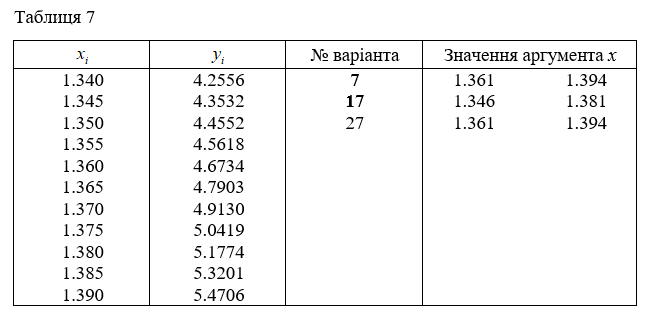
**Лабораторна робота 8 Інтерполяційний багаточлен Ньютона1**

**Завдання 1:** наближено відбудувати функцію , що задана таблицею, у довільній точці *х* за допомогою інтерполяційних багаточленів Ньютона.   
За наявним набором значень побудувати графік інтерполяційної функції.



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| I | xi | yi | 1yi | 2yi | 3yi | 4yi | 5yi | 6yi | 7yi |
| 0 | 1,340 | 4,2556 | 0,0976 | 0,0044 | 0,0002 | 0,0002 | -0,0003 | 0,0004 | -0,0008 |
| 1 | 1,345 | 4,3532 | 0,1020 | 0,0046 | 0,0004 | -0,0001 | 0,0001 | -0,0004 | 0,0006 |
| 2 | 1,350 | 4,4552 | 0,1066 | 0,0050 | 0,0003 | 0,0002 | -0,0003 | 0,0002 | 0,0001 |
| 3 | 1,355 | 4,5618 | 0,1116 | 0,0053 | 0,0005 | -0,0001 | -0,0001 | 0,0003 | -0,0007 |
| 4 | 1,360 | 4,6734 | 0,1169 | 0,0058 | 0,0004 | 0 | 0,0002 | -0,0004 |  |
| 5 | 1,365 | 4,7903 | 0,1227 | 0,0062 | 0,0004 | 0,0002 | -0,0002 |  |  |
| 6 | 1,370 | 4,9130 | 0,1289 | 0,0066 | 0,0006 | 0 |  |  |  |
| 7 | 1,375 | 5,0419 | 0,1355 | 0,0072 | 0,0006 |  |  |  |  |
| 8 | 1,380 | 5,1774 | 0,1427 | 0,0078 |  |  |  |  |  |
| 9 | 1,385 | 5,3201 | 0,1505 |  |  |  |  |  |  |
| 10 | 1,390 | 5,4706 |  |  |  |  |  |  |  |

8yi 9yi 10yi

0,0014 0,0019 -0,0022

-0,0005 -0,0003

-0,0008

За умовою крок таблиці h = 0,005.

Для x = 1,361 за початкове значення беремo x0 = 1,340.

Тоді q = (1,361 - 1,340)/0,005 = 0,021/0,005 = 4,2.

Обчислення проводимо за першою інтерполяційною формулою Ньютона:

F(x) = 4,695088 + 0,00128128 – 0,00002089472 – 0,0000026155008 =

4,797873432

Для x = 1,394 за початкове значення беремo x0 = 1,390.

q = 0,8

f(x) = 5,594451231

Код:

import numpy as np  
from math import factorial  
  
  
x = [1.340, 1.345, 1.350, 1.355, 1.360, 1.365, 1.370, 1.375, 1.380, 1.385, 1.390]  
y = [4.2556, 4.3532, 4.4552, 4.6734, 4.7903, 4.9130, 5.0419, 5.1774, 5.3201, 5.4706]  
h = x[1] - x[0]  
x1 = 1.361  
x2 = 1.394  
q = (x1 - x[0]) / h  
q1 = (x2 - x[-1]) / h  
  
def n(y,j): *#обчислення кінцевих різниць* mas=[]  
 for i in range(len(y)):  
 mas.append(y[i] - y[i-1])  
 mas.pop(0)  
 if j == 1:  
 return mas  
 else:  
 j-=1  
 return n(mas, j)  
  
*#Перша інтерполяційна формула Ньютона*s\_1 = y[0]+q\*n(y,1)[0]+q\*(q-1)\*n(y,2)[0]/factorial(2)  
s\_2 = q\*(q-1)\*(q-2)\*n(y,3)[0]/factorial(3)  
s\_3 = q\*(q-1)\*(q-2)\*(q-3)\*n(y,4)[0]/factorial(4)  
s\_4 = q\*(q-1)\*(q-2)\*(q-3)\*(q-4)\*n(y,5)[0]/factorial(5)  
n\_1 = s\_1 + s\_2 + s\_3 + s\_4  
print (**'Значення x1 ='**, x1, **'по первій інтерполяційній формулі'**, round(n\_1,5))  
  
*#Друга інтерполяційна формула Ньютона*s\_1 = y[-1]+q1\*n(y,1)[-1]+q1\*(q1+1)\*n(y,2)[-0]/factorial(2)  
s\_2 = q1\*(q1+1)\*(q1+2)\*n(y,3)[-1]/factorial(3)  
s\_3 = q1\*(q1+1)\*(q1+2)\*(q1+3)\*n(y,4)[-1]/factorial(4)  
s\_4 = q1\*(q1+1)\*(q1+2)\*(q1+3)\*(q1+4)\*n(y,5)[-1]/factorial(5)  
n\_1 = s\_1 + s\_2 + s\_3 + s\_4  
  
print (**'Значення x2 ='**, x2, **'по другій інтерполяційній формулі'**, round(n\_1, 5))

Результат:



