Завдання 1

import numpy as np

import matplotlib.pyplot as plt

from scipy.interpolate import lagrange

x=np.array([-1,0,1,2])

y=np.array([-20,-5,6,25])

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

graphX=np.linspace(np.min(x),np.max(x),100)

graphY=[lagranz(x,y,i) for i in graphX]

plt.plot(x,y,'o',graphX,graphY)

plt.show()

poly = lagrange(x, y)

print(poly)

x1 = -0.5

yp = 0

n=3

for i in range(n):

p = 1

for j in range(n):

if i != j:

p = p \* (x1 - x[j])/(x[i] - x[j])

yp = yp + p \* y[i]

print('Interpolated value at ',x1, yp)

x2 = 0.5

for i in range(n):

p = 1

for j in range(n):

if i != j:

p = p \* (x2 - x[j])/(x[i] - x[j])

yp = yp + p \* y[i]

print('Interpolated value at ',x2, yp)

x3 = 1.5

for i in range(n):

p = 1

for j in range(n):

if i != j:

p = p \* (x3 - x[j])/(x[i] - x[j])

yp = yp + p \* y[i]

print('Interpolated value at ',x3, yp)

x4 = 2.5

for i in range(n):

p = 1

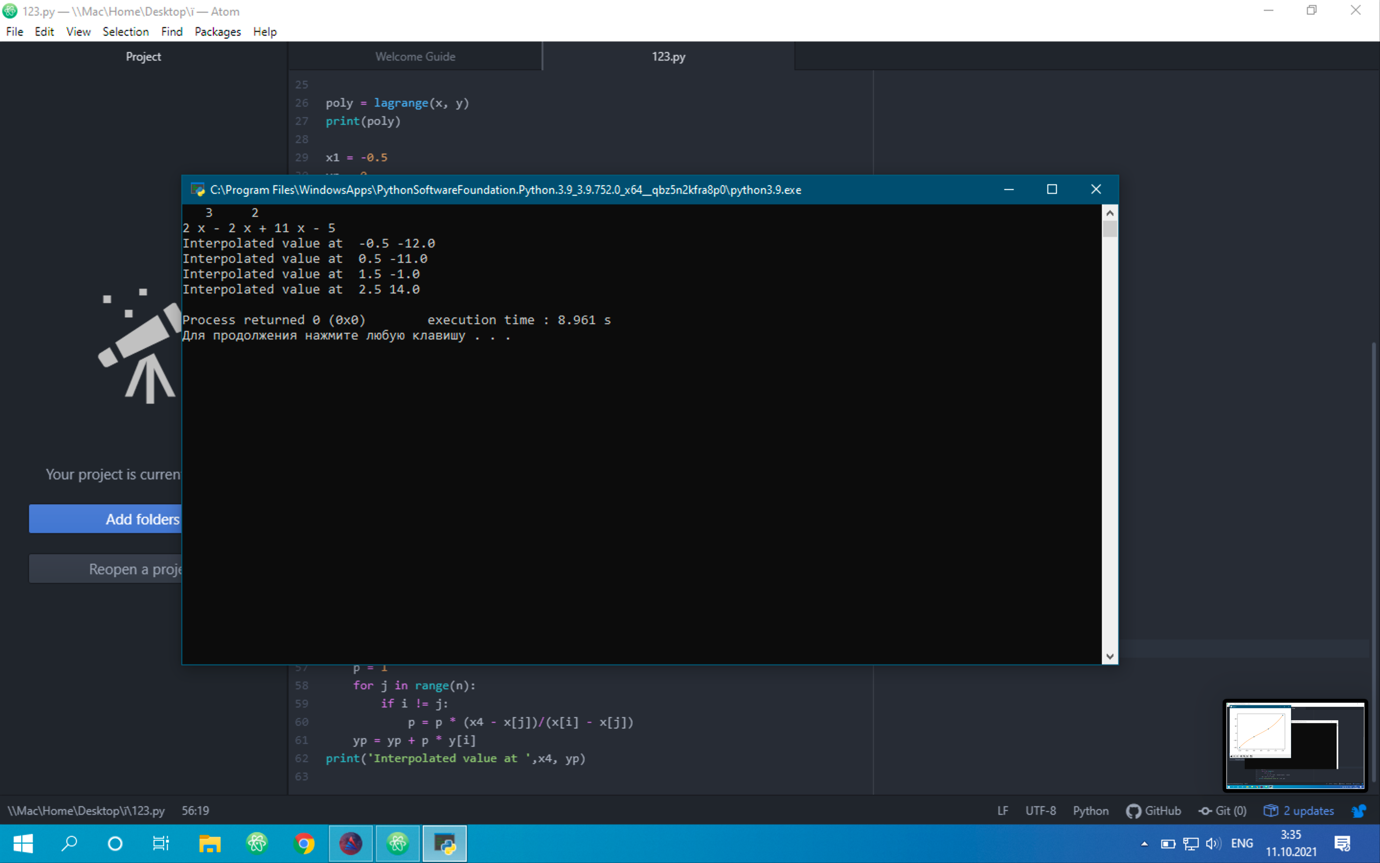
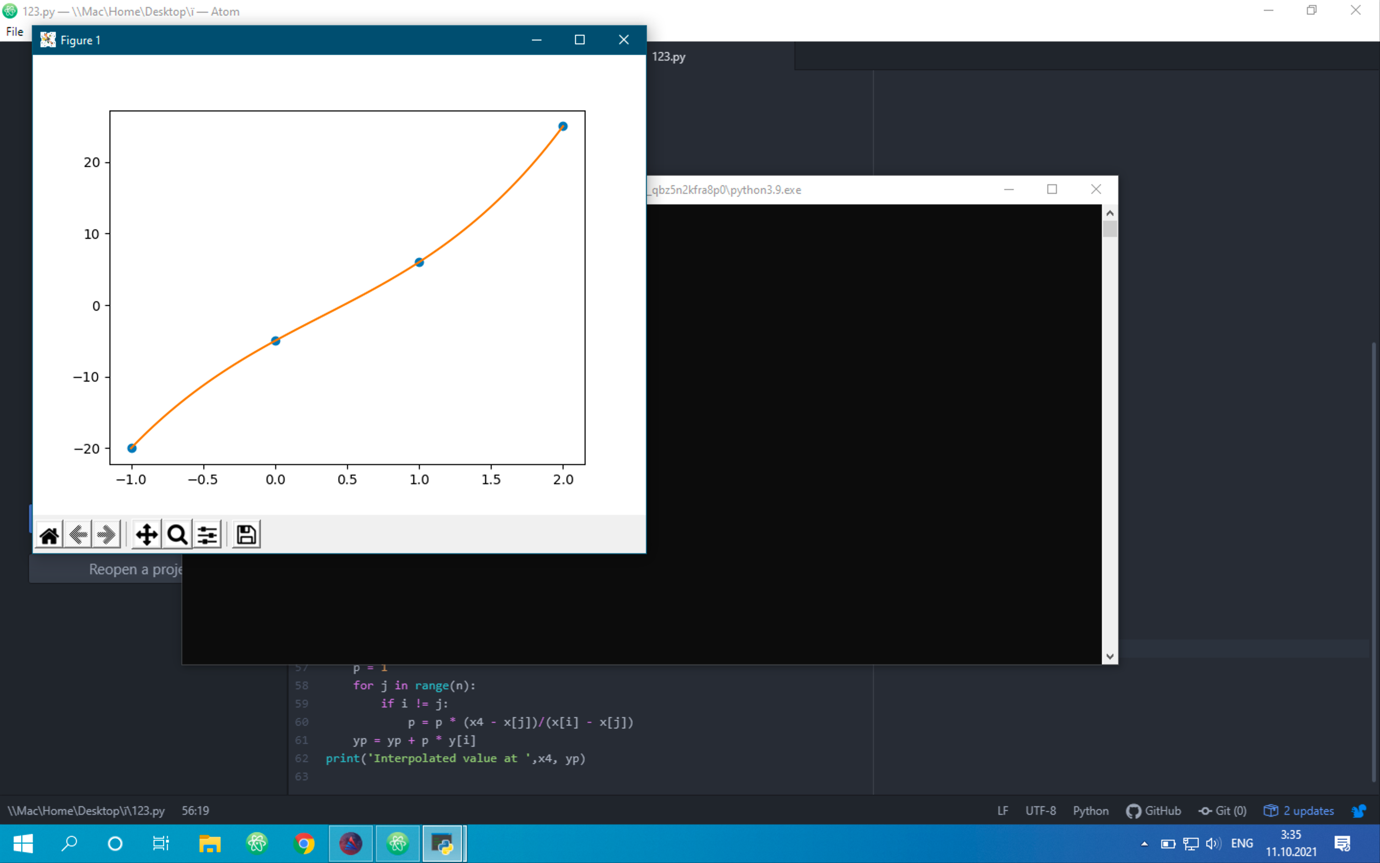
for j in range(n):

if i != j:

p = p \* (x4 - x[j])/(x[i] - x[j])

yp = yp + p \* y[i]

print('Interpolated value at ',x4, yp)



Завдання 2

import numpy as np

import matplotlib.pyplot as plt

import math

x = np.array([i for i in range(6)], dtype=np.complex)

y = np.array([pow(-i, 0.5\*math.sin(10\*i)) for i in x], dtype=np.complex)

fig, ax = plt.subplots()

ax.plot(x, y, color='r')

ax.set(title='f(x)=-x^(1/2)\*sin(10\*x)')

ax.set(xlabel='x')

ax.set(ylabel='y')

ax.grid(True)

plt.legend(['line 1'])

plt.show

