Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 1

Тема Методы Пикара, Эйлера, Рунге-Кутта

Студент Брянская Е.В.

Группа ИУ7-62Б

Оценка (баллы) ________________

Преподаватель Градов В.М.

Москва.
2021 г.

Задание

Тема. Программная реализация приближенного аналитического метода и численных алгоритмов первого и второго порядков точности при решении задачи Коши для ОДУ.

Цель работы. Получение навыков решения задачи Коши для ОДУ методами Пикара и явными методами первого порядка точности (Эйлера) и второго порядка точности (Рунге-Кутта).

Исходные данные.
ОДУ, не имеющее аналитического решения:

$$\begin{cases} u'(x) = x^2 + u^2, \\ u(0) = 0 \end{cases} \tag{1}$$

Результат работы программы. Таблица, содержащая значения аргумента с заданным шагом в интервале $[0, x_{max}]$ и результаты расчета функции $u(x)$ в приближениях Пикара (от 1-го до 4-го), а также численными методами. Границу интервала x_{max} выбирать максимально возможной из условия, чтобы численные методы обеспечивали точность вычисления решения уравнения $u(x)$ до второго знака после запятой.

Описание алгоритмов

Задача Коши

Общее решение дифференциального уравнения n-ого порядка зависит от n констант. Требуется задать n дополнительных условий:

$$u(x) = \phi(x, c_1, c_2, ...c_n) \tag{2}$$

В задаче Коши все дополнительные условия задаются в одной точке ξ:

$$u_k(\xi) = \eta_k, k = 1, ...n \tag{3}$$

Задачу Коши можно решить с помощью следующих алгоритмов.

Приближённый аналитический метод Пикара

$$\begin{cases} u'(x) = f(x, u), \\ u(\xi) = \eta \end{cases} \tag{4}$$

$$u(x) = \eta + \int\limits_{\xi}^{x} f(t, u(t))dt \tag{5}$$

Получается, что

$$y^{(s)}(x) = \eta + \int\limits_{\xi}^{x} f(t, y^{(s-1)}(t))dt \tag{6}$$

$$y^{(0)} = \eta \tag{7}$$

Найдём 1, 2, 3 и 4 приближение для (1).

$$y^{(1)} = 0 + \int\limits_{0}^{x} t^2 dt = \frac{t^3}{3}\Big|_0^x = \frac{x^3}{3} \tag{8}$$

$$y^{(2)} = 0 + \int\limits_{0}^{x} [(\frac{t^3}{3})^2 + t^2]dt = \frac{t^7}{63}\Big|_0^x + \frac{t^3}{3}\Big|_0^x = \frac{x^7}{63} + \frac{x^3}{3} \tag{9}$$

$$\begin{gathered} y^{(3)} = 0 + \int\limits_{0}^{x} [(\frac{t^3}{3} + \frac{t^7}{63})^2 + t^2]dt = \frac{t^{15}}{15 \cdot 63^2}\Big|_0^x + \frac{2 \cdot t^{11}}{3 \cdot 63 \cdot 11}\Big|_0^x + \frac{t^7}{63}\Big|_0^x + \frac{t^3}{3}\Big|_0^x = \\ = \frac{x^{15}}{59535} + \frac{2 \cdot x^{11}}{2079} + \frac{x^7}{63} + \frac{x^3}{3} \end{gathered} \tag{10}$$

$$\begin{gathered} y^{(4)} = 0 + \int\limits_{0}^{x} [(\frac{t^{15}}{59535} + \frac{2 \cdot t^{11}}{2079} + \frac{t^7}{63} + \frac{t^3}{3})^2 + t^2]dt = \frac{x^{31}}{109\,876\,902\,975} + \frac{4 \cdot x^{27}}{3\,341\,878\,155} + \\ + \frac{4 \cdot x^{23}}{99\,411\,543} + \frac{2 \cdot x^{23}}{86\,266\,215} + \frac{2 \cdot x^{19}}{3\,393\,495} + \frac{4 \cdot x^{19}}{2\,488\,563} + \frac{4 \cdot x^{15}}{93\,555} + \frac{x^{15}}{59\,535} + \frac{2 \cdot x^{11}}{2079} + \frac{x^7}{63} + \frac{x^3}{3} \end{gathered} \tag{11}$$

Реализация представлена на листинге 1.

Кроме того, поставленную задачу можно решить с помощью численных методов.

Метод Эйлера

Явная схема выглядит следующим образом (12).

$$y_{n+1} = y_n + hf(x_n, y_n) \tag{12}$$

В этом случае нужно выбирать шаг, так как он определяет точность и устойчивость. Реализация представлена на листинге 1.

