НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені Ігоря Сікорського»

Факультет прикладної математики

Кафедра прикладної математики

Звіт

із лабораторної роботи №4

із дисципліни «Методи обчислення»

на тему

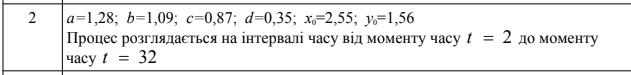
«Системи звичайних диференціальних рівнянь

у моделях хижак-здобич»

2 варіант

|  |  |
| --- | --- |
| Виконав: | Керівник: |
| студент групи КМ-81 | Костюшко Ірина Анатоліївна |
| Верзун П.В. |  |

Київ — 2020

Завдання 1. Запишіть систему рівнянь для свого варіанту завдання.

Застосуйте метод Рунге-Кутта 4-го порядку, щоб отримати розв’язок.

Побудуйте графіки результатів, щоб візуалізувати, як залежні змінні змінюються

з часом.

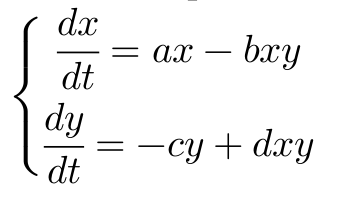
Побудуйте графіки залежності змінних одна від іншої, щоб побачити наявність

якихось закономірностей.

**Теоретичні відомості та формули.**

Модель Лотки-Вольтерри — модель взаємодії двох видів типу «хижак-здобич»,

яка задається системою диференціальних рівнянь:



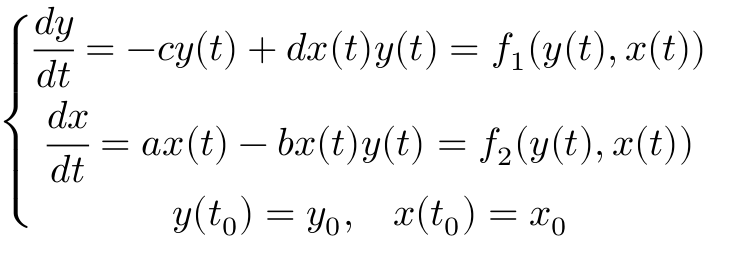
, де x, y — кількість здобичи та хижаків, a — коефіцієнт народжуваності (темп

росту) здобичі, c — коефіцієнт смертності хижака, b — коефіцієнт смерті жертви

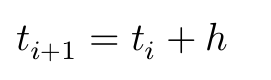
при зустрічі з хижаком, d — коефіцієнт відтворення хижака.

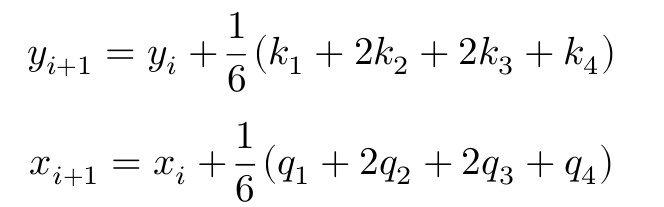
Для розв’язання задачі Коші використовуються чисельні методи Рунге-Кутта-

Фельберга. Запишемо задачу Коші:

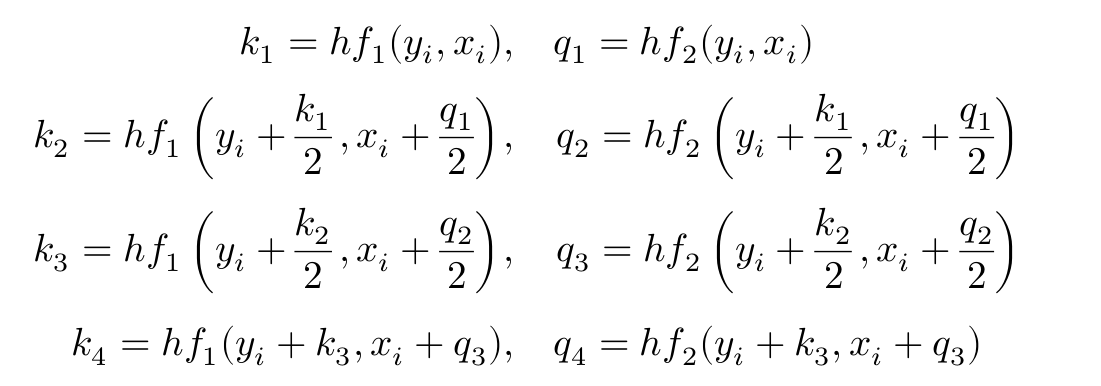


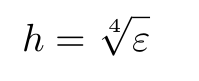
Скристаємося формулами для методу Рунге-Кутта-Фельберга 4-го порядку:





