МIНIСТЕРСТВО ОСВIТИ I НАУКИ УКРАЇНИ

НАЦIОНАЛЬНИЙ ТЕХНIЧНИЙ УНIВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛIТЕХНIЧНИЙ IНСТИТУТ»

Кафедра прикладної математики

Звіт

із лабораторної роботи №3

з дисципліни «Алгоритми і системи комп’ютерної математики.

Математичні алгоритми»

на тему:

«Метод Рунге-Кутта четвертого порядку»

|  |  |
| --- | --- |
| Виконав: | Керівник: |
| студент групи КМ-63 | *Старший викладач Бай Ю.П.* |
| *Артеменко Я.К.* |  |

Київ — 2019

# **ЗМІСТ**

[**1 ВСТУП** 2](#_Toc23016845)

[**2 ОСНОВНА ЧАСТИНА** 3](#_Toc23016846)

[**2.1 Завдання** 3](#_Toc23016847)

[**2.2 Описання методу** 3](#_Toc23016848)

[**2.3 Використані тест кейси** 4](#_Toc23016849)

[**3 ВИСНОВКИ** 6](#_Toc23016850)

[**ПЕРЕЛІК ПОСИЛАНЬ** 7](#_Toc23016851)

[**ДОДАТКИ** 8](#_Toc23016852)

[**Додаток А (код програми для алгоритму)** 8](#_Toc23016853)

[**Додаток Б (код програми для тестування)** 9](#_Toc23016854)

[**Додаток В (знімки екрану результату)** 10](#_Toc23016855)

[**Додаток Д (розв’язання сторонніми джерелами)** 10](#_Toc23016856)

# **1 ВСТУП**

Мета лабораторної роботи:

Розробити програмне забезпечення для методу Рунге-Кутти четвертого порядку, аналіз задачі, що розв’язується, та методу її розв’язання на предмет виключних ситуацій, які можуть виникати під час застосування заданого методу до розв’язання поставленої задачі. Для коректно введених даних необхідно обчислити функцію y(x) на відрізку [a; b] з заданою точністю i перевірити правильність результату за допомогою інших засобів.

Даний метод є актуальним і часто використовується для розв’язання диференціальних рівнянь. Актуальність тестування даного методу полягає в тому, що цей метод потребує послідовних обчислень, в яких може виникати помилка. Тому при розв’язанні задач даним метод, потрібно завжди перевіряти його на правильність виконання.

# **2 ОСНОВНА ЧАСТИНА**

# **2.1 Завдання**

Обчислити систему методом Рунге-Кутта 4 порядку:

A = 1, B = 3

Результат обчислення наведено в Додатку **Д** (розв’язання сторонніми джерелами)

# **2.2 Описання методу**

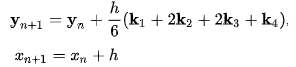
Методи Рунге-Кутти – важлива група [чисельних методів](https://uk.wikipedia.org/wiki/%D0%A7%D0%B8%D1%81%D0%B5%D0%BB%D1%8C%D0%BD%D1%96_%D0%BC%D0%B5%D1%82%D0%BE%D0%B4%D0%B8) розв’язування (систем) звичайних [диференціальних рівнянь](https://uk.wikipedia.org/wiki/%D0%94%D0%B8%D1%84%D0%B5%D1%80%D0%B5%D0%BD%D1%86%D1%96%D0%B0%D0%BB%D1%8C%D0%BD%D1%96_%D1%80%D1%96%D0%B2%D0%BD%D1%8F%D0%BD%D0%BD%D1%8F). Названі на честь німецьких математиків [Карла Рунге](https://uk.wikipedia.org/wiki/%D0%9A%D0%B0%D1%80%D0%BB_%D0%A0%D1%83%D0%BD%D0%B3%D0%B5) і [Мартіна Кутти](https://uk.wikipedia.org/w/index.php?title=%D0%9C%D0%B0%D1%80%D1%82%D1%96%D0%BD_%D0%9A%D1%83%D1%82%D1%82%D0%B0&action=edit&redlink=1), які відкрили ці методи. Це найбільш поширений метод розв'язання задачі Коші.

Метод Рунге — Кутти 4-го порядку настільки широко розповсюджений, що його часто називають просто методом Рунге — Кутти або RK4.

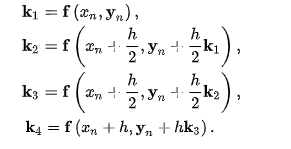
Розглянемо [задачу Коші](https://uk.wikipedia.org/wiki/%D0%97%D0%B0%D0%B4%D0%B0%D1%87%D0%B0_%D0%9A%D0%BE%D1%88%D1%96) для системи диференціальних рівнянь довільного порядку, що записується у векторній формі як



Тоді значення невідомої функції в точці {\displaystyle \ x\_{n+1}}xn+1 обчислюється відносно значення в попередній точці {\displaystyle \ x\_{n}}xn за формулою:



де {\displaystyle \ h}h — крок інтегрування, а коефіцієнти {\displaystyle {\textbf {k}}\_{n}}kn розраховуються таким чином:



Це метод 4-го порядку, тобто похибка на кожному кроці становить {\displaystyle \ O(h^{5})}O(h5), а сумарна похибка на кінцевому інтервалі інтегрування є величиною {\displaystyle \ O(h^{4})}O(h4).

