Московский авиационный институт

(национальный исследовательский университет)

Институт № 8 «Информационные технологии и прикладная математика»

**Лабораторная работа №2**

**по курсу «Теоретическая механика»**

**Анимация системы**

Выполнил студент группы М8О-208Б-20

Морозов Артем Борисович

Преподаватель: Волков Евгений Валерьевич

Оценка:

Дата: 13.12.2021

Подпись:

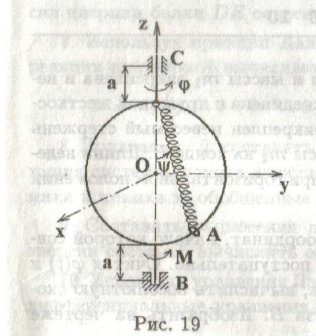
Москва, 2021

**Вариант №«19»**

**Задание:**

Реализовать анимацию движения механической системы используя язык программирования Python.

**Механическая система:**



**Текст программы:**

import numpy as np

import matplotlib.pyplot as plt

from matplotlib.animation import FuncAnimation

Steps = 1000

t = np.linspace(0, 10, Steps)

ph = np.sin(4.7 \* t)

fig = plt.figure()

ax = fig.add\_subplot(projection='3d')

ax.set(xlim=[-8, 8], ylim=[-8, 8], zlim=[0, 8])

theta = np.linspace(0, 20 \* np.pi, 1001)

y = 3.5 \* np.cos(theta)

z = 3.5 \* np.sin(theta)

Z\_PointCentral = 8.5

Y\_PointCentral = 0

X\_PointCentral = 0

PointCentral = ax.plot(X\_PointCentral, Y\_PointCentral, Z\_PointCentral, color='blue', marker='o', markeredgewidth=1)[0]

Z\_PointM = 5

X\_PointM = 0

Y\_PointM = 0

Circle\_cir = ax.plot(y \* np.cos(np.pi), y \* np.sin(np.pi), z + 5, linewidth=5, color='red', alpha=.2)[0]

PointM = ax.plot(X\_PointM, Y\_PointM, Z\_PointM, color='black', marker='o', markeredgewidth=4)[0]

def get\_spring\_line(coils, diameter, start, end):

x = np.linspace(start[0], end[0], coils \* 2)

y = np.linspace(start[1], end[1], coils \* 2)

z = np.linspace(start[2], end[2], coils \* 2)

for i in range(1, len(z) - 1):

z[i] = z[i] + diameter \* 1 \* (-1) \*\* i

return np.array([x, y, z], dtype=object)

ax.plot([0, 0], [0, 0], [0, 10.80], linewidth=2, color='black', alpha=.8) #stick

ax.plot([-2.25, -1.25], [-0.25, 0.25], [0.3, 0.3], linewidth=2, color='black', alpha=.8)

ax.plot([-1.25, -1.25], [0.23, 0.25], [-0.3, 0.3], linewidth=2, color='black', alpha=.8)

ax.plot([-1.25, 0.5], [0.25, 1.25], [-0.4, -0.4], linewidth=2, color='black', alpha=.8)

ax.plot([0.5, 0.5], [1.23, 1.25], [-0.3, 0.3], linewidth=2, color='black', alpha=.8)

ax.plot([0.5, 1.40], [1.25, 1.80], [0.3, 0.3], linewidth=2, color='black', alpha=.8)

ax.plot([-0.25, -0.25], [0, 0], [9.75, 10.5], linewidth=1.5, color='black', alpha=.8)

ax.plot([0.25, 0.25], [0.15, 0.15], [9.77, 10.52], linewidth=1.5, color='black', alpha=.8) #right

ax.plot([0.25, 0.5], [0.15, 0.3], [10.52, 10.35], linewidth=1.5, color='black', alpha=.8)

ax.plot([0.25, 0.5], [0.15, 0.3], [10.35, 10.17], linewidth=1.5, color='black', alpha=.8)

ax.plot([0.25, 0.5], [0.15, 0.3], [10.17, 10], linewidth=1.5, color='black', alpha=.8)

ax.plot([0.25, 0.5], [0.15, 0.3], [10, 9.83], linewidth=1.5, color='black', alpha=.8)

ax.plot([-0.25, -0.5], [0, -0.15], [10.5, 10.33], linewidth=1.5, color='black', alpha=.8) #left

ax.plot([-0.25, -0.5], [0, -0.15], [10.33, 10.16], linewidth=1.5, color='black', alpha=.8)

ax.plot([-0.25, -0.5], [0, -0.15], [10.16, 9.99], linewidth=1.5, color='black', alpha=.8)

ax.plot([-0.25, -0.5], [0, -0.15], [9.99, 9.82], linewidth=1.5, color='black', alpha=.8)

# Пружина

spring\_xyz = get\_spring\_line(30, 0.1, [0, 0, 8.5], [1, 3, 2])

spring = ax.plot(spring\_xyz[0], spring\_xyz[1], spring\_xyz[2], linewidth=2, color='black')[0]

def Kino(i):

Circle\_cir.set\_data\_3d(y \* np.cos(theta[i]), y \* np.sin(theta[i]), z + 5)

curz = (Z\_PointM + 3.5 \* np.cos(theta[i]))

if curz >= 5:

delta = curz - 5

curz = curz - 2 \* delta

PointM.set\_data\_3d((X\_PointM + 3.5 \* np.sin(theta[i])) \* np.cos(theta[i]),

(Y\_PointM + 3.5 \* np.sin(theta[i])) \* np.sin(theta[i]), curz)

newx = (X\_PointM + 3.5 \* np.sin(theta[i])) \* np.cos(theta[i])

newy = (Y\_PointM + 3.5 \* np.sin(theta[i])) \* np.sin(theta[i])

newz = curz

spring\_xyz = get\_spring\_line(30, 0.2, [0, 0, 8.5], [newx, newy, newz])

spring.set\_data\_3d(spring\_xyz[0], spring\_xyz[1], spring\_xyz[2])

return [Circle\_cir, PointM, spring]

anima = FuncAnimation(fig, Kino, frames=500, interval=80)

plt.show()

**Результат работы:**

|  |
| --- |
|  |
|  |