**Лабораторна робота №8**

**Сухина Денис ФІТ 2-6**

**25 варіант**

import numpy as np

from math import factorial

import matplotlib.pyplot as plt

x=[3.60, 3.65, 3.70, 3.75, 3.80, 3.85]

y=[36.59823, 8.47474, 0.44734, 2.52114, 4.70124, 6.9931]

h = x[1] - x[0]

x1=0.1

x2=0.9

q=(x1 - x[0])/h

q1 = (x2-x[-1])/h

def n(y,j):

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)

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 функції x1 = ', x1, 'використовуючи першу інтерполяційну формулу Ньютона',

round(n\_1,5))

k\_1 = y[-1] + q1 \* n(y,1)[-2] + q1 \* (q1 + 1) \* n(y,2)[-3] / factorial(2)

k\_2 = q1 \* (q1 + 1) \* (q1 + 2) \* n(y,3)[-1] / factorial(3)

k\_3 = q1 \* (q1 + 1) \* (q1 + 2) \* (q1 + 3) \* n(y,4)[-1] / factorial(4)

k\_4 = q1 \* (q1 + 1) \* (q1 + 2) \* (q1 + 3) \* (q1 + 4) \* n(y,5)[-1] / factorial(5)

n\_2 = k\_1 + k\_2 + k\_3 + k\_4

print ('Значення функції x2 = ', x2, 'використовуючи другу інтерполяційну формулу Ньютона',

round(n\_2,5))

x\_1 = np.linspace(np.min(x), np.max(x))

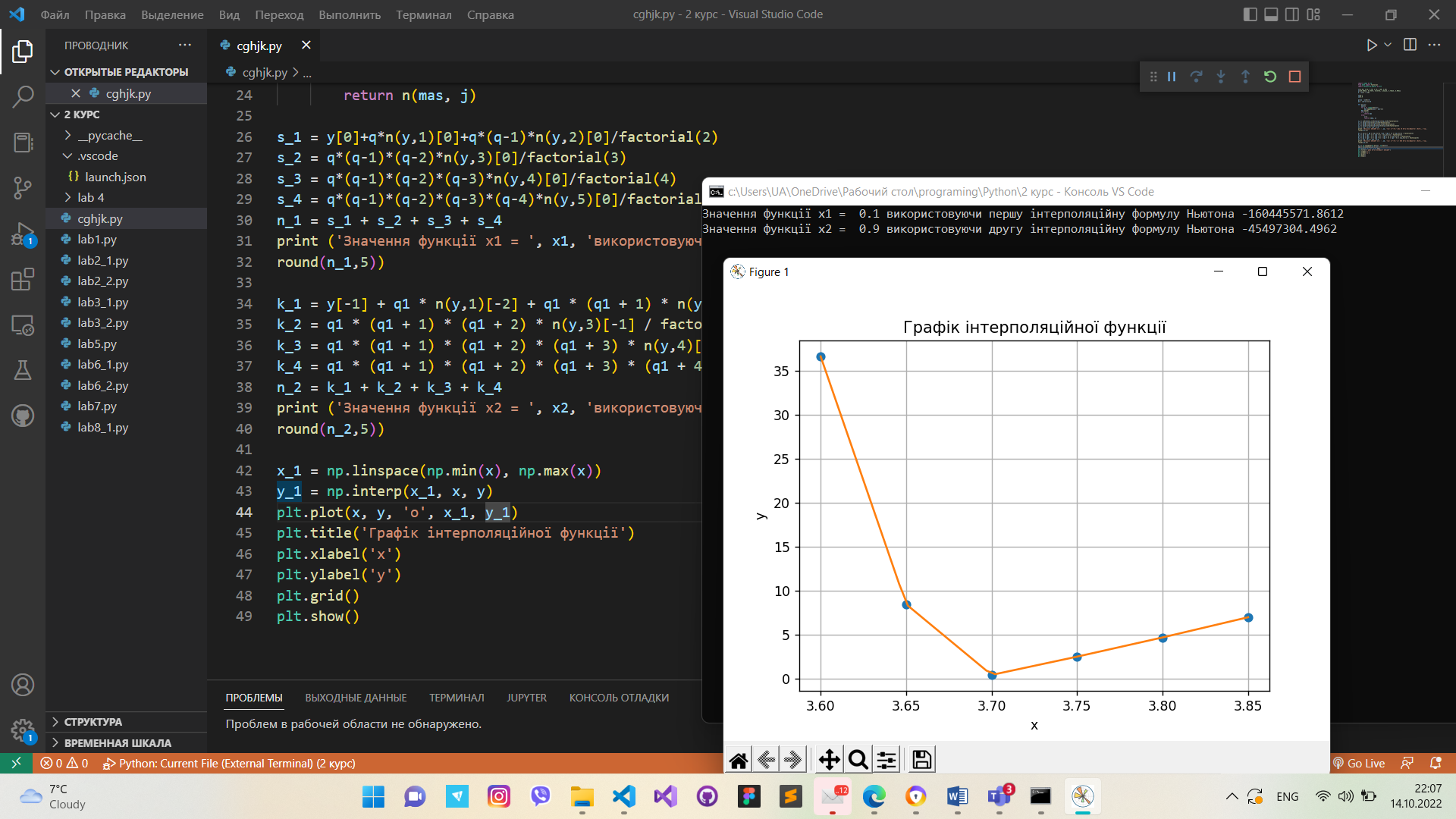
y\_1 = np.interp(x\_1, x, y)

plt.plot(x, y, 'o', x\_1, y\_1)

plt.title('Графік інтерполяційної функції')

plt.xlabel('x')

plt.ylabel('y')

plt.grid()

plt.show()