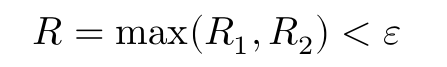
Та коефіцієнти q, k задаються формулами:

Початковий крок методу визначимо з наступним чином:

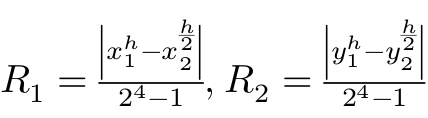


Для забезпечення бажаної точності ε, обчислення ведеться з кроком h та h/2, а потім

оцінюється за наступною формулою:



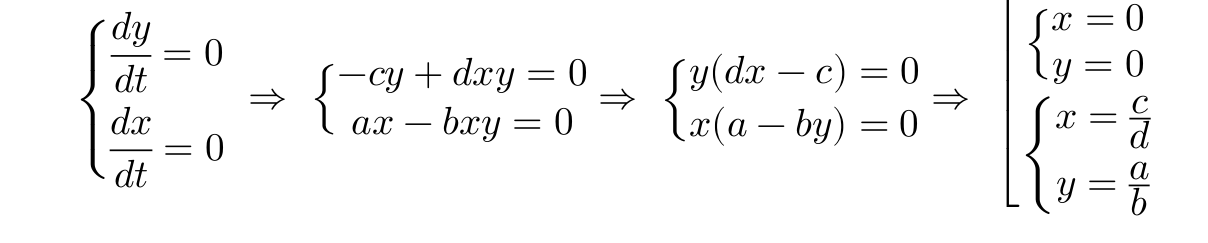
де



Якщо R < ε — досягнута потрібна точність, якщо ні, то повторюємо обчислення

для та h/2 і h/4 так далі, доки не досягнемо потрібної точності.

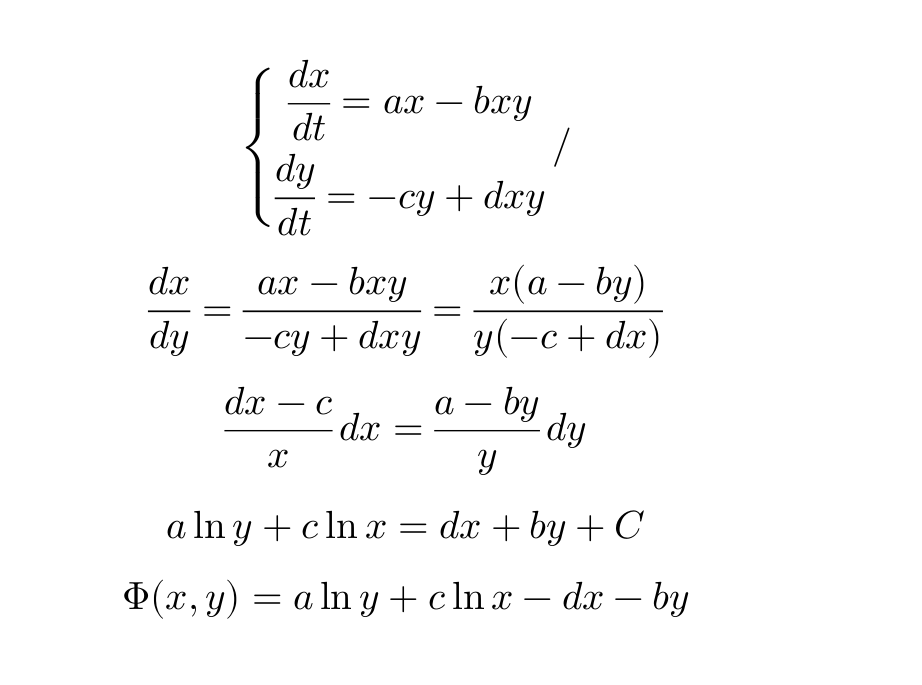
Дана система має критичні точки:



В стаціонарній точці, отримаємо, що прирости нульові, а отже, популяції будуть

константі.

