Отчет для лабораторной работы №4

НФИбд-02-18

Оразклычев Давут

Содержание

# Цель работы

Решение заданий: Постройте фазовый портрет гармонического осциллятора и решение уравнения гармонического осциллятора

# Задание

(рис. 1)

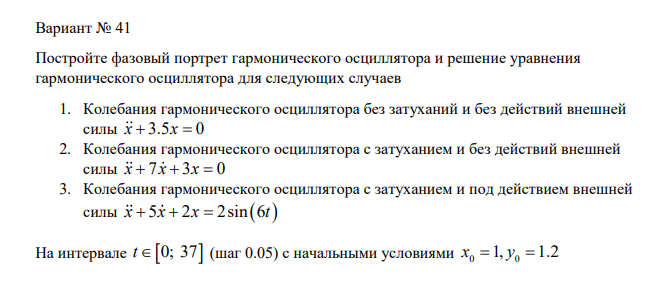


Figure 1: Задание

# Выполнение лабораторной работы

Импортируем библиотеки и переменные

import numpy as nmp  
import math  
import matplotlib.pyplot as plt  
  
from scipy.integrate import odeint  
  
a\_1 = 3.5  
b\_1 = 0.00  
  
Time\_null = 0  
Time\_Max = 37  
Step = 0.05

Создаем список t

t = nmp.arange(Time\_null, Time\_Max, Step)  
t = nmp.append(t, Time\_Max)

Создаем функции и уравнение:

def p(t):  
 return 0  
  
def syst(x,t):  
 return x[1], -a\_1 \* a\_1 \* x[0] - b\_1 \* x[1] - p(t)

Создаем вектор значений

v0 = (1, 1.2)  
  
yf = odeint(syst, v0, t)  
  
x = []  
y = []  
  
for i in range(len(yf)):  
 x.append(yf[i][0])  
 y.append(yf[i][1])

Показать результаты на дисплее

plt.figure(figsize = (10,10))  
plt.plot(x,y,'r', label = 'x')  
plt.show()

И получаем: (рис. 2)

