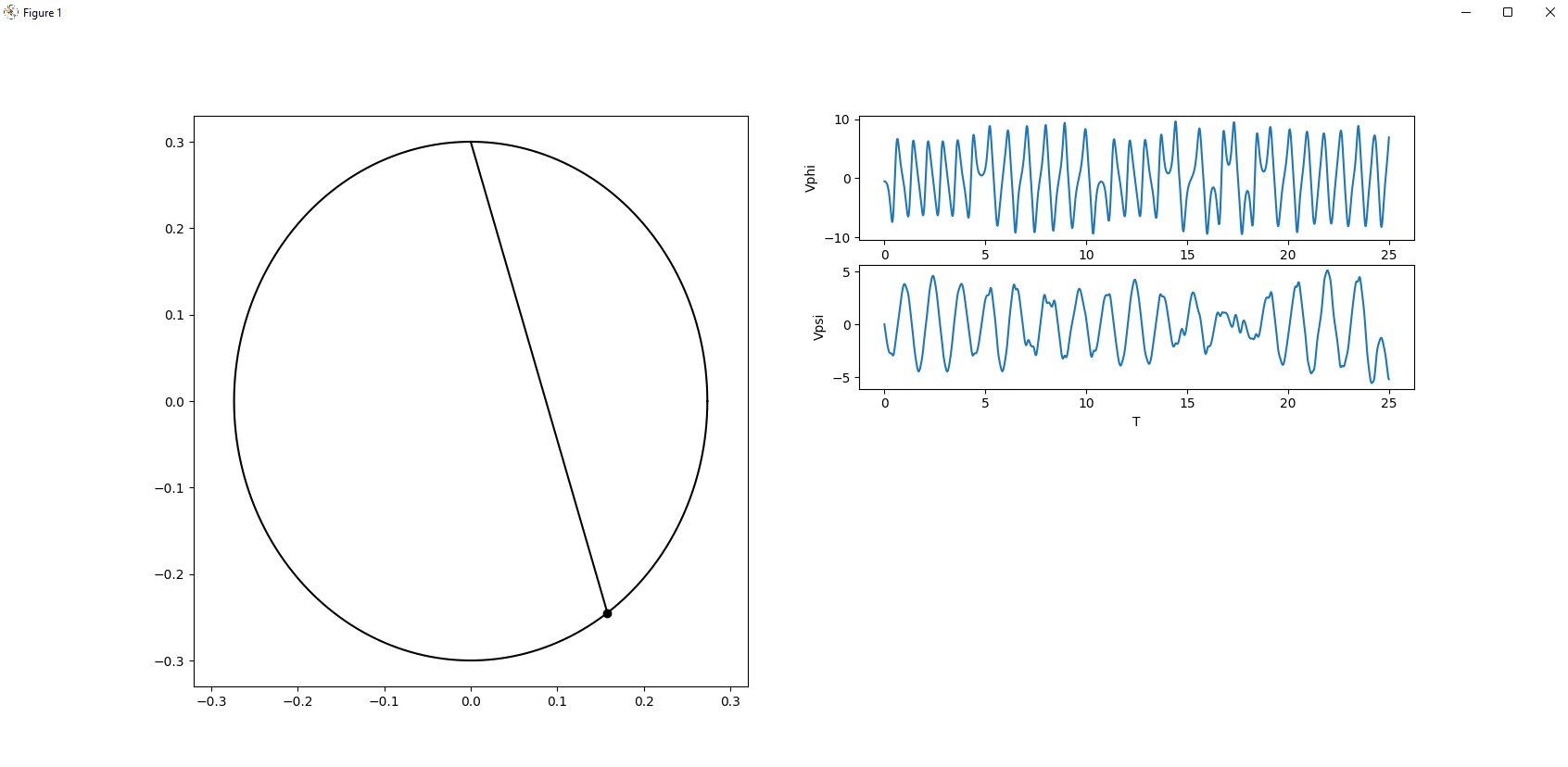
Circ.set\_data(R \* sp.cos(phi[i]) \* np.cos(w), R \* np.sin(w))

