**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информатика»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2

по дисциплине **«Математическое моделирование сложных систем»**

на тему: **«Построение статических моделей»**

Выполнил: студент гр. ИП-31

Кузнецова Е. А.

Принял: доцент

Трохова Т. А.

Дата сдачи отчета: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата допуска к защите: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Гомель 2024

**Цель работы:** получить навыки компьютерного моделирования технических объектов, представленных в виде статической модели с выводом результатов моделирования в численном и графическом виде.

***Задача 1. Компьютерная модель кривошипно-ползунного механизма***

## *Постановка задачи моделирования*

***1) Разработать компьютерную модель манипулятора в виде кривошипно-ползунного механизма, которая имеет следующие выходные параметры:***

- длины звеньев кривошипно-ползунного механизма по заданным исходным данным;

- проверить условие существования механизма.

- рассчитать функции координат характерных точек механизма в зависимости от угла поворота кривошипа. Построить графики этих функций.

- функция хода ползуна в зависимости от угла поворота кривошипа.

Результаты моделирования представить в численном и графическом виде.

***2) Исследовать модель, для чего:***

- вычислить значение угла поворота кривошипа, при котором функция хода ползуна пересекает пороговое значение (пороговое значение подобрать самостоятельно).

***3) Дополнительно.***

Построить графически вид механизма в нескольких положениях или разработать анимацию движения механизма.

Таблица 2.1 - Таблица исходных данных

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N варианта | (град) | (град) | (град) | *S1*  *(м)* | *S2*  *(м)* | *S3*  *(м)* | *a4*  *(м)* | *Β*  (град) |
| 2 | 50 | 40 | 30 | 0.57 | 0.63 | 0.68 | 0.05 | 110 |

**Код программы:**

import numpy as np

import matplotlib.pyplot as plt

import math as mt

f1 = mt.radians(50)

f2 = mt.radians(40)

f3 = mt.radians(30)

S1 = 0.57

S2 = 0.63

S3 = 0.68

a4 = 0.05

B = mt.radians(110)

#функция K

def K\_calculate():

#матрица коэффициентов

A = np.array([[S1\* np.cos(f1), np.sin(f1), -1], [S2\* np.cos(f2), np.sin(f2), -1], [S3\* np.cos(f3), np.sin(f3), -1]])

#вектор свободных членов

B = np.array([[S1\*\*2], [S2\*\*2], [S3\*\*2]])

#вектор решения

X = np.linalg.solve(A, B)

return X

K\_values = K\_calculate()

print("K:", K\_values[0], K\_values[1], K\_values[2])

#длины звеньев

a1 = K\_values[0]/2

a3 = K\_values[1]/(2\*a1)

a2 = np.sqrt(a1\*\*2 + a3\*\*2 -K\_values[2])

print("Длины звеньев:", a1, a2, a3)

#проверить условие существования

if a1 < a2 - a3:

if a3 > 0:

H = 1

else: H = -1

#координаты

xc = a1 \* np.cos(f1) + np.sqrt(a2\*\*2 - (a3\*H - a1\* np.sin(f1))\*\*2)

yc = a3\*H

xb = a1 \* np.cos(f1)

yb = a1 \* np.sin(f1)

f6 = f2 + B

xn = a1 \* np.cos(f1) + a4 \* np.cos(f6)

yn = a1 \* np.sin(f1) + a4 \* np.sin(f6)

print("Координаты: ", xc, yc, xb, yb, xn, yn)

f2 = np.arccos((xc - a1 \* np.cos(f1)/a2))

#ход ползуна

S = a1 \* np.cos(f1) + np.sqrt(a2\*\*2 - (a3 \* H - a1\*np.sin(f1))\*\*2)

print("Функция: ", S)

s\_values = np.linspace(0, 6, 100)

s\_result = []

dot = 0.7

crossings = []

for element in s\_values:

y = a1 \* np.cos(element) + np.sqrt(a2\*\*2 - (a3 \* H - a1 \* np.sin(element))\*\*2)

s\_result.append(y)

if abs(y - dot) < 0.015:

crossings.append(element)

plt.plot(np.array([xc, xb]), np.array([yc, yb]), color='black')

plt.plot(np.array([xn, xb]), np.array([yn, yb]), color='black')

plt.plot([0, xb[0]], [0, yb[0]], color='black')

plt.show()

plt.plot(s\_values, s\_result)

