

```

1  from numpy import *
2  from matplotlib.pyplot import *
3
4  # Функция f подготавливает массив, содержащий элементы вектор-функции,
5  # определяющей правую часть решаемой системы ОДУ
6  def f(u,g,k,mass):
7      f = zeros(4)
8      f[0] = u[2]
9      f[1] = u[3]
10     f[2] = 0. - k/mass*sqrt(u[2]**2 + u[3]**2)*u[2]
11     f[3] = -g - k/mass*sqrt(u[2]**2 + u[3]**2)*u[3]
12     return f
13
14 # Определение входных данных задачи
15 t_0 = 0.; T = 7.
16 x_0 = 0.; y_0 = 0.
17 v_0 = 150.; alpha = pi/4
18 g = 9.81; k = 10.; mass = 500.
19
20 # Определение числа интервалов сетки,
21 # на которой будет искаться приближённое решение
22 M = 50;
23
24 # Определение сетки
25 tau = (T - t_0)/M
26 t = linspace(t_0,T,M + 1)
27
28 # Выделение памяти под массив сеточных значений решения системы ОДУ
29 # В строке с номером m этого массива хранятся сеточные значения решения,
30 # соответствующие моменту времени t_m
31 u = zeros((M + 1,4))
32
33 # Задание начальных условий
34 # (записываются строку с номером 0 массива u)
35 u[0,0] = x_0; u[0,1] = y_0
36 u[0,2] = v_0*cos(alpha); u[0,3] = v_0*sin(alpha)
37
38 # Реализация схемы Эйлера
39 for m in range(M):
40     u[m + 1] = u[m] + tau*f(u[m],g,k,mass)
41
42 # Отрисовка решения
43 figure()
44 plot(u[:,0],u[:,1],'-ro',markersize=5)
45 title('Траектория движения тела')
46 xlabel('x'); ylabel('y')
47 xlim((0,1.62*80)); ylim((0,80))

```

Комментарий к файлу:

Листинг программы, реализующей схему Эйлера для решения системы ОДУ, возникающей при моделировании движения тела, брошенного под углом к горизонту в поле тяжести с учётом сопротивления воздуха.