A picture containing logo

Description automatically generated

Лабораторная работа №6

по Вычислительной Математики

Вариант 13

Выполнил:

Пурэвсурэн Билгуун

Группа Р3213

Преподователь:

Малышева Татьяна Алексеевна

г. Санкт-Петербург 2022г

1. **Цель работы:**

Решить задачу Коши численными методами.

**Для исследования использовать:**

* Одношаговые методы;
* Многошаговые методы.

1. **Программная реализация задачи:**
   1. Исходные данные:
      1. ОДУ вида начальные условия , интервал дифференцирования , шаг h, точность ε
   2. Составить таблицу приближенных значений интеграла дифференциального уравнения, удовлетворяющего начальным условиям. Для оценки точности использовать правило Рунге.
   3. Построить графики точного решения и полученного численного решения (разными цветами).
2. **Описание метода, расчетные формулы:**

**Метод Эйлера** (1707–1783) основан на разложении искомой функции в ряд Тейлора в окрестностях узлов

**Модификационный метод Эйлера** или метод Эйлера с пересчетом заменяет правую часть уравнения на среднюю арифметическую значений функции f(x,Y) в точках (xi,yi) и (xi+1,yi+1)

**Метод Рунге-Кутта** требует большего объема вычислений по сравнению с ме- тодом Эйлера и его модификацией, однако это окупается повышенной точностью, что да- ет возможность проводить счет с большим шагом. Другими словами, для получения ре- зультатов с одинаковой точностью в методе Эйлера потребуется значительно меньший шаг, чем в методе Рунге-Кутта.

**Text, letter

Description automatically generated** **Метод Адамса** другой путь построения разностных схем основан на том, что для вычисления значения yi+1 используются результаты не одного, а k предыдущих шагов.

Diagram

Description automatically generated with low confidence

**Метод Милна** относится к многошаговым методам и представляет один из методов прогноза и коррекции.

Text, letter

Description automatically generated

1. **Реализация расчета:**

Модифиционный метод Эйлера:

def euler\_method(f,a,b,y0,h):

""" Усовершенствованный метод Эйлера """

dots = [(a,y0)]

n = int((b-a) / h)

for i in range(1, n+1):

h2 = h / 2

y = dots[i-1][1] + h \* f(dots[i-1][0] + h2, dots[i-1][1] + h2 \* f(dots[i-1][0],dots[i-1][1]))

dots.append((dots[i-1][0] + h,y))

return dots

Метод Милна:

def milna\_method(f,a,b,y0,h):

""" Метод Милна """

n = int((b-a) / h)

b0 = min(b, a + 3 \* h)

dots = euler\_method(f,a, b0, y0, h)

if len(dots) <= 3:

return dots

for i in range(4, n+1):

y\_prognoz = dots[i-4][1] + ((4\*h)/3)\*(2\*f(dots[i-3][0], dots[i-3][1]) - f(dots[i-2][0],dots[i-2][1]) + 2\*f(dots[i-1][0],dots[i-1][1]))

y\_correct = dots[i-2][1] + (h/3) \* (f(dots[i-2][0],dots[i-2][1]) + 4 \* f( dots[i-1][0],dots[i-1][1]) + f(dots[i-1][0]+h, y\_prognoz))

dots.append((dots[i-1][0]+h, y\_correct))

return dots

Правило Рунге:

def get\_error(f,a,b,y0,h,p):

n = int((b-a) / h)

error = [0]

y = 0

if p == 2:

y2 = euler\_method(f,a,b,y0,h)

b2 = 1

j = 0

while True:

y = euler\_method(f,a,b2,y0,2\*h)

if len(y) >= 2 \* len(y2):

break

b2+=1

for i in range(1,len(y2)\*2):

if i % 2 == 0:

error.append(np.abs((y[i][1] - y2[j][1])/3))

else:

j += 1

elif p == 4:

y2 = milna\_method(f,a,b,y0,h)

b2 = 1

j = 0

while True:

y = milna\_method(f,a,b2,y0,2\*h)

if len(y) >= 2 \* len(y2):

break

b2+=1

for i in range(1,len(y2)\*2):

if i % 2 == 0:

error.append(np.abs((y[i][1] - y2[j][1])/15))

else:

j += 1

return error

1. **Результат программы**

Text

Description automatically generated

Table

Description automatically generated with medium confidence

1. **Вывод**

В результате выполнения данной лабораторной работой я познакомился с методами численного дифференцироваэия и реализовал усовершенствованный метод Эйлера и метод Милна на языке программирования Python, закрепив знания.

Одношаговые методы является более неточными чем у многошаговых. При том простой метод Эйлера является самым неточным, но самым простым.