

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра дискретной математики и информационных технологий

ЛАБОРАТОРНАЯ РАБОТА №2

студента 4 курса 421 группы

направления 09.03.01 Информатика и вычислительная техника

факультета компьютерных наук и информационных технологий

Морозова Никиты Андреевича

Преподаватель

Станкевич Елена Петровна

Саратов 2024

Вариант 10.

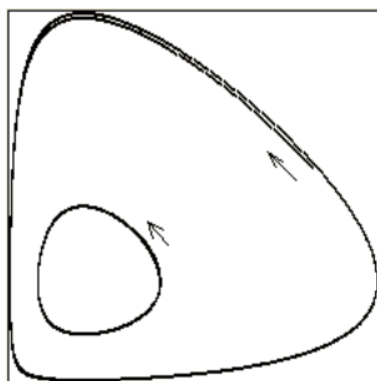
Задача 10. Простейшая модель «хищник-жертва». Система дифференциальных уравнений Лотки-Вольтерра [8]

$$\begin{cases} dx/dt = a_1x - b_1yx, \\ dy/dt = -a_2y + b_2yx \end{cases}$$

является математической моделью совместного существования двух биологических видов (популяций) типа «хищник-жертва» в некоторой изолированной среде. Здесь y – число хищников, x – число жертв. Коэффициент $a_1 > 0$ определяет уровень рождаемости жертв, $b_1 > 0$ – коэффициент уменьшения числа жертв от встречи с хищниками, $a_2 > 0$ – коэффициент уменьшения числа хищников от голода, $b_2 > 0$ – коэффициент увеличения числа хищников. Среда предоставляет жертвам питание в неограниченном количестве, а хищники питаются лишь жертвами.

Построить фазовый портрет системы дифференциальных уравнений при допустимых параметрах a_1, a_2, b_1, b_2 .

Пояснение к задаче. Схематически фазовый портрет этой системы дифференциальных уравнений имеет вид:



```

import numpy as np
import matplotlib.pyplot as plt

a1 = 1.0 # Уровень рождаемости жертв
b1 = 0.1 # Коэффициент уменьшения числа жертв от встречи с хищниками
a2 = 1.5 # Коэффициент уменьшения числа хищников от голода
b2 = 0.075 # Коэффициент увеличения числа хищников

x0 = 40 # Число жертв
y0 = 9 # Число хищников

t = np.linspace(0, 200, 10000)
dt = t[1] - t[0]

def prey_population(x, y):
    return a1 * x - b1 * y * x

def predator_population(x, y):
    return -a2 * y + b2 * y * x

def euler_method(x0, y0, t):
    x = np.zeros(len(t))
    y = np.zeros(len(t))
    x[0] = x0
    y[0] = y0
    for i in range(1, len(t)):
        x[i] = x[i - 1] + prey_population(x[i - 1], y[i - 1]) * dt
        y[i] = y[i - 1] + predator_population(x[i - 1], y[i - 1]) * dt
    return x, y

x_values, y_values = euler_method(x0, y0, t)

plt.figure(figsize=(10, 6))
plt.plot(x_values, y_values, label="Фазовый портрет")

# Построение стрелок
x = np.linspace(0, 100, 20)
y = np.linspace(0, 100, 20)
X, Y = np.meshgrid(x, y)
U = prey_population(X, Y)
V = predator_population(X, Y)
magnitude = np.sqrt(U**2 + V**2)
U = U / magnitude
V = V / magnitude
plt.quiver(X, Y, U, V, color='r', alpha=0.6)

plt.xlim(0, 100)
plt.ylim(0, 60)
plt.xlabel("Число жертв (x)")
plt.ylabel("Число хищников (y)")
plt.title("Фазовый портрет системы Лотки-Вольтерра")
plt.legend()
plt.grid(True)
plt.show()

```

Фазовый портрет системы Лотки-Вольтерра

