**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

Факультет прикладной математики и информатики

Юревич Александр Николаевич

Вариант 11

Отчет по лабораторной работе №4

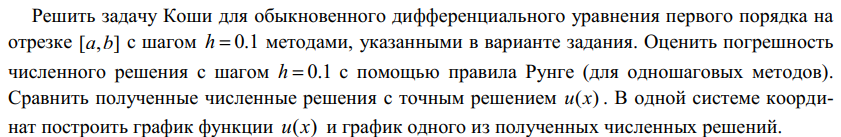
«Численные методы решения задачи Коши» студента 2 курса 13 группы

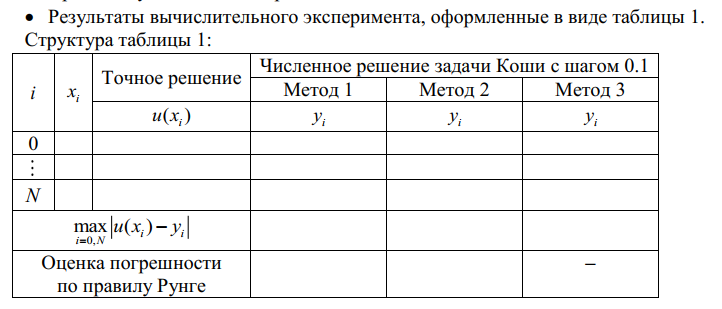
**Преподаватель**

Горбачева Ю. Н.

Минск 2024

Постановка задачи





1.Неявный метод трапеций

def iterNewton(i):

y0 = solution.loc[i-1, "yi-M1"]

yk = y0

x0 = solution.loc[i-1, "xi"]

xk = solution.loc[i, "xi"]

f0 = solution.loc[i-1, "f(x,y)"]

while True:

yk1 = yk - (yk - y0 - h/2 \* (f0 + u(xk,yk)))/(1 - h/2 \* u\_der(xk,yk))

if abs(yk1 - yk) <= 10\*\*(-10):

solution.loc[i, "yi-M1"] = yk1

solution.loc[i, "f(x,y)"] = u(xk, yk1)

return

else:

yk = yk1

Неявный метод трапеций в общем виде выглядит следующим образом:

Для решения полученного неявного уравнения на практике используется итерационный метод Ньютона, который в общем виде выглядит следующим образом:

Но в нашем случае примет вид:

Полученная погрешность:



Данный метод является методом второго порядка точности:

2. Явный метод средних прямоугольников.

def EMM():

for i in range(0,int(1/h)):

xi = solution.loc[i,"xi"]

yi = solution.loc[i, "yi-M2"]

k1 = u(xi,yi)

k2 = u(xi+h/2, yi + h/2\*k1)

solution.loc[i+1,"yi-M2"] = yi + h\*k2

Явный метод средних прямоугольников выглядит следующим образом:

Полученная погрешность:



Данный метод является методом второго порядка точности:

3. Предиктор-корректорный метод Адамса

2-го порядка.

def explAdams(i):

yi1 = solution.loc[i-1, "yi-M3"]

yi = solution.loc[i, "yi-M3"]

xi1 = solution.loc[i-1, "xi"]

xi = solution.loc[i, "xi"]

solution.loc[i+1,"yi-M3"] = yi + h\*(3/2\*u(xi, yi) - 1/2\*u(xi1, yi1))

Сначала мы вычисляем значение используя явный метод Адамса 2-го порядка:

def implAdams(i):

yi1 = solution.loc[i+1, "yi-M3"]

yi = solution.loc[i, "yi-M3"]

xi1 = solution.loc[i+1, "xi"]

xi = solution.loc[i, "xi"]

solution.loc[i+1,"yi-M3"] = yi + h/2\*(u(xi1,yi1) + u(xi, yi))

Затем уточняем полученное решение с помощью неявного метода Адамса 2-го порядка, где — полученный из явного метода Адамса:

Полученная погрешность:



Данный метод является методом второго порядка точности:

4. Вычисление погрешности по правилу Рунге

def Runge():

R1 = np.max(np.abs(solutionx2.loc[range(0,11,2),"yi-M1"] - np.array(solution.loc[range(0,6), "yi-M1"]))) / 3

R2 = np.max(np.abs(solutionx2.loc[range(0,11,2),"yi-M2"] - np.array(solution.loc[range(0,6), "yi-M2"]))) / 3

R3 = np.max(np.abs(solutionx2.loc[range(0,11,2),"yi-M3"] - np.array(solution.loc[range(0,6), "yi-M3"]))) / 3