Рівняння траєкторій системи:

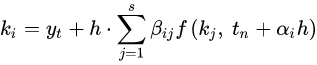


Отримання методу Рунге-Кутта:

В загальному випадку, метод Рунге-Кутта порядку s можна записати так:

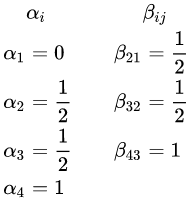


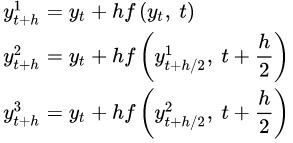
де



k отримаються з похідних yt i-го порядку

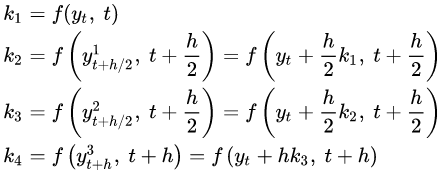
Оберемо s=4

 В інших випадках

 Де

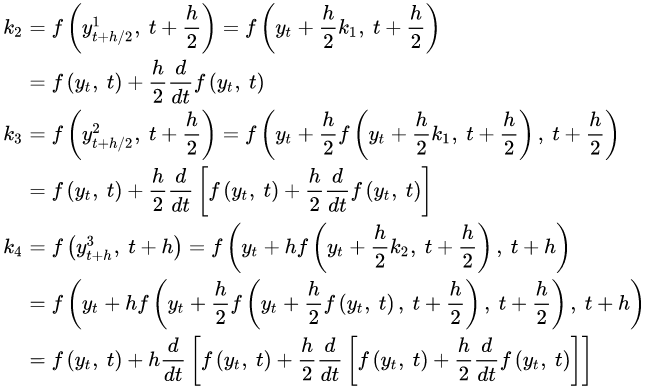
Та

Якщо ми визначимо наступні рівняння

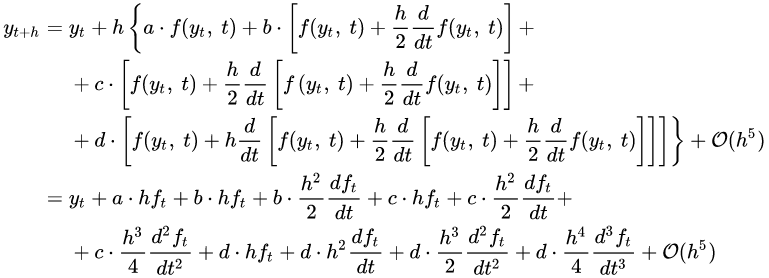
****



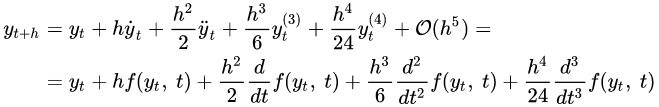
То можна показати, що наступне буде виконуватися для O(h^2)

****

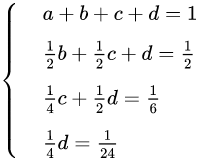
Де

Це повний диференціал f по часу. Тепер ми можемо визначити повну формулу:

Порівнявши її з рядом Тейлора для yh+t в околі yt отримаємо



З чого отримаємо систему обмежень для коєфіцієнтів



Вирішивши яку отримаємо

**Робота програмного забезпечення**

Система рівнянь для даного варіанту має вигляд:

