

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

« Российский химико-технологический университет имени Д.И. Менделеева »

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7

Вариант 7

Выполнил студент группы КС-46: Золотухин Андрей Александрович

Ссылка на репозиторий: <https://github.com/>

CorgiPuppy/

info-processes-systems-theory-labs

Принял: Зинченко Дарья Ивановна

Дата сдачи: 17.12.25

Москва

2025

Оглавление

Описание задачи	3
Работа 7	3
Выполнение задачи	4
Аналитическое решение	4
Графики	4
Код	8
Выводы	10

Описание задачи

1. Аналитическим способом найти стационарную точку и определить характер её устойчивости по 1-ому методу Ляпунова;
2. Написать программу, решающую систему ДУ и строящую следующие графики:
 1. фазовый портрет системы. Подберите начальные условия, шаг по времени и масштаб таким образом, чтобы тип точки и ее координаты на графике были очевидны (в случае неустойчивой точки начальные условия рекомендуется задавать как можно ближе к неподвижной точке). Число траекторий не меньше восьми!
 2. зависимости $x_1(t)$ и $x_2(t)$ - динамику системы во времени (таким образом, чтобы поведение системы в окрестности неподвижной точки и ее координаты были очевидны.) На одном графике должно быть несколько линий динамики при разных начальных условиях.

Работа 7

Система уравнений:

$$\begin{cases} \frac{dx_1}{dt} = (x_1 - 1)^2 + \dot{\alpha}, \\ \frac{dx_2}{dt} = x_1 - x_2. \end{cases}$$

1. Найти неподвижные точки и определить их тип при $\dot{\alpha} > 0$, $\dot{\alpha} = 0$, $\dot{\alpha} < 0$, исследовав систему по 1-му методу Ляпунова.
2. Указать тип бифуркации и точку бифуркации.
3. Построить бифуркационную диаграмму (зависимость стационарного решения от параметра $\dot{\alpha}$).
4. Построить фазовые портреты при следующих значениях управляющего параметра $\dot{\alpha}$: $-0.25, 0, 0.25$.

Для решения использовать схему явную Эйлера:

$$\begin{cases} \frac{x_1^{n+1} - x_1^n}{\Delta t} = f(x_1^n), \\ \frac{x_2^{n+1} - x_2^n}{\Delta t} = f(x_2^n). \end{cases}$$

Выполнение задачи

Аналитическое решение

Графики

На Рис. 1 представлена бифуркационная диаграмма.