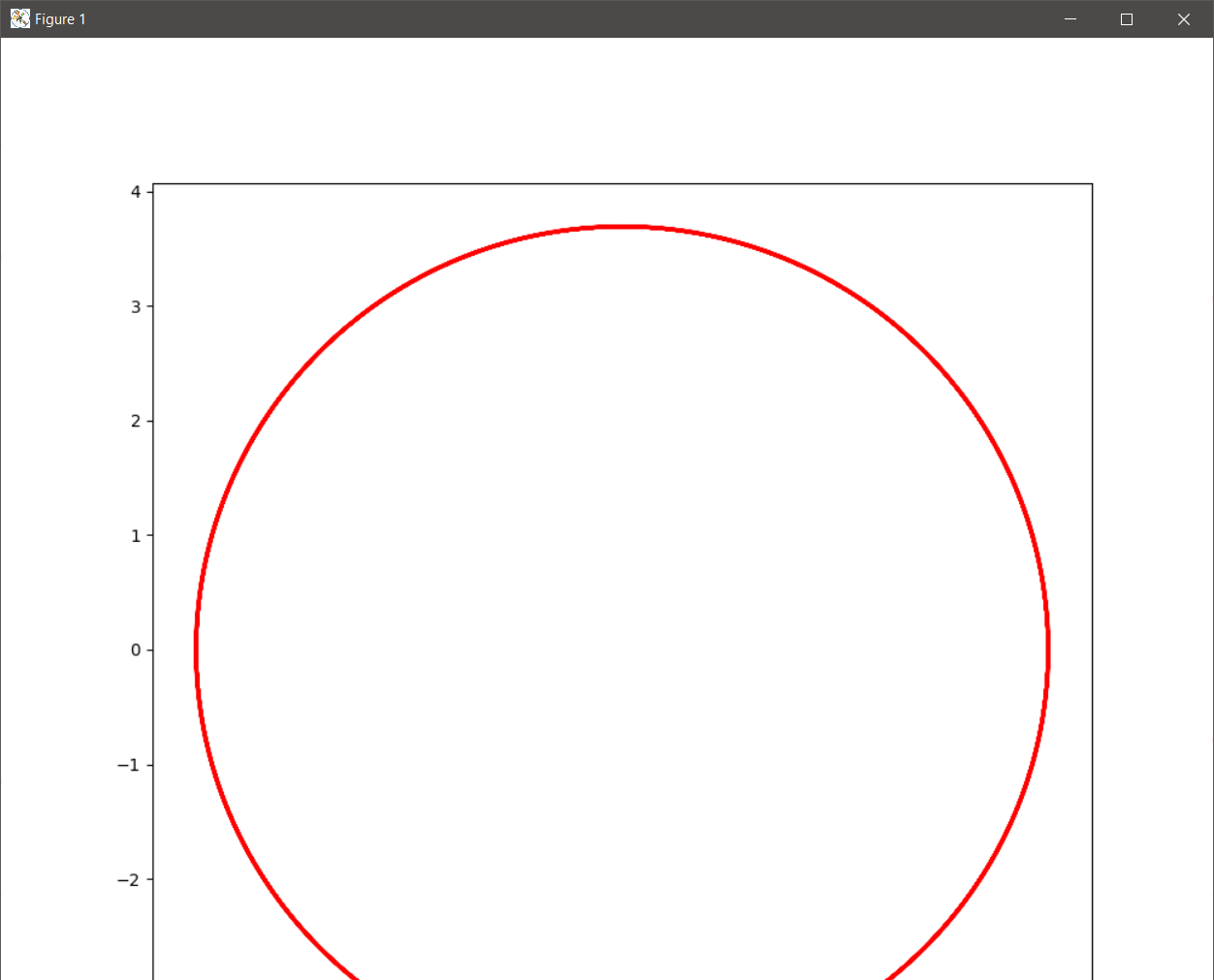


Figure 2: Колебания гармонического осциллятора без затуханий и без действий внешней силы, График 1

(рис. 3)