dx/dt = 1.28x — 1.09xy

dy/dt = -0.87x + 0.35xy

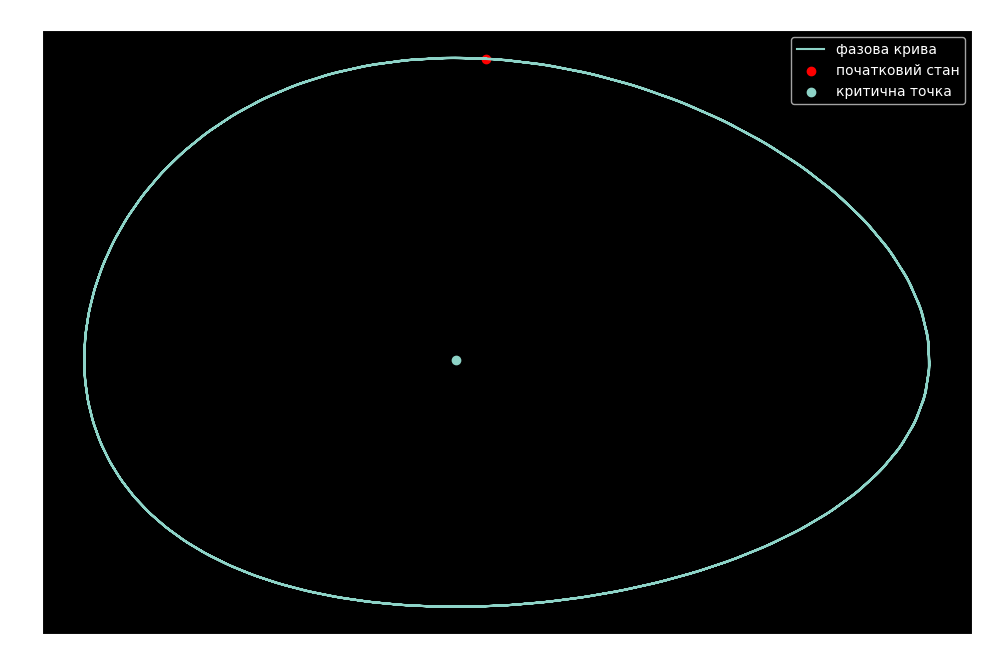
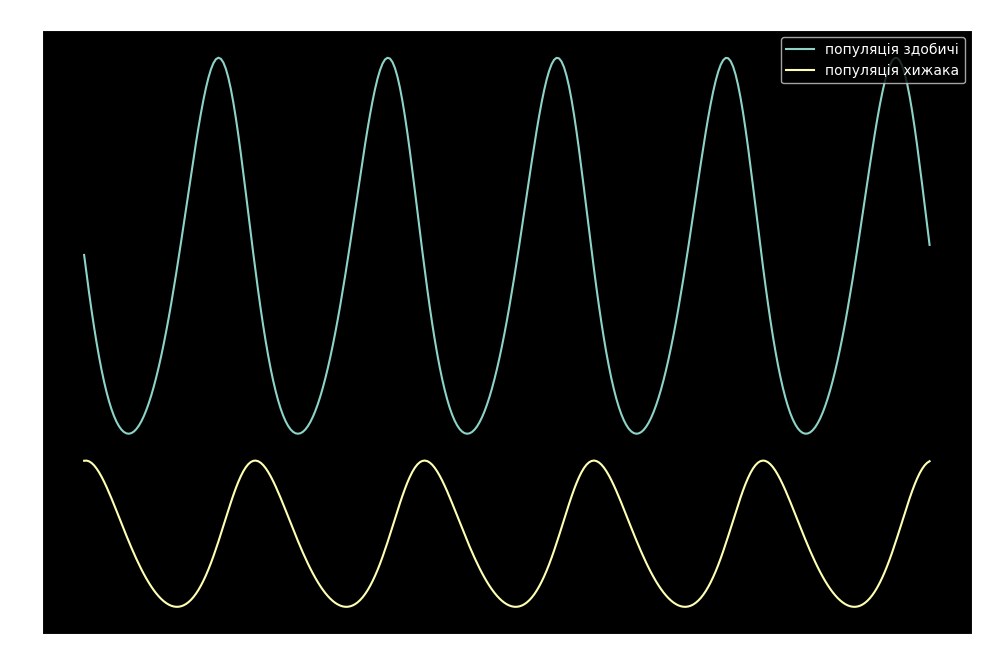
А задача Коші:

dy/dt = - 0.87y(t) + 0.35x(t)y(t) = f1(y(t),x(t))

dx/dt = 1.28x(t) - 1.09x(t)y(t) = f2(y(t),x(t))

y(t0) = 1.56, x(t0) = 2,55

Графік динаміки популяції з часом та графік залежності змінних одна від одної:



зображення траєкторій динамічної системи в фазовому просторі.

Кожен стан системи відповідає певній точці на фазовому портреті. Фазові портрети служать для наочного відображення особливостей еволюції динамічної системи: стаціонарних точок, циклів, басейнів притягання.

Для двовимірної системи фазовий портрет повністю відображає типи траєкторій, які можуть реалізуватися. Для системи більшої вимірності будуються проекції фазових траєкторій на вибрану площину фазового простору.

Подання у вигляді фазової площини є корисним в розпізнаванні внутрішньої структури

моделі. Замість побудови графіків x і y в залежності від t , можна побудувати y в

залежності від x . Цей графік ілюструє спосіб, яким взаємодіють змінні стану x і y , і

називається поданням у фазовій площині.

