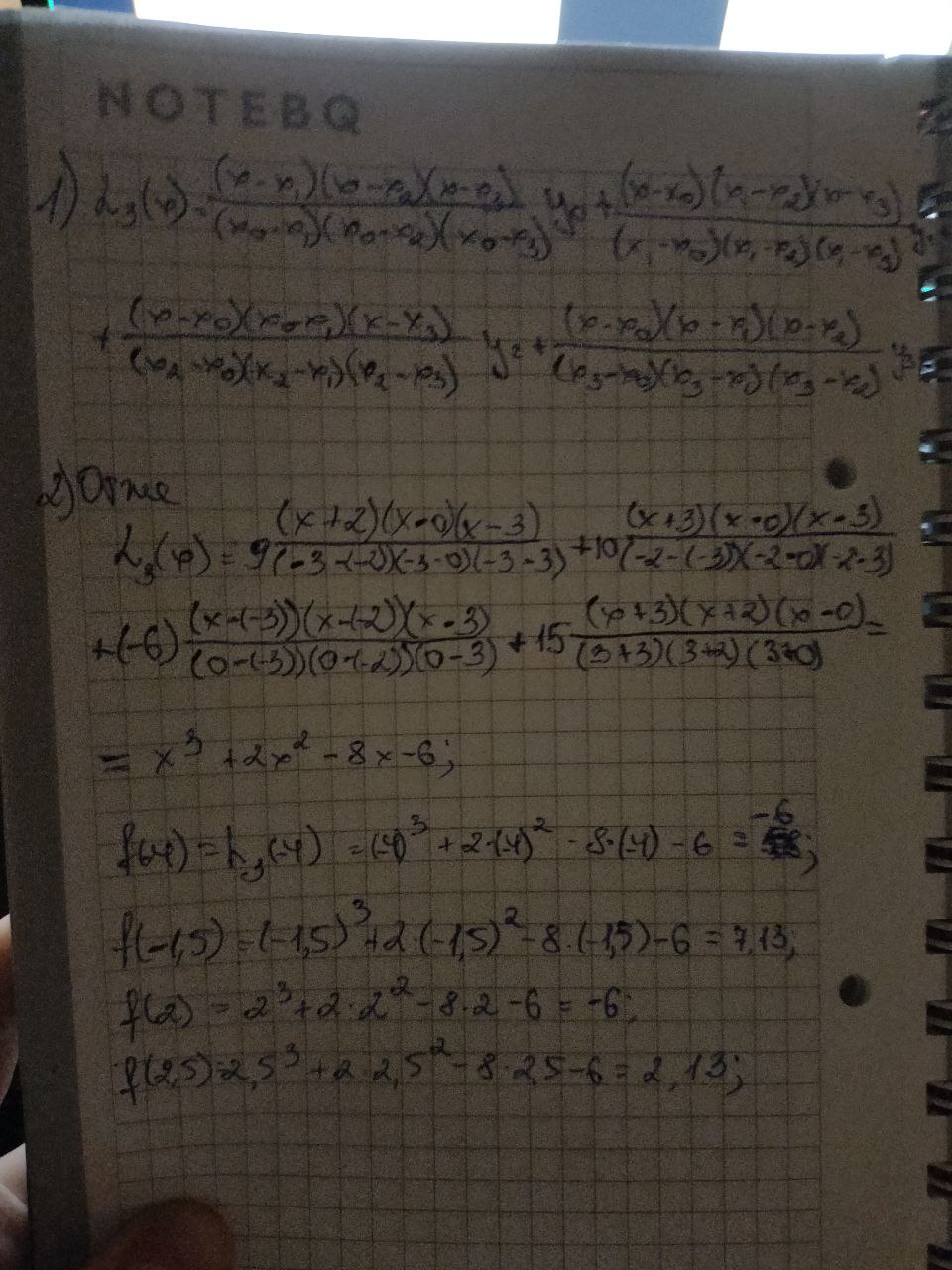
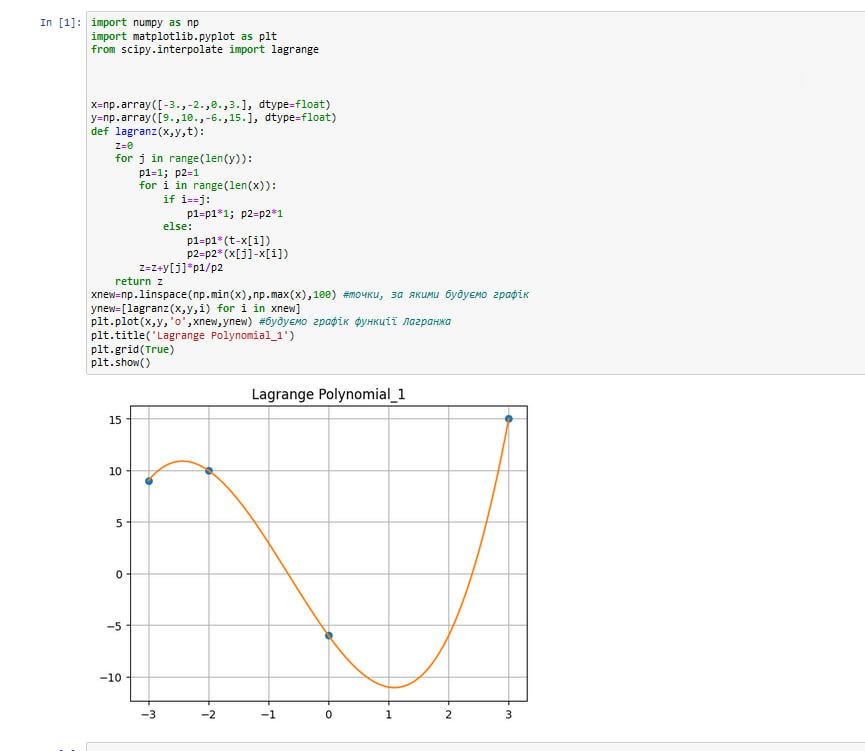
жддДЖПальонка Анастасія 2-8,Варіант 20