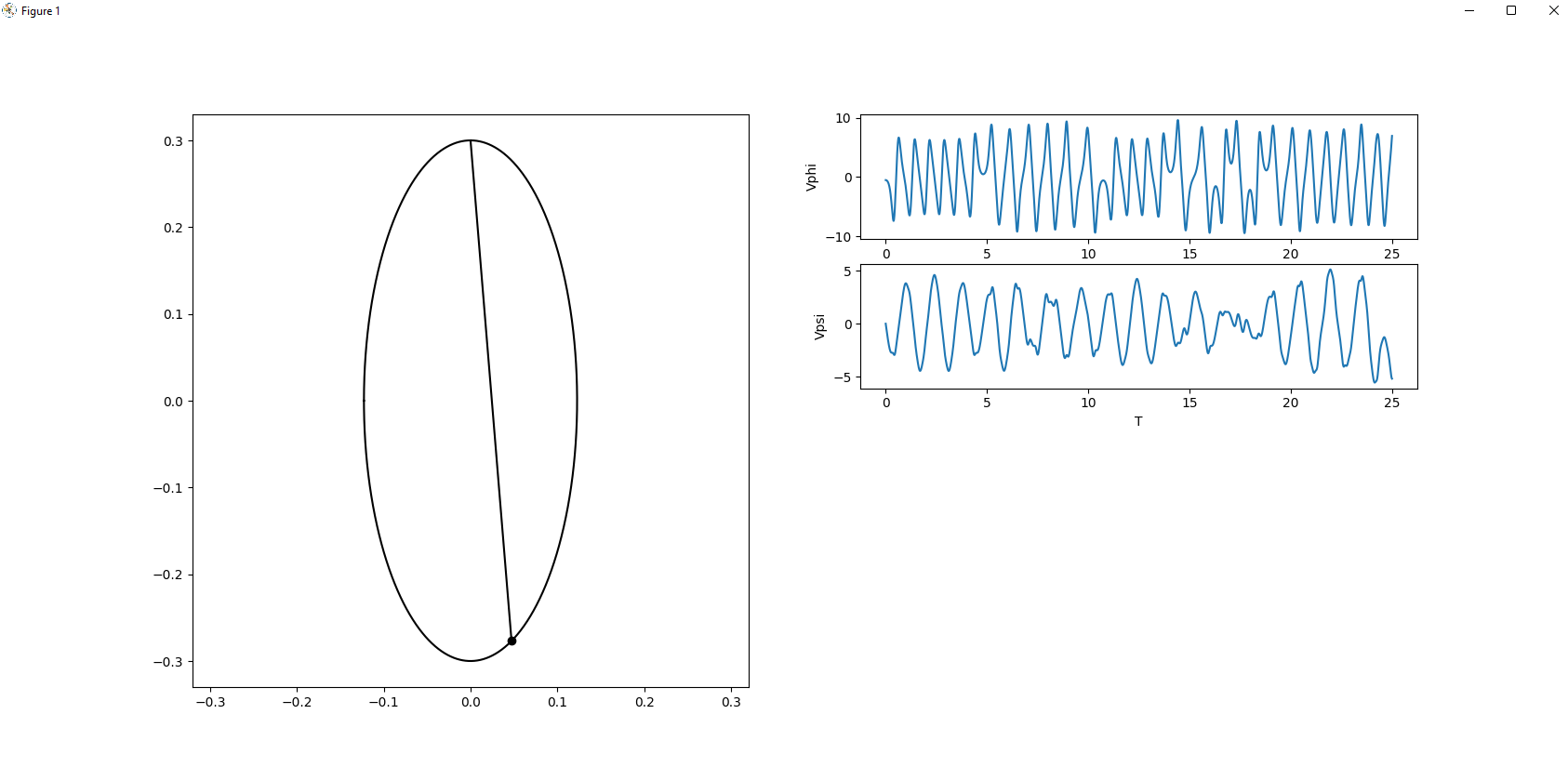
return Circ, P, conline

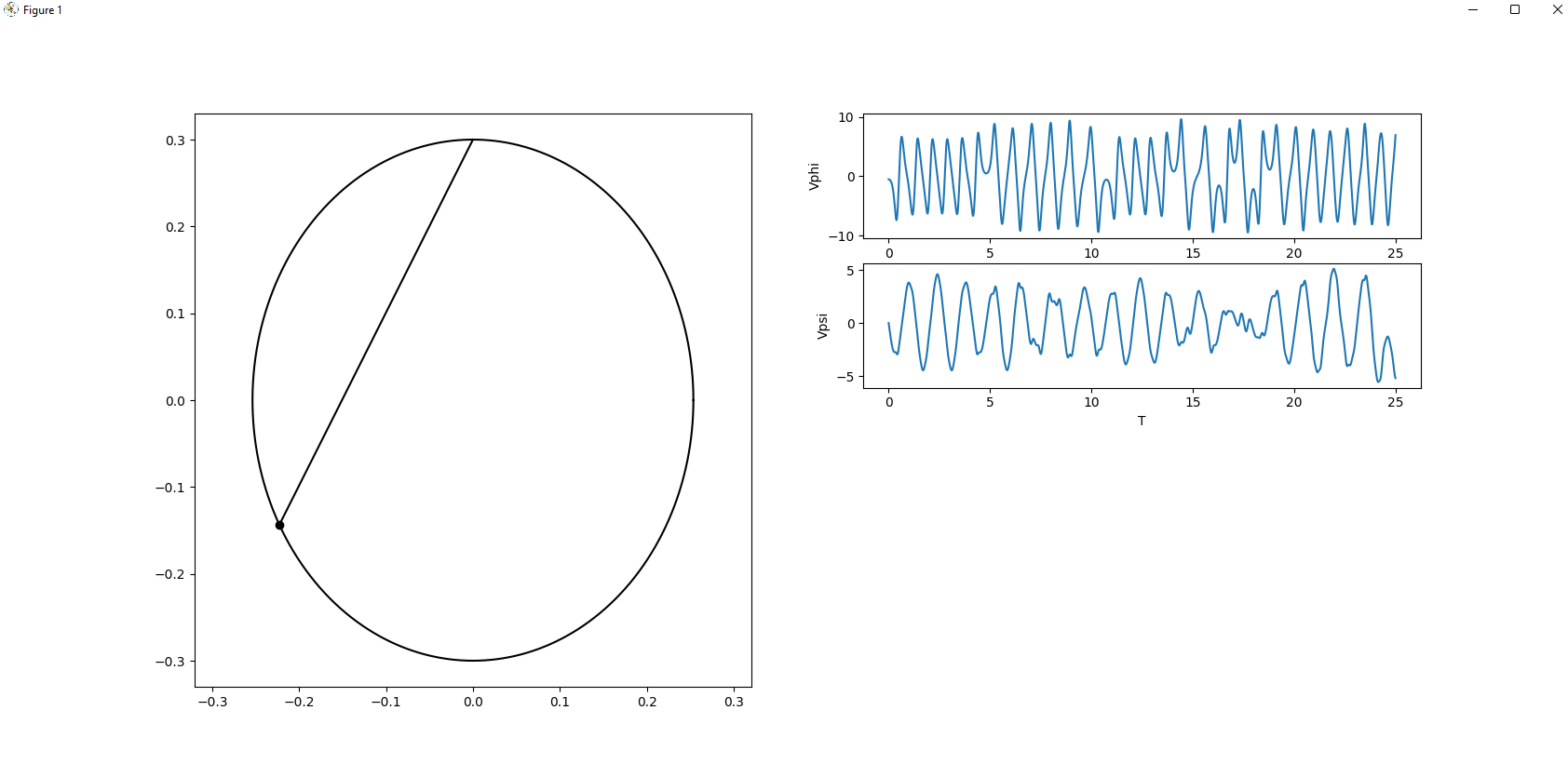
anim = FuncAnimation(fig, anima, frames=countOfFrames, interval=1, blit=True)

plt.show()

**Результат работы**







**Теоретическая часть**

Уравнения Лагранжа второго рода, которые представляют собой дифференциальные уравнения второго порядка относительно обобщенных координат.

Рассмотрим [механическую систему](https://isopromat.ru/teormeh/kratkaja-teoria/massa-mehaniceskoj-sistemy), имеющую s [степеней свободы](https://isopromat.ru/tmm/kratkij-kurs/chislo-stepenej-svobody-cepi), на которую наложены [стационарные, идеальные, голономные связи](https://isopromat.ru/teormeh/kratkaja-teoria/svyazi).

