Лабораторная работа

Построение и исследование компьютерных моделей с использованием дифференциальных уравнений

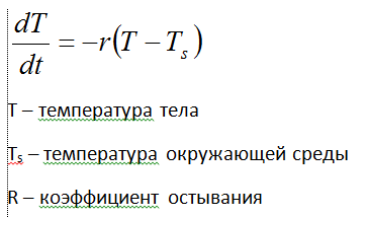
Задание 1. Остывание кофе.

Постановка задачи:

Природа переноса тепла от кофе к окружающему пространству сложна и

включает в себя механизмы конвекции, излучения, испарения и теплопроводности. Исследовать зависимость остывания кофе в чашке при следующих исходных данных t среды = 22 , t жидкости = 83,

коэффициент остывания r = 0,0373



Код программы:

import matplotlib.pyplot as plt

def fx(Ts, T):

r = 0.0373

Ts = 22

return -r \* (T - Ts)

fig = plt.figure()

print(u'Тип Figure %s' % type(fig))

grid1 = plt.grid(True)

a = 0

b = 30

n = 30

h = (b - a) / n

x = 0

y = 100

while x <= b - h:

k1 = fx(x, y)

k2 = fx(x + h / 2, y + h / 2 \* k1)

k3 = fx(x + h / 2, y + h / 2 \* k2)

k4 = fx(x + h, y + h \* k3)

y = y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4)

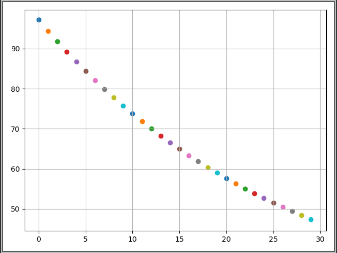
plt.scatter(x, y)

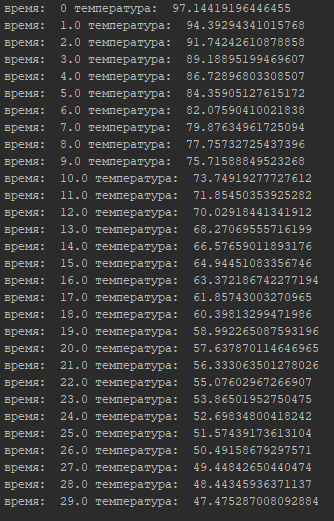
print("время: ", x, "температура: ", y)

x += h

plt.show()

Результат:





Задание 2. Задача о распаде радия.

Постановка задачи:

Установлено, что скорость распада радия прямо пропорциональна его

количеству в каждый данный момент. Определить закон изменения массы

радия в зависимости от времени, если при t = 0, масса радия была m0, к = 0,00044. Найти период полураспада радия.

Код программы:

import matplotlib.pyplot as plt

def fx(t, m):

k = 0.00044

return -k \* m

fig = plt.figure()

print(u'Тип Figure %s' % type(fig))

grid1 = plt.grid(True)

print (u'Список областей рисования после создания объекта fig %s \n' % fig.axes)

axes = plt.gca()

axes.set\_xlim([0, 1600])

axes.set\_ylim([0, 1.2])

a = 0

b = 1600

n = 160

h = (b - a) / n

x = 0

y = 1

while x <= b - h:

k1 = fx(x, y)

k2 = fx(x + h / 2, y + h / 2 \* k1)

k3 = fx(x + h / 2, y + h / 2 \* k2)

k4 = fx(x + h, y + h \* k3)

y = y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4)

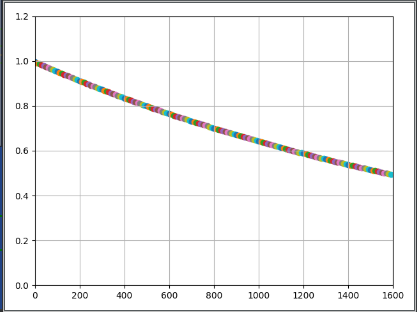
plt.scatter(x, y)

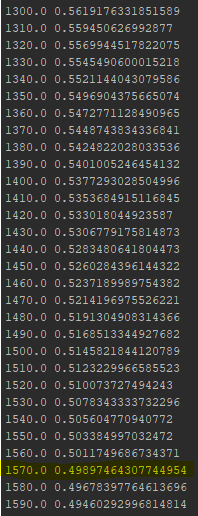
x += h

print(x, y)

plt.show()

Результат:





Задание 3.

