НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра прикладної математики

Звіт

із лабораторної роботи

з дисципліни «АЛГОРИТМИ І СИСТЕМИ КОМП’ЮТЕРНОЇ

МАТЕМАТИКИ 1.МАТЕМАТИЧНІ АЛГОРИТМИ»

на тему

“Розв’язання звичайних диференціальних рівнянь”

|  |  |
| --- | --- |
| Виконав: | Перевірила: |
| студент групи КМ-01 | Асистент кафедри ПМА |
| Романецький М.С. | Ковальчук-Химюк Л. О. |

Київ — 2023

Зміст

[Вступ 3](#_Toc149762343)

[Основна частина 4](#_Toc149762344)

[Варіант 1 4](#_Toc149762345)

[Вимоги до ПЗ 4](#_Toc149762346)

[Теоретичні відомості 4](#_Toc149762347)

[Алгоритми розв’язання 5](#_Toc149762348)

[Висновки 6](#_Toc149762349)

[Відповіді на контрольні питання 7](#_Toc149762350)

[Перелік посилань 8](#_Toc149762351)

[Додаток А – Скріншоти роботи програми 9](#_Toc149762352)

[Додаток В – Код програми 11](#_Toc149762353)

# Вступ

Метою даної роботи є вивчення правил використання програмних засобів чисельного розв’язання звичайних диференціальних рівнянь, практичне розв’язання рівнянь на ЕОМ з використанням СКМ та порівняльний аналіз методів інтегрування ОДУ.

# Основна частина

## Варіант 1

|  |  |  |  |
| --- | --- | --- | --- |
| Диференційне рівняння | Початкова умова | Проміжок інтегрування | Крок h |
|  |  | [2; 4] | 0.2 |

Метод розв’язання – метод Рунге-Кутта 4-го порядку

## Вимоги до ПЗ

1. Реалізувати перевірки на некоректний ввід (порожнє введення, символи замість числа, тощо).
2. У програмі повинно бути передбачено можливість гнучкого налагодження розмірності розв’язуваної задачі.
3. Має зберігатись логічно правильне розв’язання (нижня межа інтегрування не має перевищувати верхню, тощо)

## Теоретичні відомості

Суть методу Рунге-Кутта в покроковому обчисленні значень рішення у=у(х) диференціального рівняння виду у = f(x, y) з початковою умовою (х0, у0). Метод Рунге-Кутта 4-го порядку є методом 4-го порядку точності. При використанні формул методу Рунге-Кутта виникають питання: якими з формул доцільно користуватись у кожному конкретному випадку, як вибирати крок сітки? Якщо права частина диференційного рівняння - функція, неперервна й обмежена разом із своїми четвертими похідними, тоді добрі результати дає метод Рунге-Кутта четвертого порядку завдяки швидкому зростанню точності зі зменшенням кроку сітки. Якщо права частина не має обмежених четвертих похідних, тоді максимального порядку точності такої схеми не можна досягнути. В цьому випадку доцільно використовувати обчислювальні схеми методу меншого порядку точності, який відповідає порядку похідних. Крок сітки вибирають настільки малим, щоб забезпечити необхідну точність розрахунків. Основним практичним способом одержання заданої точності є апостеріорна оцінка похибки. Для її відшукання розрахунок проводять на

двох сітках з кроками та застосовують правило Рунге.

Формула методу Рунге-Кутта 4-го порядку:

## Алгоритми розв’язання

Імпортуємо бібліотеки Oct2py [1], pandas [2], matplotlib [3]

Створюємо власну функцію, яка приймає дані і рахує за формулою Рунге-Кутта 4-го порядку. У цю функцію передаємо дані індивідуального варіанту, а саме, початкові умови, проміжок інтегрування та крок. Оскільки задане диференційне рівняння є рівнянням другого порядку, то треба ввести заміну

Цей алгоритм розв’язання реалізований на мові Python та на мові Octave.