Висновок: популяція здобичі дуже швидко відновлюється, а популяція хижаків навпаки, має низьку народжуваність, але дуже швидко поїдає здобич. Графіки популяції здобичі та хижаків не перетинаються через велику різницю між

Проміжні результати:

t = 2.0

q1 = -0.18918000000000013, k1 = 0.006194117647058808, q2 = -0.1836252922296023, k2 = -0.0029257039331315117, q3 = -0.18167688751589753, k3 = -0.0026498137128451236, q4 = -0.17449457930700296, k4 = -0.011291771986671693

x = 2.3676201768669993, y = 1.5572918850614057

…

t = 31.823529411764675

q1 = -0.19525115210750188, k1 = 0.029192356091679923, q2 = -0.195994380589872, k2 = 0.020109181150031352, q3 = -0.193612973168766, k3 = 0.020014822731251745, q4 = -0.19172423948923448, k4 = 0.010944048741132124

x = 2.598709249507694, y = 1.5577988960711393

X = [2.55, 2.45695506465553, 2.367617191161977, 2.282769564169779, 2.203013740649603, … , 3.2412151859256024, 3.162280200088909, 3.076327090990422, 2.984988431044423, 2.889935125561417, 2.792809979925006, 2.695169323407744, 2.598436292194397]

Y = [1.56, 1.5608429547747995, 1.5572910052883888, 1.5495709976310377, 1.5379732202491687, … , 1.4146018922732362, 1.4462601484649504, 1.4748609810431088, 1.4999092156701033, 1.5209924236981327, 1.5377954770837465, 1.5501093397274381, 1.5578337601246495]