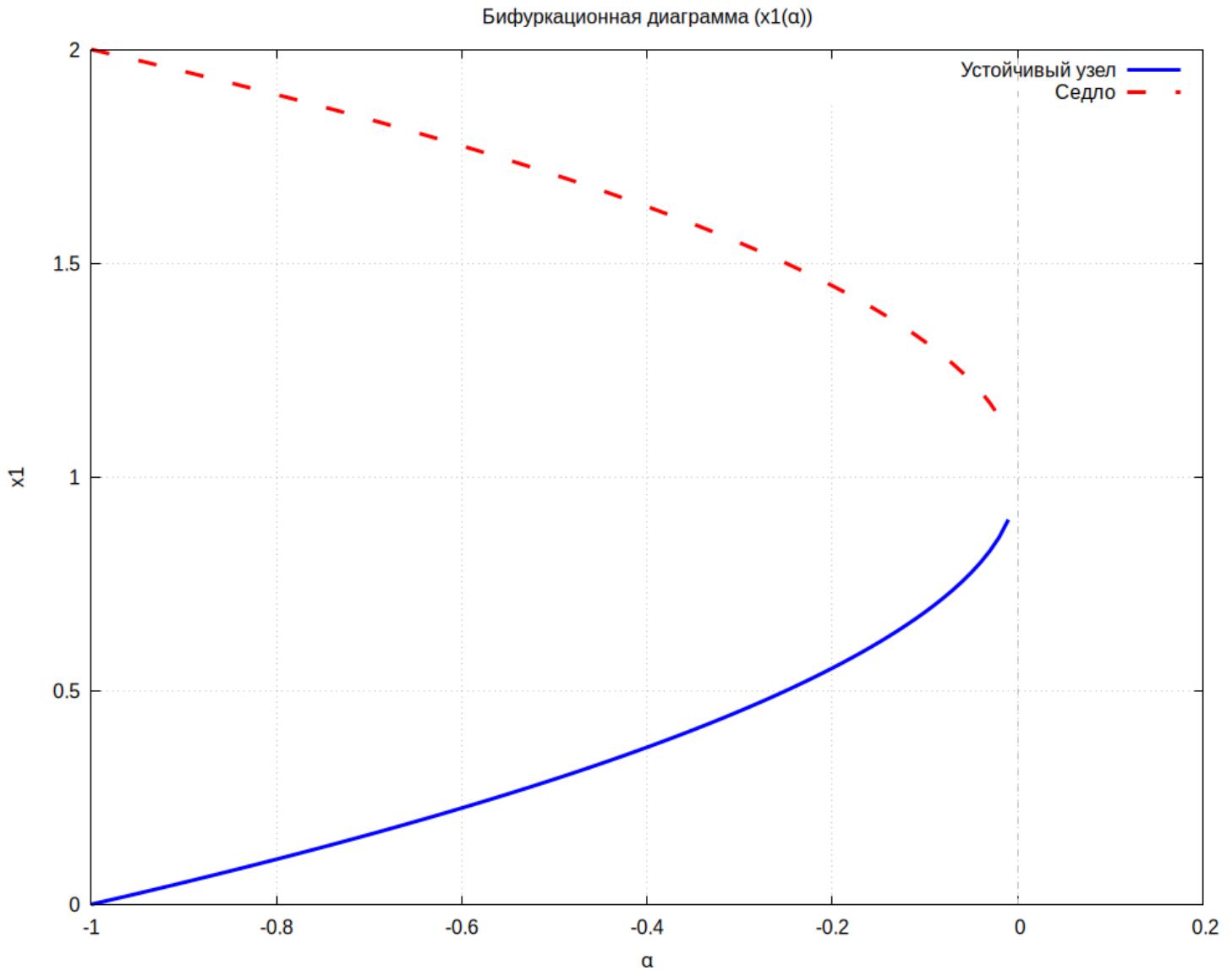


Рис. 1: Бифуркационная диаграмма ($x_1(\dot{\alpha})$).

На Рис. 2 представлен фазовый портрет при $\dot{\alpha} = -0.25$.

Фазовый портрет ($\alpha = -0.25$)