В этом случае положение системы определяется s [обобщенными координатами](https://isopromat.ru/teormeh/kratkaja-teoria/obobshennye-koordinaty) ***q1, q2,…qs.***

[Кинетическая энергия такой системы](https://isopromat.ru/teormeh/kratkaja-teoria/kineticheskaa-energia) является функцией обобщенных координат ***q1, q2,…qs***, обобщенных скоростей



и времени



Для такой системы можно записать s уравнений, которые называются уравнениями Лагранжа второго рода или [дифференциальными уравнениями движения](https://isopromat.ru/teormeh/kratkaja-teoria/differencialnye-uravnenia-dvizhenia-tochki) в обобщенных координатах:

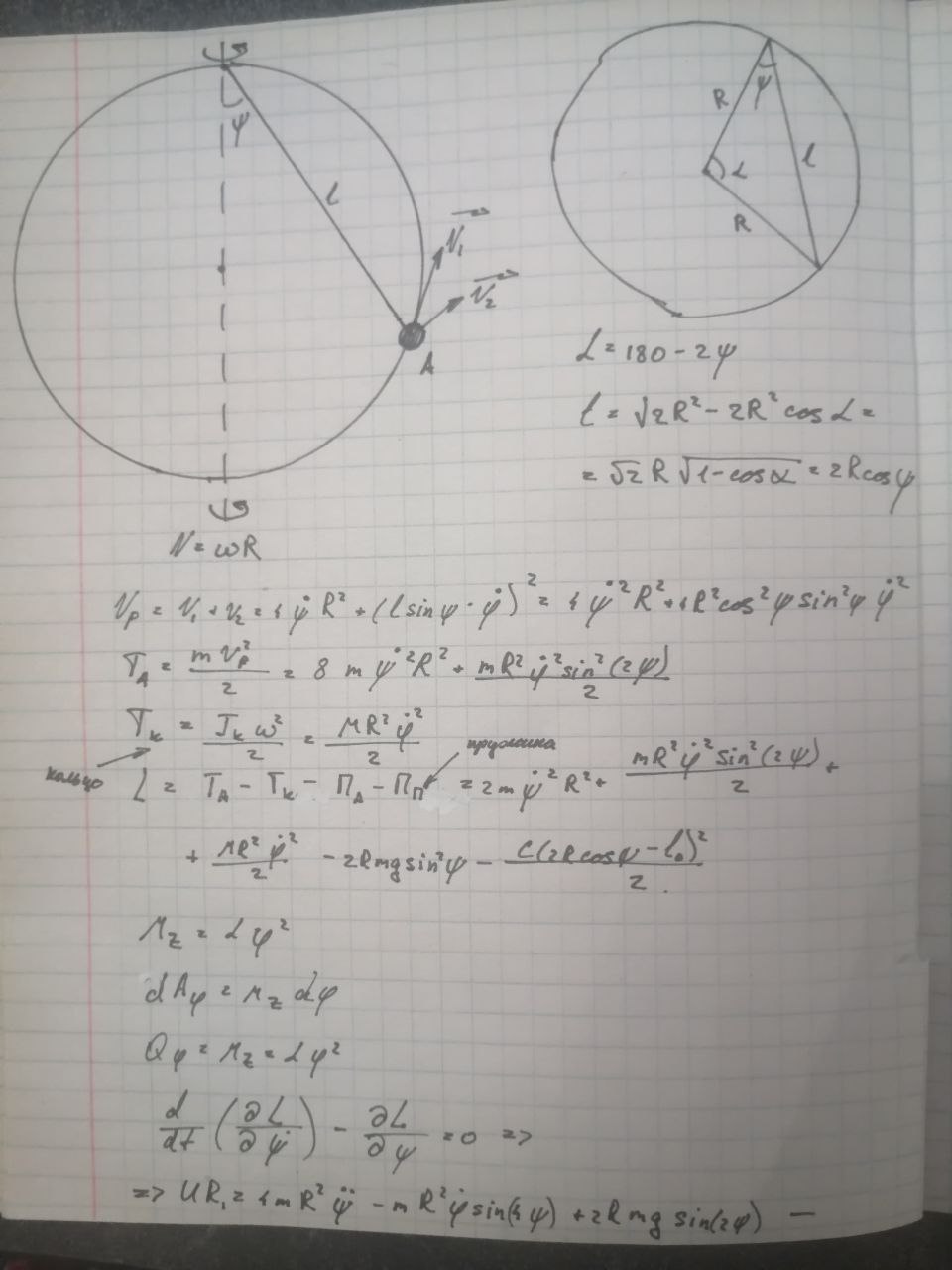


где ***Qj*** – обобщенная сила.