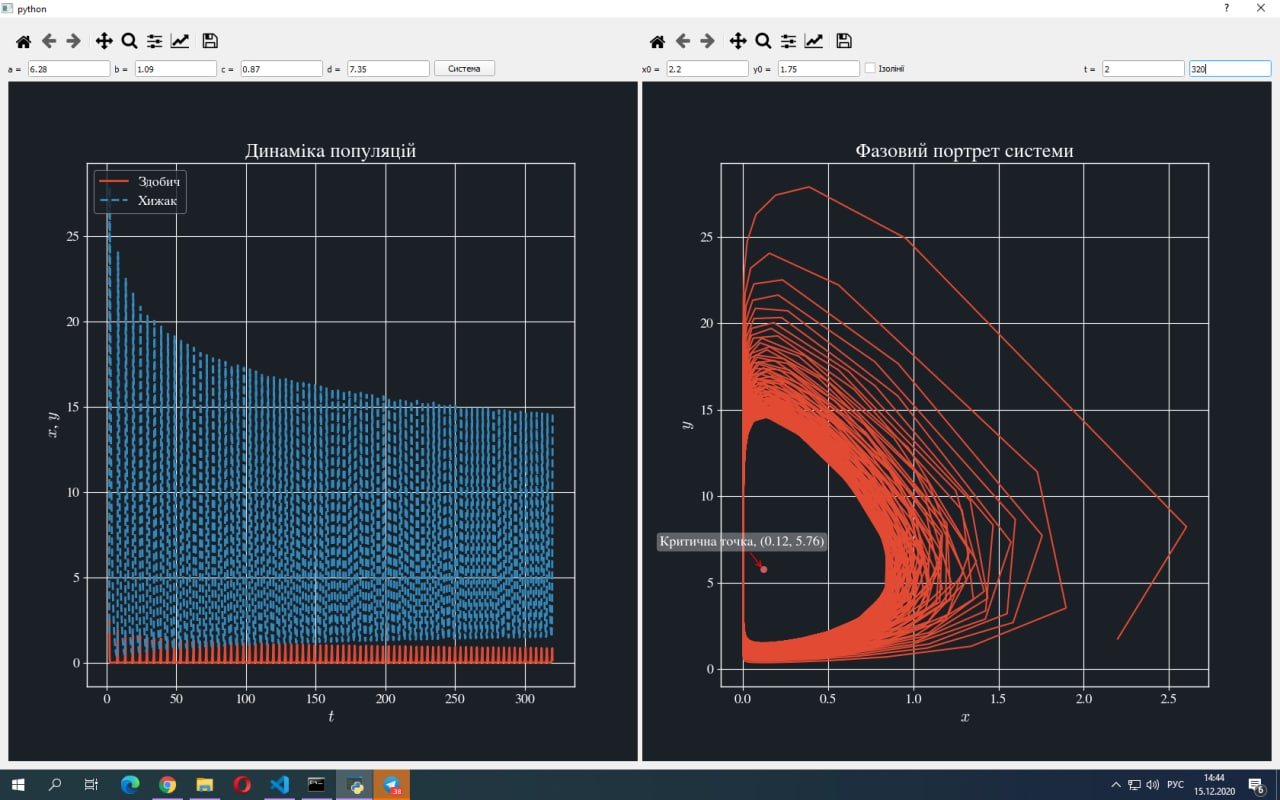
R = 0.00027295731329690653

якщо параметри, використовувані для моделювання на рис. 1, змінити, загальна

картина залишиться, проте амплітуди, фази та періоди будуть мінятися. Таким чином, є

нескінченна кількість циклів, які можуть відбутися.

При інших значеннях коефіцієнтів графіки будуть змінюватися:



Код програми:

from math import \*

import numpy as np

import matplotlib.pyplot as plt

fig=plt.figure(figsize=(12,8), dpi= 100, facecolor='w', edgecolor='k')

def Runge\_Kutta(x0,y0,t0,tn,h,f1,f2):

x,y = x0,y0

X,Y = [x], [y]

for t in np.arange(t0,tn,h):

q1 = h\*f1(x,y)

k1 = h\*f2(x,y)

q2 = h\*f1(x+(q1/2),y+(k1/2))

k2 = h\*f2(x+(q1/2),y+(k1/2))

q3 = h\*f1(x+(q2/2),y+(k2/2))

k3 = h\*f2(x+(q2/2),y+(k2/2))

q4 = h\*f1(x+q3,y+k3)

k4 = h\*f2(x+q3,y+k3)

x = x + (1/6)\*(q1 + 2 \* q2 + 2 \* q3 + q4)

y = y + (1/6)\*(k1 + 2 \* k2 + 2 \* k3 + k4)

if t==np.arange(t0,tn,h)[0] or t==np.arange(t0,tn,h)[-1]:

print("t = ", t)

print("q1 = {}, k1 = {}, q2 = {}, k2 = {}, q3 = {}, k3 = {}, q4 = {}, k4 = {}".format(q1,k1,q2,k2,q3,k3,q4,k4))

print("x = {}, y = {}".format(x, y))

X.append(x)

Y.append(y)

return X,Y

def Lotka\_Volterra(a,b,c,d,x0, y0, t0, tn,epsilon):

h = epsilon \*\* (1/4)

n = ceil((tn-t0)/h)

if n % 2:

n+=1

h = (tn-t0)/n

def f1(x,y): return a\*x - b\*x\*y

def f2(x,y): return -c\*y + d\*x\*y

X,Y = Runge\_Kutta(x0,y0,t0,tn,h,f1,f2)

while True:

h /= 2

X2, Y2 = Runge\_Kutta(x0,y0,t0,tn,h,f1,f2)

print("X = ", X2)

print("\n")

print("Y = ", Y2)

r1 = max([abs(x - x2) for x, x2 in zip(X, X2[::2])])

r2 = max([abs(y - y2) for y, y2 in zip(Y, Y2[::2])])

r = max(r1, r2)

print("\n")

print("R = ", r)

X, Y = X2, Y2

if r < epsilon:

return X2, Y2, h, c / d, a / b

epsilon = 0.001

a,b,c,d = 1.28, 1.09, 0.87, 0.35

x0,y0 = 2.55, 1.56

t0,tn = 2, 32

plt.style.use('dark\_background')

X2, Y2, h, xkr,ykr = Lotka\_Volterra(a,b,c,d,x0, y0, t0, tn,epsilon)

T = np.arange(t0, tn + h / 2, h)

plt.title("Графіки популяцій з часом")

plt.plot(T, X2, label = 'популяція здобичі')

plt.plot(T, Y2, label = 'популяція хижака')

plt.legend(loc= 'upper right')

plt.show()

fig=plt.figure(figsize=(12,8), dpi= 100, facecolor='w', edgecolor='k')

plt.title("Фазовий портрет популяцій")

plt.scatter(x0,y0, label = 'початковий стан', color='red')

plt.scatter(xkr,ykr, label = 'критична точка')

plt.plot(X2,Y2,label = 'фазова крива')

plt.legend(loc= 'upper right')

plt.show()