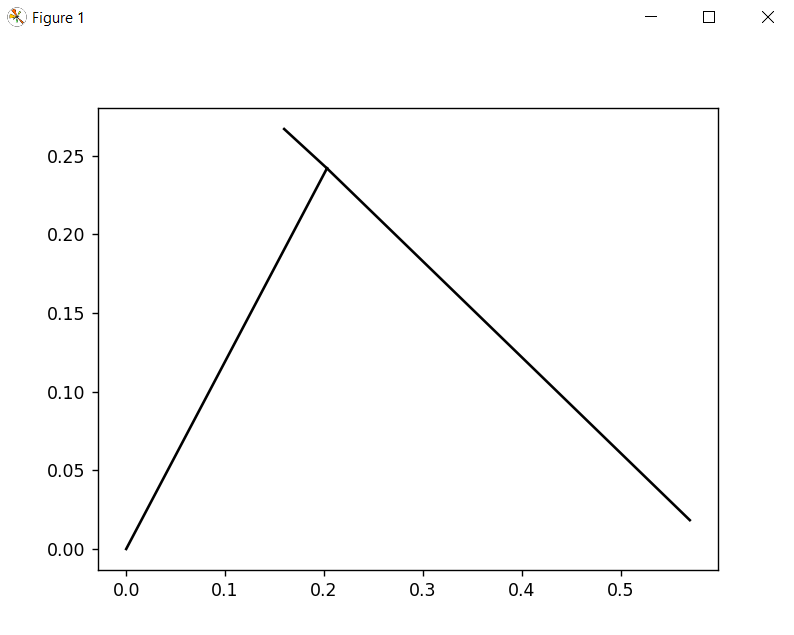
for x in crossings:

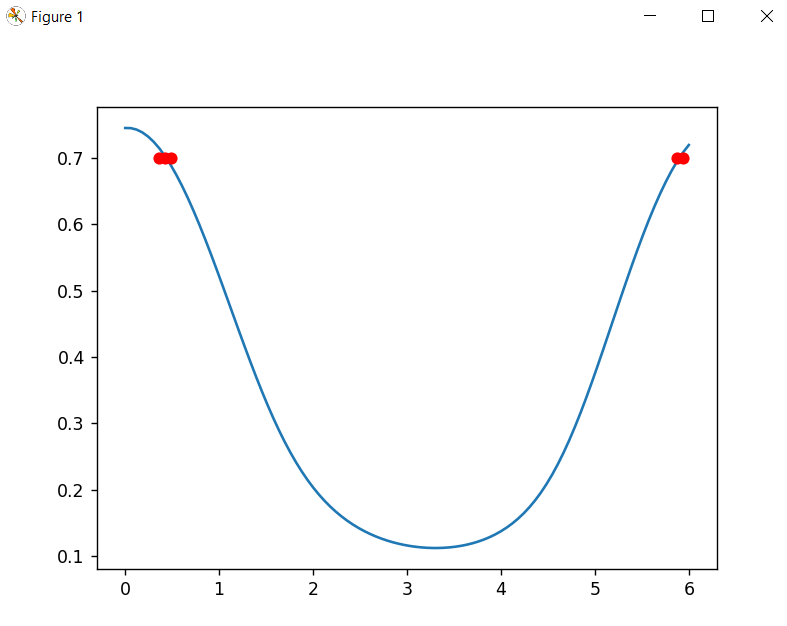
plt.plot(x, dot, 'ro')

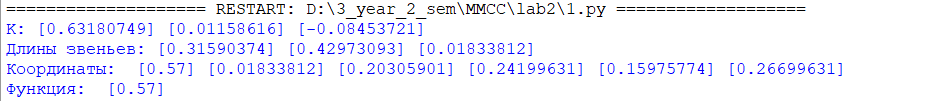
plt.show()

else: print("Механизм не существует")

**Результат выполнения:**

****

****

****

**Задание 3.**

1. Задавшись параметрами r , h (из таблицы), определите размер l, соответствующий требуемому сопротивлению R. Доказать графически, что значение l найдено верно.



1. Рассчитать значение размера l для 6 -7 значений из диапазона значений варьируемого параметра, указанного в таблице. Построить сводный график зависимости полученных значений размера l от варьируемого параметра.

|  |  |  |
| --- | --- | --- |
| Вариант | ***H*** | ***r*** |
| 2-2 |  | 0.01 – 0.02 |

**Код программы:**

import numpy as np

import matplotlib.pyplot as plt

G = 0.015

r\_values = [0.01, 0.0125, 0.015, 0.0175, 0.02]

h = 1.2

R\_target = 12

def calculate\_R(L, r):

return (np.log((L\*\*2)/(2\*r\*h)) + 4.95) / (2 \* np.pi \* L \* G)