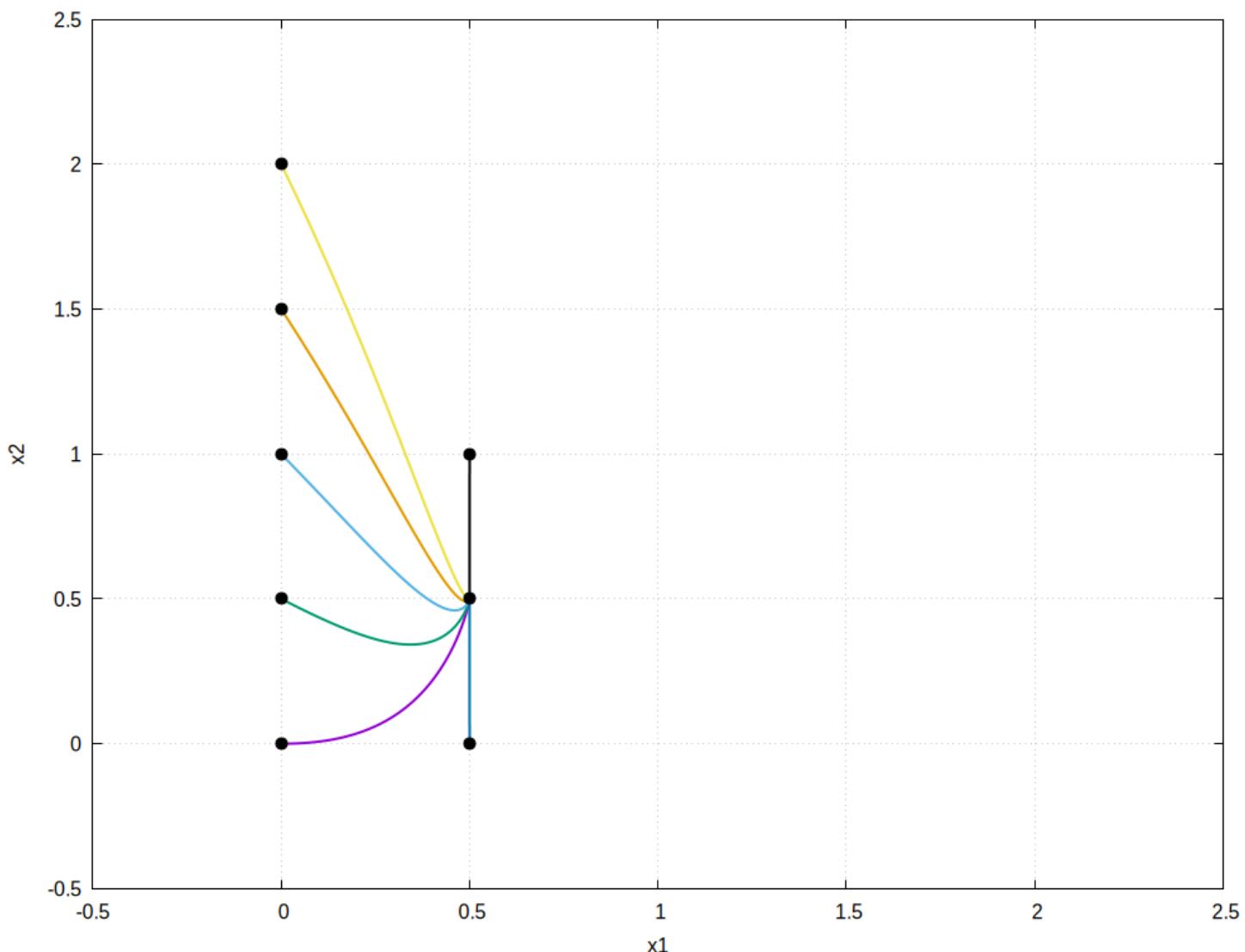


Рис. 2: Фазовый портрет при $\dot{\alpha} = -0.25$.

На Рис. 3 представлен фазовый портрет при $\dot{\alpha} = 0$.

Фазовый портрет ($\alpha = 0$)

