import numpy as np

import matplotlib.pyplot as plt

from scipy import optimize

import math

# Графік системи рівнянь

x\_min, x\_max = -1, 1

y\_min, y\_max = -4, 1

step = 0.01

x, y = np.meshgrid(np.arange(x\_min, x\_max, step),

np.arange(y\_min, y\_max, step))

eq1 = np.sin(x + 2) - y - 1.5

eq2 = x + np.cos(y - 2) - 0.5

fig, ax = plt.subplots(figsize=(10, 10))

ax.contour(x, y, eq1, levels=[0], colors='red')

ax.contour(x, y, eq2, levels=[0], colors='blue')

ax.set\_xlim([x\_min, x\_max])

ax.set\_ylim([y\_min, y\_max])

ax.set\_xlabel('x')

ax.set\_ylabel('y')

ax.set\_title('Графік системи рівнянь')

plt.grid(True)

plt.show()

# Метод простої ітерації

x0 = 0.15

y0 = -2.1

def f1(y):

return 1/3 \* np.cos(y) + 0.3

def f2(x):

return np.sin(x - 0.6) - 1.6

def simple\_iteration(x, y, e):

xn = x

yn = y

xn1 = f2(x)

yn1 = f1(y)

n = 1

while (abs(xn1 - xn) >= e) or (abs(yn1 - yn) >= e):

xn = xn1

yn = yn1

xn1 = f2(yn)

yn1 = f1(xn)

n += 1

print('Simple iteration:')

print('x =', xn, '\ny =', yn, '\nThe amount of iteration =', n)

simple\_iteration(x0, y0, 0.001)

# Перевірка за допомогою optimize.root()

def f3(xy):

x, y = xy

return [np.sin(x + 2) - y - 1.5, x + np.cos(y - 2) - 0.5]

initial\_guess = [0., 0.]

solution\_check = optimize.root(f3, initial\_guess, method='hybr')

print("\nCheck using optimize.root:", solution\_check.x)  
  
