import numpy as np

from numpy import \*

import math

import matplotlib.pyplot as plt

def fac1(i1, qq1):

j1=1

dq1=1

while j1<=i1:

dq1\*=qq1 - j1

j1+=1

return dq1

def fac2(i2, qq2):

j2=1

dq2=1

while j2<=i2:

dq2\*=qq2 + j2

j2+=1

return dq2

mas\_x=[0.01,0.06,0.11,0.16,0.21,0.26,0.31,0.36,0.41,0.46,0.51]

mas\_y=[0.9918,0.9519, 0.9136, 0.8769, 0.8416, 0.8077,0.7753,0.7441,0.7141, 0.6854, 0.6579]

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

x=0.1

xo=0

xn=0.410

q=(x-xo)/h

k=0

def y(mas\_y,j):

mas=[]

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

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

mas.pop(0)

if j == 1:

return mas

else:

j-=1

return y(mas, j)

yx1=1/h\*((y(mas\_y, 1)[1])-(y(mas\_y, 2)[1])/2+(y(mas\_y, 3)[1])/3-(y(mas\_y, 4)[1])/4)

yx2=1/h\*\*2\*((y(mas\_y, 2)[1])-(y(mas\_y, 3)[1])+11/12\*(y(mas\_y, 4)[1]))

q=(x-xn)/h

Nx1=mas\_y[0]+q\*mas\_y[1]-mas\_y[0]

print ("Напишіть порядок до якого буде працювати підрахунок")

jkl=int(input())-1

while k<jkl:

k+=1

Nx1+=(q\*(fac1(k, q))/math.factorial(k+1))\*(y(mas\_y,k+1)[0])

N1= mas\_y[0] + q\*mas\_y[1]-mas\_y[0]+(q\*(q-1)/math.factorial(2))\*(y(mas\_y, 2)[0]) + (q\*(q-1)\*(q-2)/math.factorial(3))\*(y(mas\_y, 3)[0])

k=0

Nx2=mas\_y[jkl]+q\*mas\_y[1]-mas\_y[0]

while k<jkl:

k+=1

Nx2+=(q\*(fac2(k, q))/math.factorial(k+1))\*(y(mas\_y,k+1)[jkl-k])

print ("1 інтерполяційна формула Ньютона: //", Nx1, " // (через цикл до)", jkl," -----------------------------------------")

print ("1 інтерполяційна формула Ньютона: //", N1," // (через Формулу до 3 порядка) --------------------------------------")

print ("2 інтерполяційна формула Ньютона: //", Nx2, " // (через цикл до)", jkl," -----------------------------------------")

plt.plot(mas\_x, mas\_y, label='Nx1 , Nx2')

plt.title('LB\_9 Omelchuk Dmytro')

plt.legend(loc='upper left')

plt.xlabel('x')

plt.ylabel('y')

plt.grid()

plt.show()