Метод Рунге-Кутта

Будем рассматривать метод второго порядка точности.

$$y_{n+1} = y_n + h[(1-\alpha)k_1 + \alpha k_2], \tag{13}$$

где

$$\begin{gathered} k_1 = f(x_n, y_n), \\ k_2 = f(x_n + \frac{h}{2\alpha}, y_n + \frac{h}{2\alpha}k_1), \alpha = \frac{1}{2} \text{ или } \alpha = 1 \end{gathered} \tag{14}$$

Реализация представлена на листинге 1.

Демонстрация работы программы

Результаты работы программы на отрезке $[0; 2]$ с шагом 10^{-6} представлены в таблице ниже (отражены только значения x с шагом 0.05).

x	Метод Пикара (1 приб-е)	Метод Пикара (2 приб-е)	Метод Пикара (3 приб-е)	Метод Пикара (4 приб-е)	Метод Эйлера	Метод Рунге-Кутта
0.00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
0.05	4.17e-05	4.17e-05	4.17e-05	4.17e-05	4.17e-05	4.17e-05
0.10	3.33e-04	3.33e-04	3.33e-04	3.33e-04	3.33e-04	3.33e-04
0.15	1.13e-03	1.13e-03	1.13e-03	1.13e-03	1.13e-03	1.13e-03
0.20	2.67e-03	2.67e-03	2.67e-03	2.67e-03	2.67e-03	2.67e-03
0.25	5.21e-03	5.21e-03	5.21e-03	5.21e-03	5.21e-03	5.21e-03
0.30	9.00e-03	9.00e-03	9.00e-03	9.00e-03	9.00e-03	9.00e-03
0.35	1.43e-02	1.43e-02	1.43e-02	1.43e-02	1.43e-02	1.43e-02
0.40	2.13e-02	2.14e-02	2.14e-02	2.14e-02	2.14e-02	2.14e-02
0.45	3.04e-02	3.04e-02	3.04e-02	3.04e-02	3.04e-02	3.04e-02
0.50	4.17e-02	4.18e-02	4.18e-02	4.18e-02	4.18e-02	4.18e-02
...	...	...	...	...	...	...
0.80	1.71e-01	1.74e-01	1.74e-01	1.74e-01	1.74e-01	1.74e-01
0.85	2.05e-01	2.10e-01	2.10e-01	2.10e-01	2.10e-01	2.10e-01
0.90	2.43e-01	2.51e-01	2.51e-01	2.51e-01	2.51e-01	2.51e-01
0.95	2.86e-01	2.97e-01	2.97e-01	2.97e-01	2.97e-01	2.97e-01
1.00	3.33e-01	3.49e-01	3.50e-01	3.50e-01	3.50e-01	3.50e-01
1.05	3.86e-01	4.08e-01	4.10e-01	4.10e-01	4.10e-01	4.10e-01
1.10	4.44e-01	4.75e-01	4.77e-01	4.78e-01	4.78e-01	4.78e-01
1.15	5.07e-01	5.49e-01	5.54e-01	5.54e-01	5.54e-01	5.54e-01
1.20	5.76e-01	6.33e-01	6.40e-01	6.41e-01	6.41e-01	6.41e-01
1.25	6.51e-01	7.27e-01	7.38e-01	7.40e-01	7.40e-01	7.40e-01
1.30	7.32e-01	8.32e-01	8.50e-01	8.53e-01	8.53e-01	8.53e-01
1.35	8.20e-01	9.50e-01	9.77e-01	9.82e-01	9.83e-01	9.83e-01
1.40	9.15e-01	1.08e+00	1.12e+00	1.13e+00	1.13e+00	1.13e+00
1.45	1.02e+00	1.23e+00	1.29e+00	1.31e+00	1.31e+00	1.31e+00
1.50	1.12e+00	1.40e+00	1.49e+00	1.51e+00	1.52e+00	1.52e+00
1.55	1.24e+00	1.58e+00	1.71e+00	1.76e+00	1.77e+00	1.77e+00
1.60	1.37e+00	1.79e+00	1.98e+00	2.05e+00	2.08e+00	2.08e+00
1.65	1.50e+00	2.03e+00	2.29e+00	2.41e+00	2.47e+00	2.47e+00
1.70	1.64e+00	2.29e+00	2.67e+00	2.86e+00	2.97e+00	2.97e+00
1.75	1.79e+00	2.58e+00	3.11e+00	3.42e+00	3.67e+00	3.67e+00
1.80	1.94e+00	2.92e+00	3.65e+00	4.15e+00	4.69e+00	4.69e+00
1.85	2.11e+00	3.29e+00	4.29e+00	5.10e+00	6.35e+00	6.35e+00
1.90	2.29e+00	3.71e+00	5.08e+00	6.37e+00	9.57e+00	9.57e+00
1.95	2.47e+00	4.17e+00	6.04e+00	8.10e+00	1.87e+01	1.87e+01
2.00	2.67e+00	4.70e+00	7.22e+00	1.05e+01	3.17e+02	3.18e+02