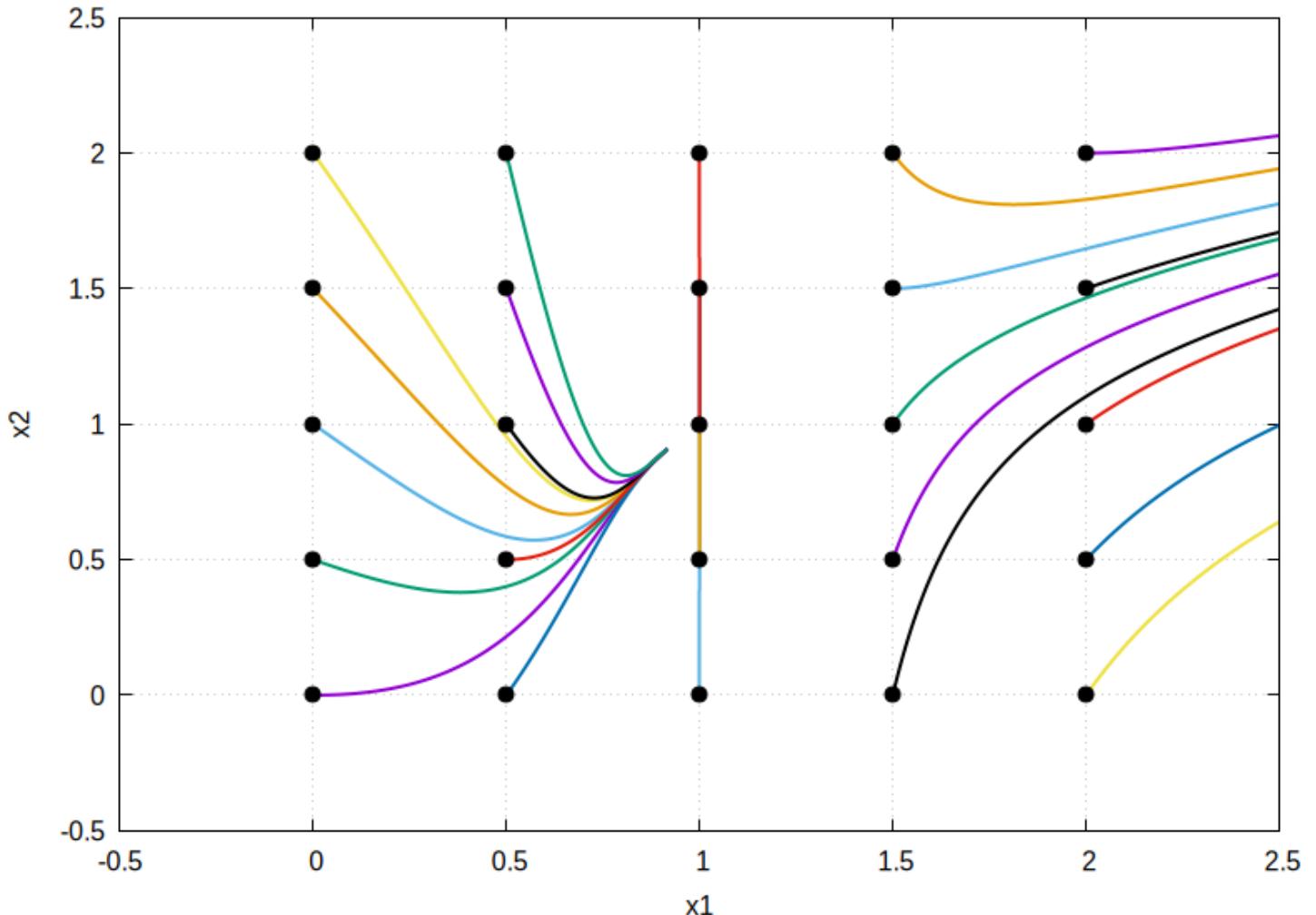


Рис. 3: Фазовый портрет при $\dot{\alpha} = 0$.

На Рис. 4 представлен фазовый портрет при $\dot{\alpha} = 0.25$.

Фазовый портрет ($\alpha = 0.25$)

