

```

1  #! python3.7
2  # -*- coding: utf-8 -*-
3  from numpy import zeros, linspace
4  from matplotlib.pyplot import style, figure, axes
5
6  def f(u, lambda) :
7      f = -lambda*u
8      return f
9
10 # Определение входных данных задачи
11 t_0 = 0.; T = 3600.
12 u_0 = 1000.; lambda = 8.75*10**(-3)
13
14 # Определение числа интервалов сетки,
15 # на которой будет искаться приближённое решение
16 M = 50
17
18 # Определение сетки
19 tau = (T - t_0)/M
20 t = linspace(t_0, T, M + 1)
21
22 # Выделение памяти под массив сеточных значений решения ОДУ
23 u = zeros(M + 1)
24
25 # Задание начального условия
26 u[0] = u_0
27
28 # Реализация схемы ERK1
29 for m in range(M):
30     u[m + 1] = u[m] + tau*f(u[m], lambda)
31
32 # Отрисовка решения
33 style.use('dark_background')
34
35 fig = figure()
36 ax = axes(xlim=(t_0, T), ylim=(0, 1000))
37 ax.set_xlabel('t'); ax.set_ylabel('u');
38 ax.plot(t, u, '-ow', markersize=5)
39 ax.set_title('График решения u(t)')
40
41 # Листинг программы, реализующей решение жёсткого ОДУ с помощью схемы ERK1
42 # (на примере моделирования задачи о радиоактивном распаде Таллия-210)
43
44 # Ti-210 (Таллий-210)
45 # Период полураспада 1,32 минуты
46 # Постоянная распада 8,75·10-3 с-1

```