У кінці ми виводимо 3 графіки:

1. Розв’язки мовою Python
2. Розв’язки мовою Octave
3. Розв’язки обидвома мовами, для порівняння кривих

# Висновки

У результаті виконання роботи досягнуто таких результатів:

Розв'язок Python:

w(x) y(x)

x

2.0 -2.000000 1.000000

2.2 -3.175600 0.488600

2.4 -4.773720 -0.298266

2.6 -6.923826 -1.457506

2.8 -9.792000 -3.115438

3.0 -13.590811 -5.436057

3.2 -18.591750 -8.631535

3.4 -25.140913 -12.975505

3.6 -33.678714 -18.819886

3.8 -44.764698 -26.616125

4.0 -59.108694 -36.942026

Розв'язок Octave:

w(x) y(x)

x

2.0 -2.000000 1.000000

2.2 -3.175600 0.488600

2.4 -4.773720 -0.298266

2.6 -6.923826 -1.457506

2.8 -9.792000 -3.115438

3.0 -13.590811 -5.436057

3.2 -18.591750 -8.631535

3.4 -25.140913 -12.975505

3.6 -33.678714 -18.819886

3.8 -44.764698 -26.616125

4.0 -59.108694 -36.942026

Як бачимо, результати є ідентичними, щоб переконатись у цьому не занурюючись у порівняння чисел можна подивитись на графіки розв’язків у Додатку А.

# Відповіді на контрольні питання

1. **Що значить вирішити диференціальні рівняння чисельним методом?** Вирішення диференціальних рівнянь чисельним методом означає знаходження наближеного чисельного розв'язку диференціального рівняння за допомогою обчислювальних методів. Це необхідно, коли аналітичне розв'язання диференціального рівняння неможливе або складне, і ми використовуємо чисельні методи для наближеного знаходження функції, яка задовольняє рівняння.
2. **Охарактеризуйте метод Рунге-Кутта четвертого порядку –** Метод Рунге-Кутта четвертого порядку - це чисельний метод для розв'язання звичайних диференціальних рівнянь. Він є одним з найпоширеніших методів і має четвертий порядок точності. Метод використовує ітерації для оновлення значень функції на кожному кроці, використовуючи середні значення на кількох під кроках для покращення точності результатів.
3. **Дайте порівняльну характеристику методів Рунге-Кутта і методів прогнозу-корекції –** Методи Рунге-Кутта та методи прогнозу-корекції - це дві різні класи чисельних методів для розв'язання диференціальних рівнянь. Методи Рунге-Кутта використовують ітерації та середні значення для оновлення функцій на кожному кроці, тоді як методи прогнозу-корекції використовують два окремі кроки: прогноз та корекцію. Обидва підходи мають свої переваги та недоліки, і вибір залежить від конкретного завдання та вимог до точності.
4. **Опишіть процедуру автоматичного обрання кроку в методі Рунге-Кутта –** Процедура автоматичного обрання кроку в методі Рунге-Кутта називається "кроком адаптації". Вона полягає в тому, щоб динамічно змінювати розмір кроку в залежності від поточних умов розв'язку. На кожному кроці обчислюється деякий показник точності, і наступний крок вибирається так, щоб досягнути заданого рівня точності. Це дозволяє ефективно розв'язувати різні диференціальні рівняння з різними характерами розв'язків.
5. **Як оцінюють точність отриманого розв'язку в методах прогнозу і корекції?** Оцінка точності отриманого розв'язку в методах прогнозу і корекції відбувається шляхом порівняння результату прогнозу (наближеного розв'язку) з результатом корекції (кінцевим розв'язком). Точність оцінюється за допомогою різниці між цими двома значеннями, і зазвичай використовується показник помилки, такий як середньоквадратична помилка або абсолютна помилка. Мета полягає в тому, щоб зробити прогнози та корекції настільки точними, наскільки це можливо, і зменшити вплив наближень на кінцевий результат.

# Перелік посилань

1. Oct2py - <https://pypi.org/project/oct2py/>
2. Pandas - <https://pandas.pydata.org>
3. Matplotlib - <https://matplotlib.org>

