**Лабораторна робота 7**

**Тема: Числове диференціювання**

**Завдання:** за допомогою інтерполяційних формул Ньютона з точністю до 0.001 знайти значення першої та другої похідних за даних значень аргумента для функції , що задана таблицею:

Таблиця 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 2.4 | 2.6 | 2.8 | 3.0 | 3.2 | 3.4 |
|  | 3.526 | 3.782 | 3.945 | 4.043 | 4.104 | 4.155 |
|  | 3.6 | 3.8 | 4.0 | 4.2 | 4.4 | 4.6 |
|  | 4.222 | 4.331 | 4.507 | 4.775 | 5.159 | 5.683 |

Для***n*=1, 3, 5, 7, …, 25**: *x* =2.4.

Таблиця 2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1.5 | 2.0 | 2.5 | 3.0 | 3.5 | 4.0 |
|  | 10.517 | 10.193 | 9.807 | 9.387 | 8.977 | 8.637 |
|  | 4.5 | 5.0 | 5.5 | 6.0 | 6.5 | 7.0 |
|  | 8.442 | 8.482 | 8.862 | 9.701 | 11.132 | 13.302 |

Для***n*=2, 4, 6, 8, …, 24**: *x=1.5.*

**Теоретичні відомості**

Один із способів розв’язання задачі диференціювання – це використання інтерполяційних багаточленів.

Для виведення формул наближеного диференціювання дану функцію  на відрізку замінюють інтерполяційною функцією та покладають, що

 за .

Аналогічно поступають для знаходження значення похідних функції  вищих порядків.

Розглянемо числове диференціювання на основі *інтерполяційної формули Ньютона*, тобто покладемо, що функція  задана у вигляді таблиці з постійним кроком.

Запишемо для функції  перший інтерполяційний багаточлен Ньютона:

.

Перепишемо, розкриваючи дужки:

.

Враховуючи правило диференціювання складної функції



матимемо:

. (1)

Аналогічно, враховуючи, що

,

отримуємо:

. (2).

У такий самий спосіб за необхідністю можна обрахувати похідні функції  будь-якого порядку.

Проте кожного разу, обчислюючи значення похідної у фіксованій точці *х* за формулами (1) та (2), в якості  слід обирати найближче зліва вузлове значення аргумента.

Формули (1) та (2) значно спрощуються, якщо шуканим значенням  виявляється один з вузлів таблиці. Оскільки в цьому випадку кожне табличне значення можна вважати за початкове, то, покладаючи, отримуємо  Тоді формули матимуть вигляд:

,

.

Аналогічно виводяться формули для числового диференціювання на основі другої інтерполяційної формули Ньютона.

Запишемо для функції  другий інтерполяційний багаточлен Ньютона:

Перепишемо цей поліном, розкриваючи дужки:

Враховуючи правило диференціювання складної функції матимемо:

. (3)

. (4)

У вузлових точках формули (3) та (4) матимуть вигляд:

,

.

**Зразок виконання завдання**

**Завдання:** за допомогою інтерполяційних формул Ньютона з точністю до 0.0001 знайти значення першої та другої похідних для функції , що задана таблицею, в точці.

*Розв’язання:*

Складемо для заданої функції таблицю кінцевих різниць:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| 0 | 0 | 1.2733 | 0.5274 | 0.0325 | 0.0047 | 0.0002 | 0 |
| 1 | 0.1 | 1.8007 | 0.5599 | 0.0372 | 0.0049 | 0.0002 |  |
| 2 | 0.2 | 2.3606 | 0.5971 | 0.0421 | 0.0051 |  |  |
| 3 | 0.3 | 2.9577 | 0.6392 | 0.0472 |  |  |  |
| 4 | 0.4 | 3.5969 | 0.6864 |  |  |  |  |
| 5 | 0.5 | 4.2833 |  |  |  |  |  |

За умовою крок таблиці . Шукане значення *х* співпадає з вузлом таблиці. Тобто табличне значення  можна вважати за початкове: . Тоді .

Обчислення проводимо за формулами:

,.

Отже



.



.

*Відповідь*: .

**Код**

import numpy as np

import math

mas\_x = [2.4,2.6,2.8,3.0,3.2,3.4]

mas\_y = [3.526,3.782,3.945,4.043,4.104,4.155]

h = mas\_x[1] - mas\_x[0]

print (h)

mas\_1 = []

mas\_2 = []

mas\_3 = []

mas\_4 = []

for i in range(len(mas\_y)):

mas\_1.append(mas\_y[i] - mas\_y[i-1])

mas\_1.pop (0)#вилучаємо нульовий елемент масиву

print('mas\_1 = ', mas\_1)#cкінченні різниці 1-го порядку

for j in range(len(mas\_1)):

mas\_2.append(mas\_1[j] - mas\_1[j-1])

mas\_2.pop (0)

print('mas\_2=',mas\_2 )#cкінченні різниці 2-го порядку

for k in range(len(mas\_2)):

mas\_3.append(mas\_2[k] - mas\_2[k-1])

mas\_3.pop (0)

print('mas\_3=',mas\_3)#cкінченні різниці 3-го порядку

for l in range(len(mas\_3)):

mas\_4.append(mas\_3[l] - mas\_3[l-1])

mas\_4.pop (0)

print('mas\_4=',mas\_4)#cкінченні різниці 4-го порядку

y1 = 1/ h \* (mas\_1[1] - (mas\_2[1]/ 2) + (mas\_3[1] /3) - (mas\_4[1]/4))

y2 = 1/ (h\*\*2) \* (mas\_2[1] - mas\_3[1] + 11/12\*mas\_4[1])

print ('First derivative =', y1)

print ('Second derivative =', y2)

**Результат**

First derivative = 1.0254166666666604

Second derivative = -2.3479166666665954

**Звіт має містити:**

1. ПІП група, варіант
2. Аналітичні обчислення
3. Код+ скрін