# **2.3 Використані тест кейси**

Для заданого алгоритму було розроблено 6 перевірки на вірність вводу даних. Для перевірки подається 5 матриці, що ілюструють можливі помилки при введенні.

Перевірки на некоректність введених даних:

1. Виконується перевірка на відповідність типів даних.
2. Виконується перевірка на те, чи коректно введена функція.
3. Виконується перевірка на те, чи введені підходящі дані.
4. Виконується перевірка на те, шо a и b > 0.

Перевірки для коректно введених даних:

1. Для лінійної функції метод дає точній результат.  
   Наприклад:
2. Для заданої задачі:

# **3 ВИСНОВКИ**

В даній лабораторній роботі було розроблено програмне забезпечення для чисельного обчислення системи диференціальних рівнянь методом Рунге-Кутти четвертого порядку. Також було розроблено ряд тестів, які перевіряють наявність різних помилок у введених даних, перевіряють чи є функція коректною, а також коректність знайдених коренів для з відповідною точністю 0.00001. У випадку наявності помилок у вхідних даних виводиться відповідне повідомлення. У випадку неіснуючої функції система виводиться повідомлення, що неправильна вхідна функція. Також були додані коректні вхідні дані для них обчислені значення коренів і проведено порівняння цих коренів з результатами одержаними в [https://www.wolframalpha.com](https://www.wolframalpha.com/).

# **ПЕРЕЛІК ПОСИЛАНЬ**

* + - 1. <https://uk.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%A0%D1%83%D0%BD%D0%B3%D0%B5_%E2%80%94_%D0%9A%D1%83%D1%82%D1%82%D0%B8>
      2. <https://mipt.ru/education/chair/computational_mathematics/upload/9d5/rk.pdf>

# **ДОДАТКИ**

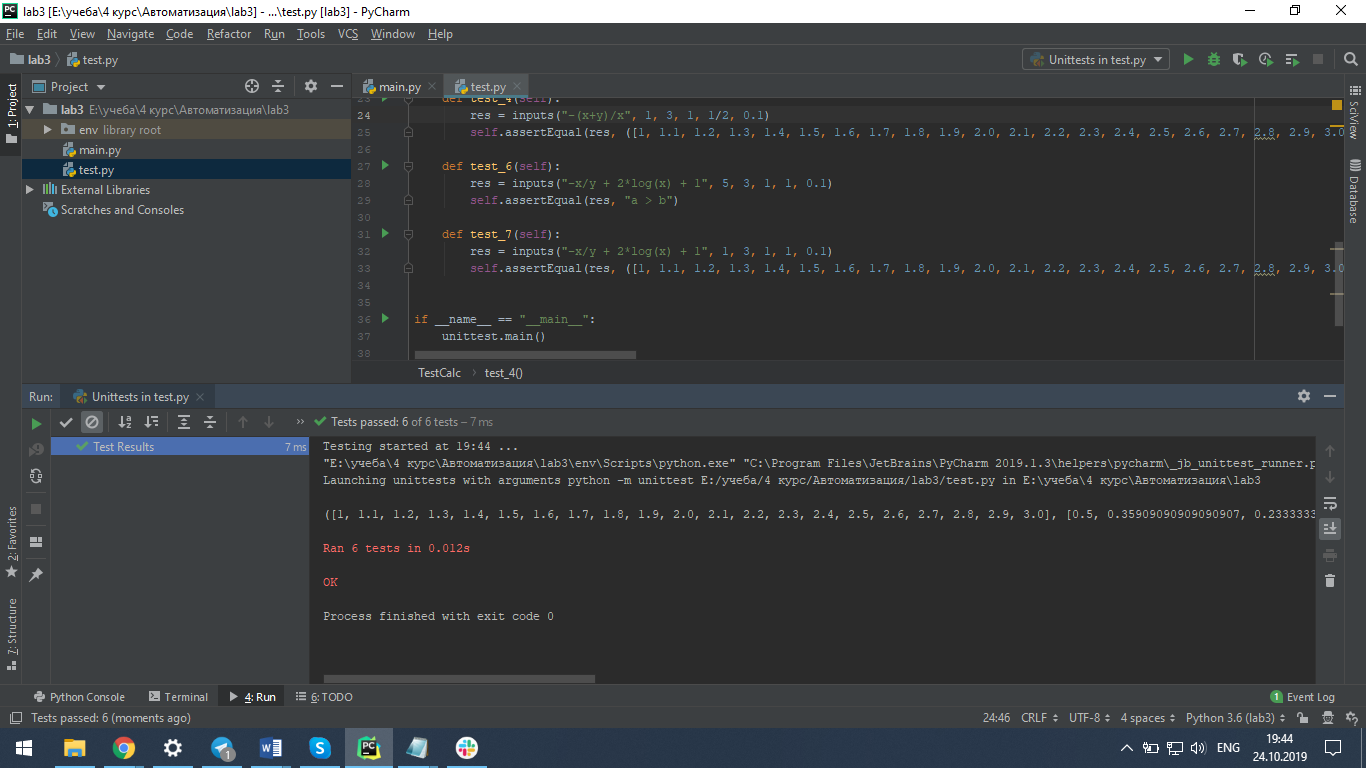
# **Додаток А (код програми для алгоритму)**