L\_values = np.linspace(10, 15, 100)

plt.figure(figsize=(10, 6))

for r in r\_values:

R\_values = calculate\_R(L\_values, r)

plt.plot(L\_values, R\_values, label=f'R(L) для r={r}')

plt.axhline(y=R\_target, color='r', linestyle='--', label='Искомая R')

for r in r\_values:

R\_values = calculate\_R(L\_values, r)

index = np.argwhere(np.diff(np.sign(R\_values - R\_target))).flatten()

if len(index) > 0:

intersection\_L = L\_values[index[0]]

intersection\_R = R\_values[index[0]]

plt.plot(intersection\_L, intersection\_R, 'ro')

print(f"Пересечение найдено в точке L = {intersection\_L:.2f} для r = {r}")

else:

print(f"Не найдено пересечения для r = {r}.")

plt.xlabel('Длина (L)')

plt.ylabel('R')

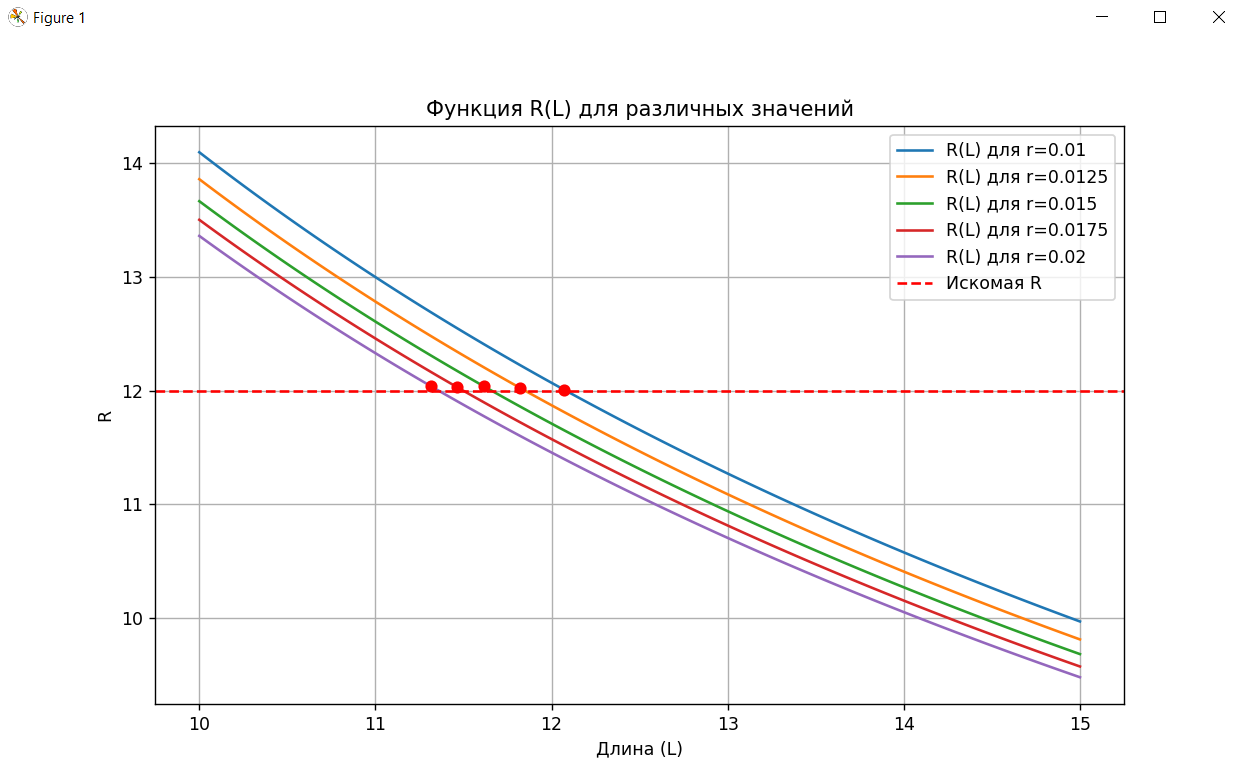
plt.title('Функция R(L) для различных значений')

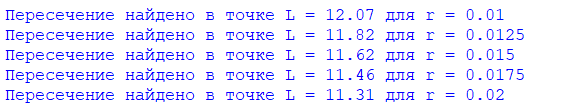
plt.legend()

plt.grid(True)

plt.show()

**Результат выполнения:**

****

****

**Вывод:** в результате выполнения лабораторной работы мноюполучены навыки компьютерного моделирования технических объектов, представленных в виде статической модели с выводом результатов моделирования в численном и графическом виде.