Постановка задачи:

Проходя через лес и испытывая сопротивление деревьев, ветер теряет

часть своей скорости. На бесконечно малом пути эта потеря пропорциональна скорости в начале этого пути и длине его. Найти скорость ветра, прошедшего в лесу 150 м, зная, что до вступления в лес начальная скорость ветра v0=12 м/с; после прохождения в лесу пути s=1 м, скорость ветра уменьшилась до величины v1=11,8 м/с.

Код программы:

import matplotlib.pyplot as plt

import math

def fx(t, S):

k = - (math.log(11.8/12))

return -k\*S

fig = plt.figure()

print(u'Тип Figure %s' % type(fig))

grid1 = plt.grid(True)

a = 0

b = 150

n = 15

h = (b - a) / n

x = 0

y = 12

while x <= b - h:

k1 = fx(x, y)

k2 = fx(x + h / 2, y + h / 2 \* k1)

k3 = fx(x + h / 2, y + h / 2 \* k2)

k4 = fx(x + h, y + h \* k3)

y = y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4)

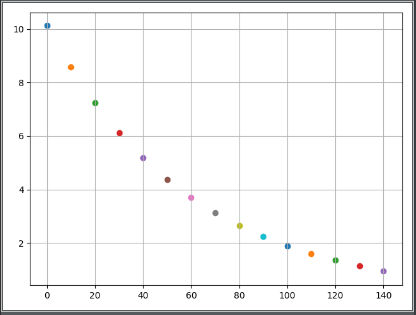
plt.scatter(x, y)

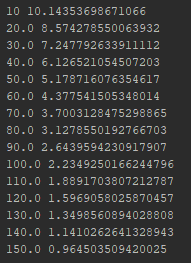
print(x, y)

x += h

plt.show()

Результат:





Задание 4.

Постановка задачи:

В цепи поддерживается напряжение E=300 В. Сопротивление цепи R=150

Ом. Коэффициент самоиндукции равен L=30 Гн. За какое время с момента

замыкания цепи возникающий в ней ток I достигнет 99% своей предельной величины.

Код программы:

import matplotlib.pyplot as plt

def fx(t, S):

e = 300.0

r = 150.0

l = 30.0

return -r \* (S / l)

fig = plt.figure()

print(u'Тип Figure %s' % type(fig))

grid1 = plt.grid(True)

a = 0

b = 1

n = 100

h = (b - a) / n

x = 0

y = 2

while x <= b - h:

k1 = fx(x, y)

k2 = fx(x + h / 2, y + h / 2 \* k1)

k3 = fx(x + h / 2, y + h / 2 \* k2)

k4 = fx(x + h, y + h \* k3)

y = y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4)

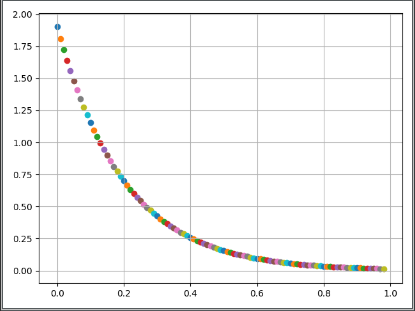
plt.scatter(x, y)

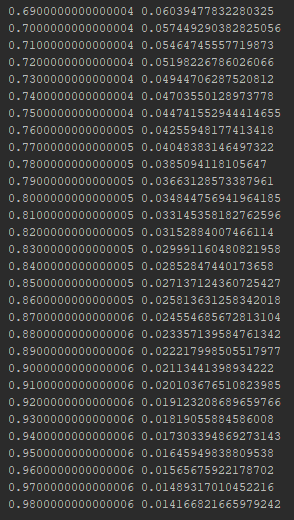
print(x, y)

x += h

plt.show()

Результат:





Задание 5.

Постановка задачи:

Исследовать зависимость остывания хлеба при:

T среды = 25 C

t хлеба = 100 C

Код программы:

import matplotlib.pyplot as plt

import math as m

def fx(t, T):

t\_s = 25

k = - m.log(7/15.0)/20.0

return -k \* (T - t\_s)

fig = plt.figure()

print(u'Тип Figure %s' % type(fig))

grid1 = plt.grid(True)

a = 0

b = 80

n = 80

h = (b - a) / n

x = 0

y = 100

while x <= b - h:

k1 = fx(x, y)

k2 = fx(x + h / 2, y + h / 2 \* k1)

k3 = fx(x + h / 2, y + h / 2 \* k2)

k4 = fx(x + h, y + h \* k3)

y = y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4)

plt.scatter(x, y)

print(x, y)

x += h

plt.show()

Результат:

