

```

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, lambd):
7      f = zeros(2)
8      f[0] = lambd*u[0]/sqrt(1 + (lambd*u[0])**2)
9      f[1] = 1/sqrt(1 + (lambd*u[0])**2)
10     return f
11
12     # Определение входных данных задачи
13     t_0 = 0.; T = 1.; lambd = -10.; u_0 = 1.
14
15     # Определение шага сетки вдоль интегральной кривой
16     dl = 0.05
17
18     # Выделение памяти под массив сеточных значений решения системы ОДУ
19     # В строке с номером m этого массива хранятся сеточные значения решения,
20     # соответствующие моменту времени t_m
21     # Число J, определяющее число строк массива, задаём с запасом
22     J = 1000
23     u = zeros((J,2))
24
25     # Задание начальных условий
26     # (записываются строку с номером 0 массива u)
27     u[0,0] = u_0; u[0,1] = t_0
28
29     # Реализация схемы Эйлера
30     j = 0
31     while u[j,1] < T:
32         u[j + 1] = u[j] + dl*f(u[j], lambd)
33         j = j + 1
34
35     # Отрисовка решения
36     figure()
37     # Приближённое решение
38     plot(u[1:j + 1,1],u[1:j + 1,0], '-ro', markersize=5, label='приближённое решение')
39     # Точное решение
40     plot(u[1:j + 1,1],u_0*exp(lambd*(u[1:j + 1,1]-t_0)), '-g', label='точное решение')
41     title('Результат с использованием перехода к длине дуги кривой')
42     xlabel('t'); ylabel('u')
43     xlim((t_0,T)); ylim((0,u_0))
44     legend()

```

Комментарий к файлу:

Листинг программы, реализующей решение задачи Коши для системы ОДУ, возникающей при переходе к длине дуги кривой в Примере 4 с выполнением процедуры формальной автономизации. Для простоты реализована схема Эйлера.