**Завдання:** Побудувати інтерполяційний багаточлен Лагранжа  для функції , що задана таблицею, та з точністю до 0.001 обчислити наближені значення функції у заданих точках. Побудувати графік інтерполяційної функції  за наявним набором точок.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **№20** |  | -3 | -2 | 0 | 3 | -4 | -1,5 | 2 | 2,5 |
|  | 9 | 10 | -6 | 15 | ? | ? | ? | ? |

|  |  |  |  |
| --- | --- | --- | --- |
| -4 | -1,5 | 2 | 2,5 |
| -6 | 7,13 | -6 | 2,13 |





import numpy as np

import matplotlib.pyplot as plt

from scipy.interpolate import lagrange

x=np.array([-3.,-2.,0.,3.], dtype=float)

y=np.array([9.,10.,-6.,15.], dtype=float)

def lagranz(x,y,t):

z=0

for j in range(len(y)):

p1=1; p2=1

for i in range(len(x)):

if i==j:

p1=p1\*1; p2=p2\*1

else:

p1=p1\*(t-x[i])

p2=p2\*(x[j]-x[i])

z=z+y[j]\*p1/p2

return z

xnew=np.linspace(np.min(x),np.max(x),100) #точки, за якими будуємо графік

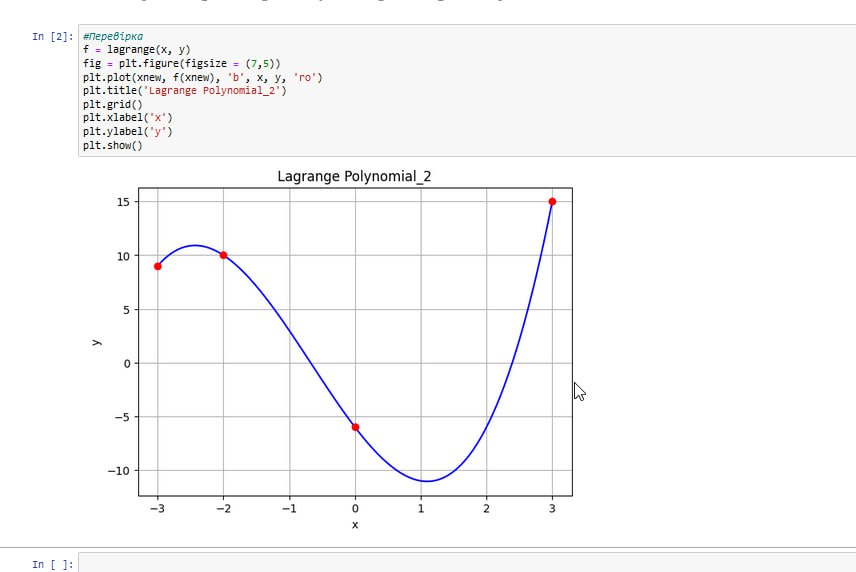
ynew=[lagranz(x,y,i) for i in xnew]

plt.plot(x,y,'o',xnew,ynew) #будуємо графік функції Лагранжа

plt.title('Lagrange Polynomial\_1')

plt.grid(True)

plt.show()



#Перевірка

f = lagrange(x, y)

fig = plt.figure(figsize = (7,5))

plt.plot(xnew, f(xnew), 'b', x, y, 'ro')

plt.title('Lagrange Polynomial\_2')

plt.grid()

plt.xlabel('x')

plt.ylabel('y')

plt.show()