

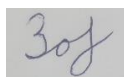
МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Санкт-Петербургский государственный технологический институт
(технический университет)»
(СПбГТИ (ТУ))

Отчет об учебной практике

**Тема: «Разработка программного комплекса, реализующего поддержку решения
прямой задачи кинетики»**

Направление подготовки (специальность)	09.03.03	Прикладная информатика
Направленность (специализация)	Прикладная информатика в химии	
Факультет	Информационных технологий и управления	
Кафедра	Систем автоматизированного проектирования и управления	
Группа	485	

Студент



(подпись, дата)

Зобнин И.М.

(инициалы, фамилия)

Руководитель,
доц., к.т.н.

(подпись, дата)

Уланов В.Н.

(инициалы, фамилия)

Руководитель,
доц., к.т.н.

(подпись, дата)

Разыграев А.С.

(инициалы, фамилия)

доц., к.т.н.

(подпись, дата)

Макарук Р.В.

(инициалы, фамилия)

Санкт-Петербург
2020

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	2
1. АНАЛИТИЧЕСКАЯ ЧАСТЬ.....	3
1.1 Выбор языка программирования.....	3
1.2 Выбор среды разработки	4
1.3 Знакомство с математическим моделированием кинетики (решение прямой задачи кинетики).....	5
1.3.1 Аналитический метод решения прямой задачи кинетики.....	5
1.3.2 Численный метод решения прямой задачи кинетики	7
1.3.3 Вывод по аналитическому и численному методам	8
2. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ	10
2.1 Краткое описание и блок-схема алгоритма.....	10
2.2 Обзор интерфейса программы	11
2.3 Тестирование программного комплекса, анализ результатов	11
ВЫВОДЫ.....	13
ПРИЛОЖЕНИЕ А.....	14

ВВЕДЕНИЕ

Данная учебная практика необходима для получения первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности и исполнительской практики. Целью этой практической работы является разработка программного комплекса, моделирующего кинетику гомогенной реакции. Задание следующее:

$A \xrightarrow{k_1} B \xrightarrow{k_2} A$, где $k_1 = 2$, $k_2 = 2$ – скорости протекания реакций.

С концентрацией веществ А, В: 100 и 50 соответственно.

Виртуальный эксперимент предполагает решение прямой задачи кинетики, вывод концентрации веществ с течением времени с заданным шагом, а также начальными концентрациями как в табличном, так и в графическом виде.

1. АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1 Выбор языка программирования

Для разработки программного комплекса, предназначенного для проведения виртуального эксперимента было рассмотрено несколько языков программирования.

Первым рассмотрим C++. Этот язык был разработан как универсальный язык со статическими типами данных, без промежуточной среды выполнения.

Достоинства: полный контроль практически над всем, и использование только тех элементов, которые необходимы; прямая работа с памятью; кроссплатформенность.

Недостатки: высокий порог вхождения; необходимость следить за многими мелкими деталями (например, освобождение памяти); сложность обнаружения ошибок.

Java является гибким языком высокого уровня, который включает в себя концепцию объектно-ориентированного программирования (ООП).

Достоинства: независимость от платформы; удобен для совместной работы; автоматическое управление памятью; многопоточность; большое количество стандартных библиотек;

Недостатки: низкая производительность; сложность работы с дизайном программы; громоздкость кода;

Для C# характерен фокус на абстракциях, ООП и ограничения на прямой доступ к памяти.

Достоинства: небольшой порог вхождения для людей, знакомых с C-подобными языками; необязательное управление памятью, но есть возможность делать это самостоятельно; богатые стандартные библиотеки и сторонние библиотеки, освобождающие от необходимости писать множество кода («синтаксический сахар»).

Недостатки: возможность работы с памятью напрямую понижает надежность, иногда ограничивает функционал программы; многие абстракции мешают в освоении языка; зависимость от Microsoft и Windows;

Тем не менее, для разработки программного комплекса был выбран язык C#, поскольку данный язык использует объектно-ориентированный подход к программированию, имеет за собой множество встроенных и сторонних библиотек, позволяющих не задумываться о написании множества кода и сильно ускоряющих написание программы. Также этот язык считается наилучшим в среде программистов для написания интерфейсных приложений под ОС Windows, что и является целью этой работы.