# Додаток А – Скріншоти роботи програми

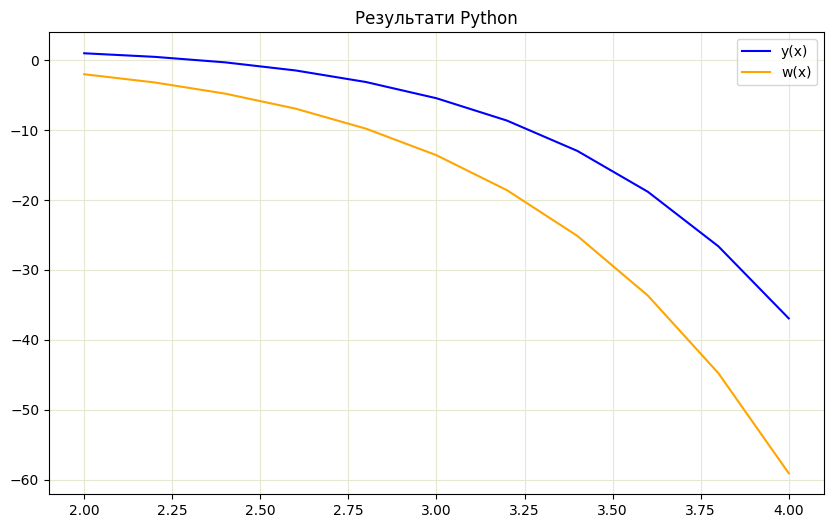


Рис. 1 – Графічне представлення результатів отриманих мовою Python

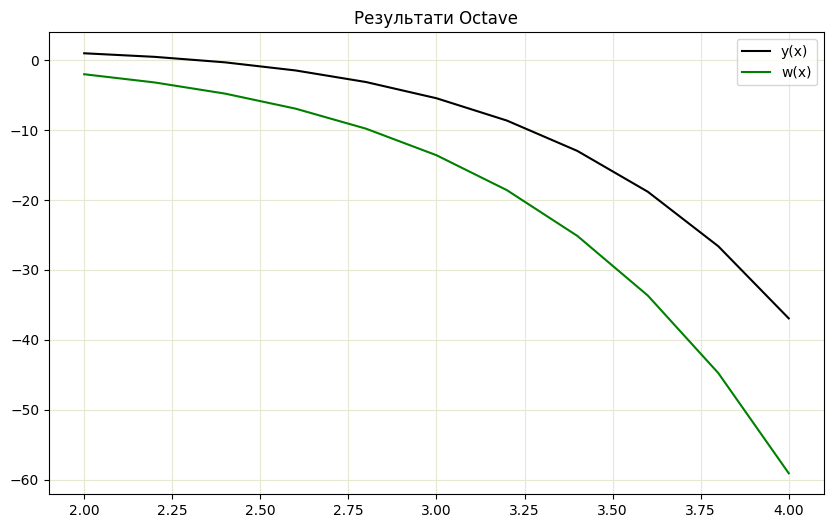


Рис. 2 – Графічне представлення результатів отриманих мовою Octave

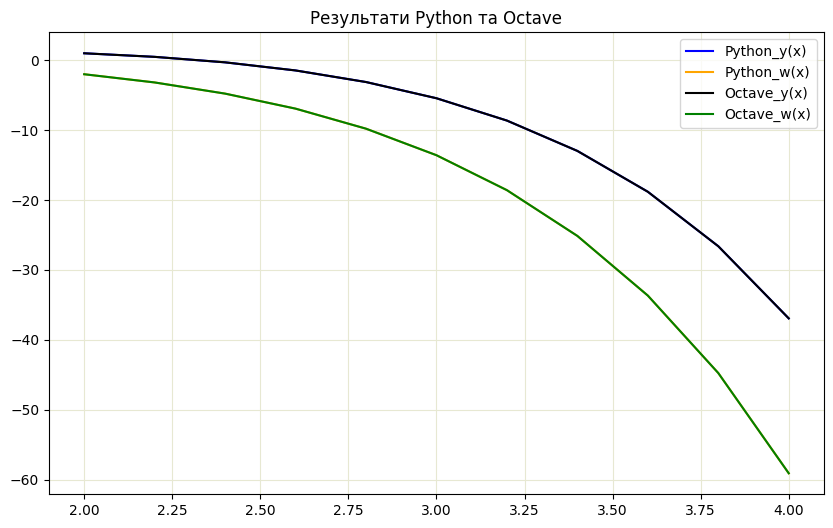


Рис. 3 – Графічне представлення результатів отриманих обидвома мовами програмування