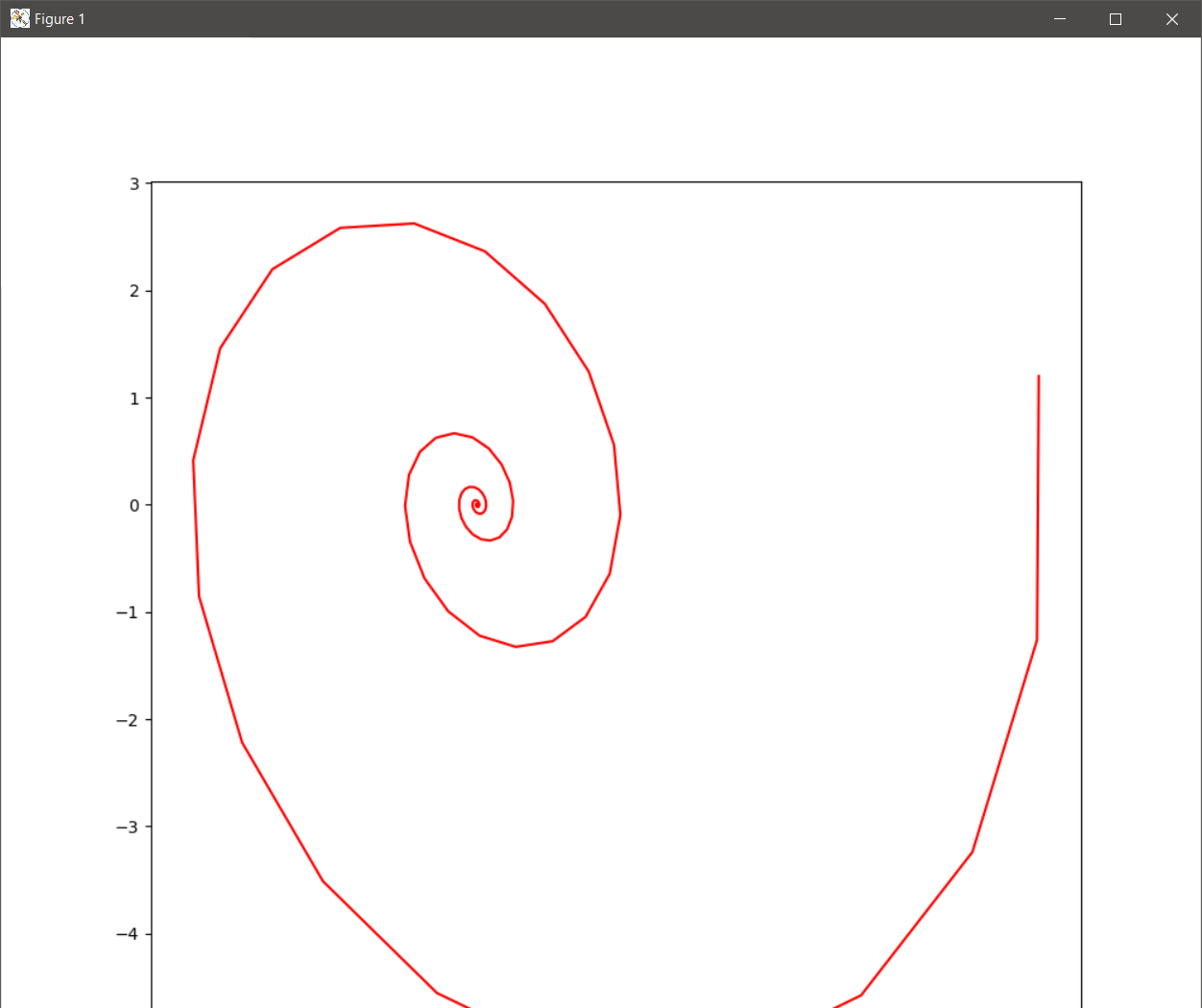


Figure 3: Колебания гармонического осциллятора c затуханием и без действий внешней силы, График 2

(рис. 4)