return np.array([np.NAN, np.NAN, R1, R2, R3])r

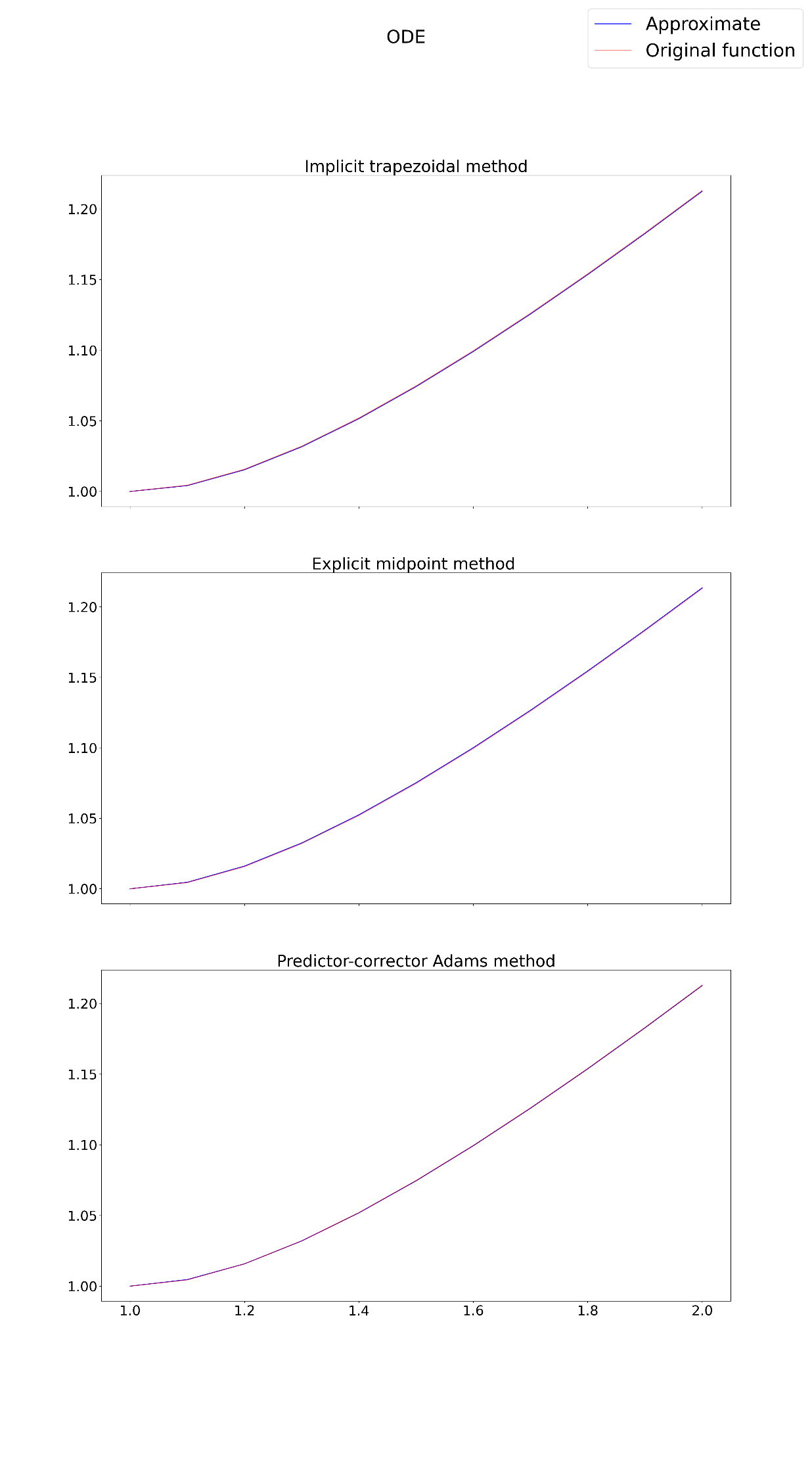
Данный код реализует правило Рунге для расчета апостериорной погрешности:

Где p — степень точности метода

5.Таблица результатов

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i** | **xi** | Точное решение | Решение задач Коши с шагом 0.1 | | |
| Метод 1 | Метод 2 | Метод 3 |
| **ui** | **yi-M1** | **yi-M2** | **yi-M3** |
| **0** | 1 | 1 | 1 | 1 | 1 |
| **1** | 1.1 | 1.004410788 | 1.00416079 | 1.00464668 | 1.004646682 |
| **2** | 1.2 | 1.01577807 | 1.01538909 | 1.01613922 | 1.015796554 |
| **3** | 1.3 | 1.032099455 | 1.03163111 | 1.03252873 | 1.031989365 |
| **4** | 1.4 | 1.05206821 | 1.05155411 | 1.05253455 | 1.051879485 |
| **5** | 1.5 | 1.074796966 | 1.07425679 | 1.07528283 | 1.074558783 |
| **6** | 1.6 | 1.099662846 | 1.09910848 | 1.10015803 | 1.099392973 |
| **7** | 1.7 | 1.12621623 | 1.12565498 | 1.12671469 | 1.125925855 |
| **8** | 1.8 | 1.154124699 | 1.15356113 | 1.15462283 | 1.153821063 |
| **9** | 1.9 | 1.183137343 | 1.18257432 | 1.18363299 | 1.182825237 |
| **10** | 2 | 1.213061319 | 1.21250063 | 1.21355323 | 1.212743968 |
| **Правило Рунге** | | | 0.0005636 | 0.00054097 | 0.000510857 |
| max *u*(*xi* )  *yi*  *i*0,*10* | | | 0.00056357 | 0.00049846 | 0.000317352 |

6.Графики решений



Выводы

В данной работе были выполнены следующие задачи:

* Реализованы: Неявный метод трапеций; явный метод средних прямоугольников; предиктор-корректорный метод Адамса 2-го порядка.
* Построили 3 графика для визуальной проверки полученного решения
* Построили таблицу погрешностей полученных решений

На основе выполненной работы можно сделать следующие выводы:

Данные методы являются эффективными методами нахождения приблизительного решения ОДУ с задачей Коши. В зависимости от усложнения метода возрастает и его эффективность. Также убедились, что по правилу Рунге можно лишь приблизительно найти ожидаемую погрешность метода.