# Додаток В – Код програми

Вміст файлу main.ipynb :

from oct2py import octave

import pandas as pd

import matplotlib.pyplot as plt

def runge\_kutta(x0, y0, z0, h, x\_target):

    x = x0

    y = y0

    z = z0

    results = [('x', 'w(x)', 'y(x)')]

    while x <= x\_target:

        results.append((x, z, y))

        k1y = h \* z

        k1z = h \* (2 \* z - y)

        k2y = h \* (z + 0.5 \* k1z)

        k2z = h \* (2 \* (z + 0.5 \* k1z) - (y + 0.5 \* k1y))

        k3y = h \* (z + 0.5 \* k2z)

        k3z = h \* (2 \* (z + 0.5 \* k2z) - (y + 0.5 \* k2y))

        k4y = h \* (z + k3z)

        k4z = h \* (2 \* (z + k3z) - (y + k3y))

        y = y + (1/6) \* (k1y + 2 \* k2y + 2 \* k3y + k4y)

        z = z + (1/6) \* (k1z + 2 \* k2z + 2 \* k3z + k4z)

        x = x + h

    return results

x0 = 2

y0 = 1

z0 = -2

h = 0.2

x\_target = 4

results = runge\_kutta(x0, y0, z0, h, x\_target+h)

*# Створюємо датафрейм з результатами*

df\_python = pd.DataFrame(results[1:], columns=results[0]).set\_index('x')

print('Розв\'язок Python:\n')

print(df\_python)

*# Визначаємо функцію Runge-Kutta в Octave*

octave.eval("""

function results = runge\_kutta\_oct(x0, y0, z0, h, x\_target)

    x = x0;

    y = y0;

    z = z0;

    results = [x, z, y];

    while x <= x\_target

        k1y = h \* z;

        k1z = h \* (2 \* z - y);

        k2y = h \* (z + 0.5 \* k1z);

        k2z = h \* (2 \* (z + 0.5 \* k1z) - (y + 0.5 \* k1y));

        k3y = h \* (z + 0.5 \* k2z);

        k3z = h \* (2 \* (z + 0.5 \* k2z) - (y + 0.5 \* k2y));

        k4y = h \* (z + k3z);

        k4z = h \* (2 \* (z + k3z) - (y + k3y));

        y = y + (1/6) \* (k1y + 2 \* k2y + 2 \* k3y + k4y);

        z = z + (1/6) \* (k1z + 2 \* k2z + 2 \* k3z + k4z);

        x = x + h;

        results = [results; x, z, y];

    end

end

""")

x0 = 2

y0 = 1

z0 = -2

h = 0.2

x\_target = 4

*# Викликаємо функцію Runge-Kutta в Octave*

results\_octave = octave.runge\_kutta\_oct(x0, y0, z0, h, x\_target)

*# Створюємо датафрейм з результатами*

df\_octave = pd.DataFrame(results\_octave, columns=['x', 'w(x)', 'y(x)']).set\_index('x')

print('Розв\'язок Octave:\n')

print(df\_octave)

*# Створюємо графік для результатів Python*

plt.figure(figsize=(10, 6))

plt.plot(df\_python.index, df\_python['y(x)'], label='y(x)', color='blue')

plt.plot(df\_python.index, df\_python['w(x)'], label='w(x)', color='orange')

plt.title('Результати Python')

plt.legend()

plt.grid(c='#E7E8D2')

plt.show()

*# Створюємо графік для результатів Octave*

plt.figure(figsize=(10, 6))

plt.plot(df\_octave.index, df\_octave['y(x)'], label='y(x)', color='black')

plt.plot(df\_octave.index, df\_octave['w(x)'], label='w(x)', color='green')

plt.title('Результати Octave')

plt.legend()

plt.grid(c='#E7E8D2')

plt.show()

*# Об'єднуємо результати на одному графіку*

print('Як бачимо, результати спіпадають')

plt.figure(figsize=(10, 6))

plt.plot(df\_python.index, df\_python['y(x)'], label='Python\_y(x)', color='blue')

plt.plot(df\_python.index, df\_python['w(x)'], label='Python\_w(x)', color='orange')

plt.plot(df\_octave.index, df\_octave['y(x)'], label='Octave\_у(x)', color='black')

plt.plot(df\_octave.index, df\_octave['w(x)'], label='Octave\_w(x)', color='green')

plt.title('Результати Python та Octave')

plt.legend()

plt.grid(c='#E7E8D2')

plt.show()