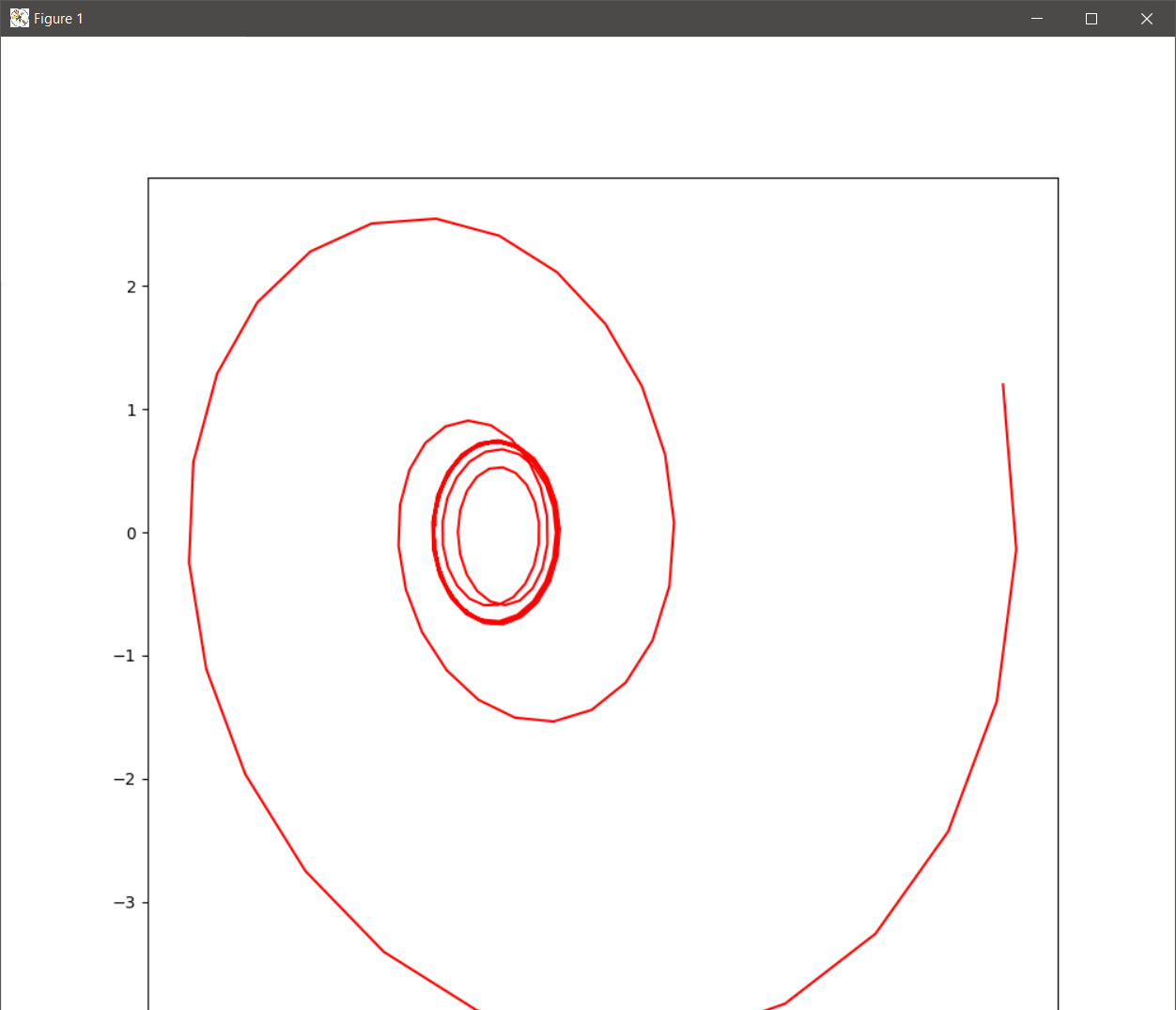


Figure 4: Колебания гармонического осциллятора c затуханием и под действием внешней силы, График 3

Код на Python для случая 2:

import numpy as np  
import math  
import matplotlib.pyplot as plt  
  
from scipy.integrate import odeint  
  
w = 8  
g = 0.00  
  
t0 = 0  
tmax = 45  
dt = 0.05  
  
t = np.arange(t0, tmax, dt)  
t = np.append(t, tmax)  
  
  
def p(t):  
 #return (math.sin(t\*0.5))  
 return 0  
  
  
def syst(x, t):  
 return x[1], -w \* w \* x[0] - g \* x[1] - p(t)  
  
  
v0 = (-1, 0)  
  
yf = odeint(syst, v0, t)  
  
x = []  
y = []  
  
for i in range(len(yf)):  
 x.append(yf[i][0])  
 y.append(yf[i][1])  
  
zero = []  
for i in range(len(t)):  
 zero = np.append(zero, 0)  
  
plt.figure(figsize=(10, 10))  
plt.plot(x, y, 'r', label='x')  
plt.show()

Код на Python для случая 1:

import numpy as nmp  
import math  
import matplotlib.pyplot as plt  
  
from scipy.integrate import odeint  
  
a\_1 = 3.5  
b\_1 = 0.00  
  
Time\_null = 0  
Time\_Max = 37  
Step = 0.05  
  
t = nmp.arange(Time\_null, Time\_Max, Step)  
t = nmp.append(t, Time\_Max)  
  
def p(t):  
 return 0  
  
def syst(x,t):  
 return x[1], -a\_1 \* a\_1 \* x[0] - b\_1 \* x[1] - p(t)  
  
v0 = (1, 1.2)  
  
yf = odeint(syst, v0, t)  
  
x = []  
y = []  
  
for i in range(len(yf)):  
 x.append(yf[i][0])  
 y.append(yf[i][1])  
  
plt.figure(figsize = (10,10))  
plt.plot(x,y,'r', label = 'x')  
plt.show()

Код на Python для случая 3:

import numpy as nmp  
import math  
import matplotlib.pyplot as plt  
  
from scipy.integrate import odeint  
  
a\_1 = 5  
b\_1 = 2.00  
  
Time\_null = 0  
Time\_Max = 37  
Step = 0.05  
  
t = nmp.arange(Time\_null, Time\_Max, Step)  
t = nmp.append(t, Time\_Max)  
  
def p(t):  
 return (2 \* math.sin(t\*6))  
 return 0  
  
def syst(x,t):  
 return x[1], -a\_1 \* a\_1 \* x[0] - b\_1 \* x[1] - p(t)  
  
v0 = (1, 1.2)  
  
yf = odeint(syst, v0, t)  
  
x = []  
y = []  
  
for i in range(len(yf)):  
 x.append(yf[i][0])  
 y.append(yf[i][1])  
  
plt.figure(figsize = (10,10))  
plt.plot(x,y,'r', label = 'x')  
plt.show()

# Выводы

Построили код на Python для решения осциллятора и решение уравнения гармонического осциллятора.

# Ответы на вопросы

Вопросы:

1. Запишите простейшую модель гармонических колебаний
2. Дайте определение осциллятора
3. Запишите модель математического маятника
4. Запишите алгоритм перехода от дифференциального уравнения второго порядка

к двум дифференциальным уравнениям первого порядка

1. Что такое фазовый портрет и фазовая траектория?

Ответы:

1. sin(x)
2. Система, которая при выведении её из положения равновесия испытывает действие возвращающей силы
3. ¨ф+w20ф=0
4. Дважды интегрируем и получаем общее решение
5. Геометрическое представление траекторий динамической системы на фазовой плоскости
6. Проекция интегральной кривой