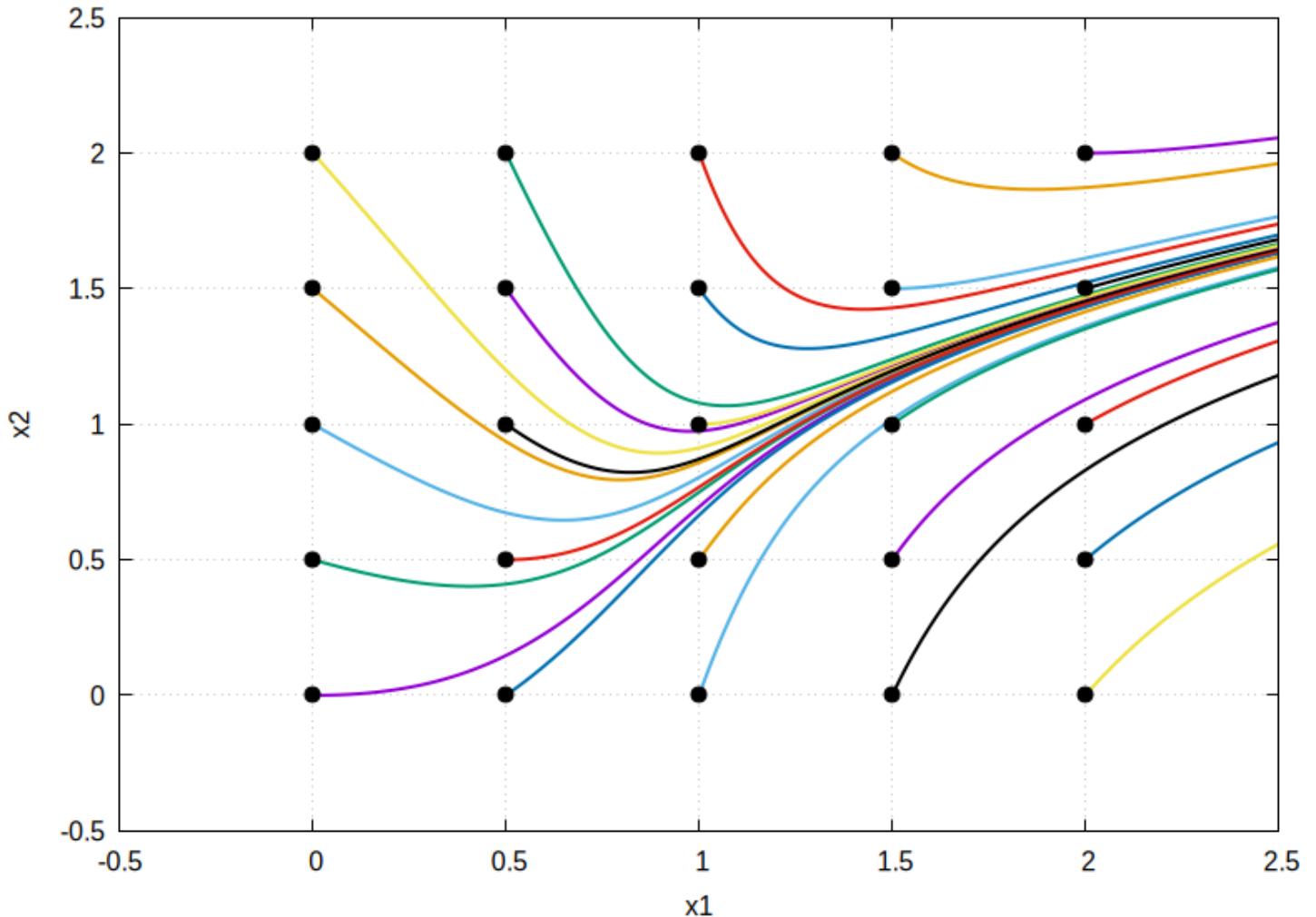


Рис. 4: Фазовый портрет при $\dot{\alpha} = 0.25$.

Код

```
#ifndef CONSTANTS_H
#define CONSTANTS_H

#include <string>

namespace Constants {
    const std::string DAT_DIR = "plots/dat-files/";

    const double dt = 0.01;
    const double T = 10.0;

    const std::vector<double> alphas = {
        -0.25,
        0.0,
        0.25
    };

    const std::vector<std::string> filenames = {
        "alpha_minus_0_25.dat",
        "alpha_0.dat",
        "alpha_plus_0_25.dat"
    };

    const std::string bif_filename = "bifurcation.dat";
}

#endif
```

```
#include <iostream>
#include <vector>
#include <cmath>
#include <fstream>
#include <string>
#include <iomanip>

#include "Constants.h"

void step(double&, double&, double, double);

struct Point {
    double x;
    double y;
};

int main() {
    std::vector<Point> starts;
    for (double x = 0.0; x <= 2.0; x += 0.5)
        for (double y = 0.0; y <= 2.0; y += 0.5)
            starts.push_back({x, y});

    for (long unsigned int i = 0; i < Constants::alphas.size(); i++) {
        double alpha = Constants::alphas[i];
        std::string path = Constants::DAT_DIR + Constants::filenames[i];

        std::ofstream file(path);
        if (!file.is_open()) {
```

```

    std::cerr << "Ошибка открытия файла: " << path << std::endl;
    continue;
}
file << std::fixed << std::setprecision(6);

for (Point start : starts) {
    double x1 = start.x;
    double x2 = start.y;

    for (double t = 0; t <= Constants::T; t += Constants::dt) {
        if (std::abs(x1) > 50.0 || std::abs(x2) > 50.0)
            break;

        file << t << " " << x1 << " " << x2 << "\n";
        step(x1, x2, Constants::dt, alpha);
    }
    file << "\n\n";
}

file.close();
}

std::string bif_path = Constants::DAT_DIR + Constants::bif_filename;
std::ofstream bifFile(bif_path);

if (bifFile.is_open()) {
    for (double a = -1.0; a <= 0.2; a += 0.01) {
        if (a <= 0) {
            double x_stable = 1.0 - std::sqrt(-a);
            double x_unstable = 1.0 + std::sqrt(-a);

            bifFile << a << " " << x_stable << " " << x_unstable << "\n";
        }
    }
    bifFile.close();
}

return 0;
}

void step(double& x1, double& x2, double dt, double alpha) {
    double dx1 = (std::pow(x1 - 1.0, 2) + alpha) * dt;
    double dx2 = (x1 - x2) * dt;

    x1 += dx1;
    x2 += dx2;
}

```

Выводы