from math import exp, log, sin, cos, tan  
  
class Diapason(Exception):  
 def \_\_init\_\_(self, text):  
 self.text = text  
  
  
def inputs(f, a, b, x, y, h):  
 try:  
 if a > b:  
 raise Diapason("a > b")  
 x\_l = [x]  
 y\_l = [y]  
 i = 0  
 while x\_l[i] < b:  
 y\_l.append(y\_l[i] + trapezoidal(f, x\_l[i], y\_l[i], h))  
 x\_l.append(round(x\_l[i] + h, 5))  
 i+=1  
 res = x\_l ,y\_l  
 except NameError:  
 res = "Неправильна вхідна функція"  
 except Diapason as e:  
 return str(e)  
  
 return res  
  
  
def trapezoidal(f, x, y, h):  
 k = []  
 k.append(func(x, y, f))  
 k.append(func(x + h/2, y + h\*k[0]/2, f))  
 k.append(func(x + h / 2, y + h \* k[1] / 2, f))  
 k.append(func(x + h, y + h \* k[2], f))  
  
 result = h/6 \* (k[0] + 2\*k[1] + 2\*k[2] + k[3])  
 return result  
  
  
def func(x, y, func):  
 dd = eval(func)  
 return dd  
  
  
  
res = inputs("-(x+y)/x", 1, 3, 1, 1/2, 0.1)  
  
print(res)

# **Додаток Б (код програми для тестування)**

from main import inputs  
import unittest  
from math import exp, sqrt  
  
e = 0.00001  
  
  
class TestCalc(unittest.TestCase):  
 def test\_1(self):  
 res = inputs("2", 3, 4, 3, 5, 0.1)  
 for i in range(len(res[0])):  
 self.assertTrue(abs(res[1][i] - (2\*res[0][i] - 1)) < e)  
  
 def test\_2(self):  
 res = inputs("-x/y", 0, 1, 0, 2, 0.1)  
 for i in range(len(res[0])):  
 self.assertTrue(abs(res[1][i] - (sqrt(4-res[0][i]\*\*2))) < e)  
  
 def test\_3(self):  
 res = inputs("-x/y + 2\*lor(x) + 1", 1, 3, 1, 1, 0.1)  
 self.assertEqual(res, "Неправильна вхідна функція")  
  
 def test\_4(self):  
 res = inputs("-(x+y)/x", 1, 3, 1, 1/2, 0.1)  
 self.assertEqual(res, ([1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0], [0.5, 0.35909090909090907, 0.2333333333333333, 0.1192307692307692, 0.014285714285714277, -0.08333333333333336, -0.17500000000000002, -0.26176470588235295, -0.34444444444444444, -0.4236842105263158, -0.5, -0.5738095238095238, -0.6454545454545454, -0.7152173913043478, -0.7833333333333333, -0.85, -0.9153846153846154, -0.9796296296296296, -1.042857142857143, -1.1051724137931036, -1.1666666666666667]))  
  
 def test\_6(self):  
 res = inputs("-x/y + 2\*log(x) + 1", 5, 3, 1, 1, 0.1)  
 self.assertEqual(res, "a > b")  
  
 def test\_7(self):  
 res = inputs("-x/y + 2\*log(x) + 1", 1, 3, 1, 1, 0.1)  
 self.assertEqual(res, ([1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0], [1, 1.0048567711392322, 1.019027360425735, 1.0422423991076557, 1.0745891546604185, 1.1164176827487045, 1.1682544866049964, 1.2307201796473815, 1.304452301085999, 1.3900375929345512, 1.4879589748246138, 1.5985610609871461, 1.7220351442129154, 1.8584214330013147, 2.0076241171832523, 2.1694340543748374, 2.3435543428600427, 2.529625240509925, 2.727246255127115, 2.9359944054001117, 3.1554384846421533]))  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 unittest.main()

# **Додаток В (знімки екрану результату)**



# **Додаток Д (розв’язання сторонніми джерелами)**