Вопросы при защите лабораторной работы

1. **Укажите интервалы значений аргумента, в которых можно считать решением заданного уравнения каждое из первых 4-х приближений Пикара. Точность результата оценивать до второй цифры после запятой. Объяснить свой ответ.**

   Левая граница у каждого из четырёх интервалов равна 0. Для того, чтобы определить правую, нужно сравнить полученные значения для разных приближений или с численными методами. Сравнивая значение текущего приближения со значением, которое было расчитано методом более высокого порядка, можно определить правую границу (это последнее значение, при котором наблюдается совпадение результатов).

   Проведя анализ были получены следующие интервалы:

   - Первое приближение: [0; 0.88]
   - Второе приближение: [0; 1.17]
   - Третье приближение: [0; 1.40]
   - Четвёртое приближение: [0; 1.47]

2. **Пояснить, каким образом можно доказать правильность полученного результата при фиксированном значении аргумента в численных методах.**

   В силу того, что численные методы зависят от величины шага, то изменяя его, можно прийти к наиболее точному результату при фиксированном значении аргумента. Как только результат перестаёт отличатся от результатов, полученных ранее, то можно сделать вывод о том, что корректный результат получен.

3. **Каково значение функции при x=2, т.е. привести значение u(2).**

   При $x = 2$ и шаге 10^{-6} было получено значение функции равное 317.82.

Код программы

Листинг 1 — Лабораторная работа №1

```
from math import sqrt

def f(x, y):
  return x**2 + y**2

def picard_1(x_args):
  res = []
  for x in x_args:
    res.append(x**3 / 3)
  return res

def picard_2(x_args):
  res = []
  for x in x_args:
    res.append(x**3 / 3 + x**7 / 63)
  return res

def picard_3(x_args):
  res = []
  for x in x_args:
    res.append(x**3 / 3 + x**7 / 63 + x**15 / 59535 + 2*x**11 / 2079)
  return res

def picard_4(x_args):
  res = []
  for x in x_args:
    res.append(x**3/3 + x**7/63 + x**15/59535 + 2*x**11/2079 +
    x**31/109876902975 + 4*x**23/99411543 + 4*x**27/3341878155 + 2*x**23/86266215 + 2*x**19/3393495 + 4*x**19/2488563 + 4*x**15/93555)
  return res

def runge_kutta(x, y, h, num):
  alpha = 0.5
  res = []
  temp = h / (2 * alpha)

  for i in range(num):
    res.append(y)
```

```

    k1 = f(x, y)
    k2 = f(x + temp, y + temp * k1)

    y += h * ((1 - alpha) * k1 + alpha * k2)
    x += h

    return res


def euler_explicit(x, y, h, num):
    res = []

    for i in range(num):
        res.append(y)

    try:
        y += h * f(x, y)
        x += h
    except OverFlowError:
        for k in range(i, num):
            res.append('-----')
        break
    return res

def count_x_args(x, x_max, h):
    x_args = []
    while x <= x_max:
        x_args.append(x)
        x += h
    return x_args


def print_head():
    print(' '*4+'x'+' '*4+'|'+' '*17+'Метод Пикара'+' '*18+'|'+' '*6+'
     Метод Эйлера'+' '*5+'|'+' '*3+'Метод РунгеКутта-'+' '*3+'\n'+' '*9+'|'+'
      '*5+'1'+' '*5+'|'+' '*5+'2'+' '*5+'|'+' '*5+'3'+' '*5+'|'+' '*5+'4'+
     ' '*5+'|'+' '*3+'Явный'+' '*3+'\n'+'-'*55)


```

```
def main():
  print_head()

  x, x_max, y = 0, 2, 0
  h = 10 ** -6

  x_args = count_x_args(x, x_max, h)

  res_runge_kutta = runge_kutta(x, y, h, num)
  res_euler_explicit = euler_explicit(x, y, h, num)
  res_picard_1 = picard_1(x_args)
  res_picard_2 = picard_2(x_args)
  res_picard_3 = picard_3(x_args)
  res_picard_4 = picard_4(x_args)

  for i in range(len(x_args)):
    print('{:9.3f}|{:11.3e}|{:11.3e}|{:11.3e}|{:11.3e}|{:11.3e}|{:11.3e}'.format(x_args[i], res_picard_1[i], res_picard_2[i], res_picard_3[i], res_picard_4[i], res_euler_explicit[i], res_runge_kutta[i]))


if __name__ == '__main__':
  main()
```