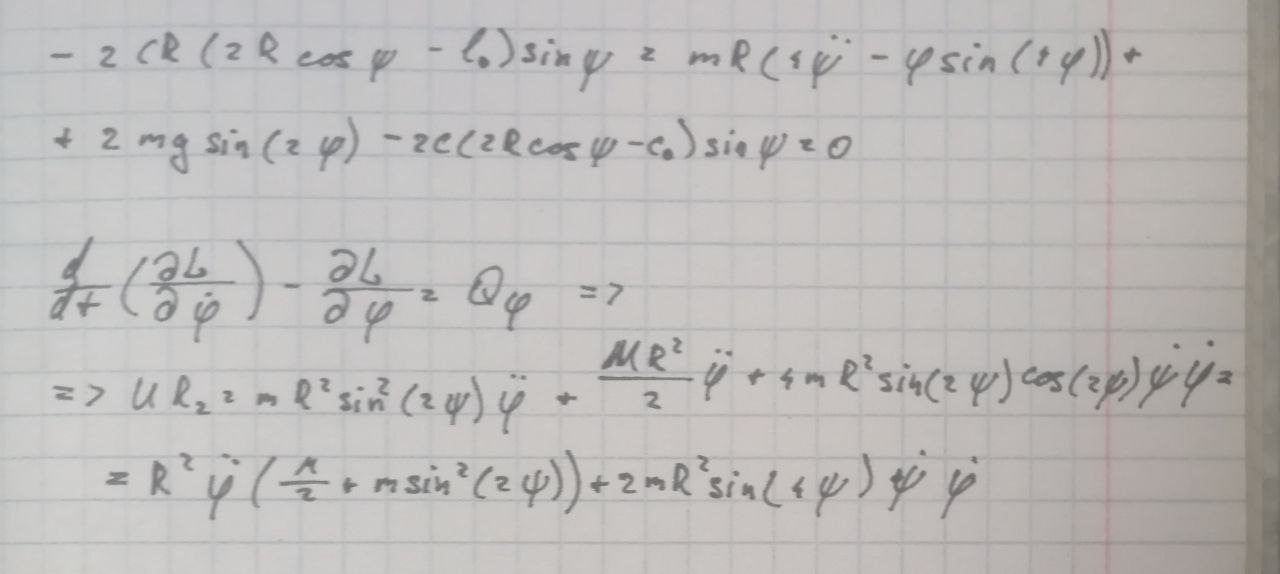
Уравнения Лагранжа второго рода представляют собой [дифференциальные уравнения второго порядка](https://online-matematika.ru/%D0%B4%D0%B8%D1%84%D1%84%D0%B5%D1%80%D0%B5%D0%BD%D1%86%D0%B8%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B5-%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D1%8F/%D0%B4%D0%B8%D1%84-%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D1%8F-%D0%B2%D1%82%D0%BE%D1%80%D0%BE%D0%B3%D0%BE-%D0%BF%D0%BE%D1%80%D1%8F%D0%B4%D0%BA%D0%B0) относительно обобщенных координат ***q1, q2,…qs***.

Дважды интегрируя эти уравнения и определяя по начальным условиям постоянные интегрирования, получим систему уравнений движения в обобщенных координатах:

***qi = qi(t),  j ÷ s***







**Вывод**

В процессе выполнения данной лабораторной работы я познакомился с уравнениями Лагранжа 2 рода и использовал полученные знания для написания программы анимации системы из своего варианта задания. Данные уравнения - наиболее удобный и совершенный способ составления уравнений движения механических систем. Число таких уравнений минимально и равно числу степеней свободы механической системы, что позволяет легко представлять их компьютеру.