1.2 Выбор среды разработки

Для создания программного комплекса были рассмотрены следующие среды разработки.

Visual Studio — среда разработки от Microsoft, известна для написания приложений, включающих в себя .NET. Это полный набор инструментов, позволяющий произвести точную отладку и настройку приложения.

Достоинства: поддержка множества языков; постоянно увеличивающаяся огромная библиотека расширений; отладчик Windows, позволяющий найти ошибку в программе по ходу её выполнения; список ошибок, упрощающих отладку; есть бесплатная версия.

Недостатки: для работы требуются значительные ресурсы компьютера; высокий порог вхождения для полноценного использования; множество непонятных и ненужных новичку функций.

Eclipse — среда разработки, изначально ориентированная на работу с Java, прославилась большим количеством внешних модулей, существенно расширяющих её функциональность (в том числе, это касается количества поддерживаемых языков).

Достоинства: множество пакетных решений, обеспечивающих многоязычную поддержку; интерфейс, ориентированный на задачи, включая уведомления в системном трее; автоматическое создание отчетов об ошибках; платная.

Недостатки: нехватка документации; отсутствие единого сообщества разработчиков; среда разработки считается устаревшей.

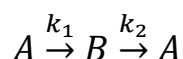
Выбранной средой разработки стала Visual Studio, так как она многофункциональна, к ней возможно подключение дополнительных пакетов, данных. Также для выбранного языка, а именно C#, трудно найти другую среду разработки.

1.3 Знакомство с математическим моделированием кинетики (решение прямой задачи кинетики)

Решить прямую задачу кинетики можно несколькими путями, например, аналитическим или числовым. Для более наглядной демонстрации простоты последнего лучше начать с аналитического.

1.3.1 Аналитический метод решения прямой задачи кинетики

Итак, дана обратимая реакция:



По определению, скорость рассматриваемой химической реакции $w = \frac{dC}{dt}$, где C – концентрация вещества, t – момент времени.

Согласно основному постулату химической кинетики, $w = kC$. Тогда уравнения для расчёта концентрации имеют вид:

$$\begin{cases} \frac{dC_a}{dt} = -k_1 C_a + k_2 C_b \\ \frac{dC_b}{dt} = k_1 C_a - k_2 C_b, \end{cases} \quad (\text{с. 1})$$

где k_1 и k_2 – скорости протекания реакций, C_a и C_b – искомые концентрации веществ А и В, t – время.

Для упрощения дальнейших вычислений составим уравнение материального баланса, отражающее соотношение между количествами исходных материалов и полученного готового продукта:

$$C_a + C_b = C_a^0 + C_b^0$$

или

$$C_b = C_a^0 + C_b^0 - C_a, \text{ (ур.1)}$$

где C_a^0, C_b^0 – начальные концентрации веществ А и В (в нашем случае 100 и 50).

Сначала найдём формулу для расчёта концентрации вещества А в момент времени t :

$$\frac{dC_a}{dt} = -k_1 C_a + k_2 C_b \text{ (ур.2)}$$

Подставим ур.1 в ур.2.

$$\frac{dC_a}{dt} = -k_1 C_a + k_2 (C_a^0 + C_b^0 - C_a)$$

$$\frac{dC_a}{dt} = -k_1 C_a + k_2 C_a^0 + k_2 C_b^0 - k_2 C_a$$

$$\frac{dC_a}{dt} = k_2 (C_a^0 + C_b^0) - C_a (k_1 + k_2) \quad | : (k_2 (C_a^0 + C_b^0) - C_a (k_1 + k_2)) , \times dt$$

$$\frac{dC_a}{k_2 (C_a^0 + C_b^0) - C_a (k_1 + k_2)} = dt$$

Проинтегрируем левую часть по dC_a , а правую по dt .

$$\int_{C_a^0}^{C_a} \frac{dC_a}{k_2 (C_a^0 + C_b^0) - C_a (k_1 + k_2)} = \int_0^t dt$$

В результате интегрирования получим следующее:

$$-\frac{1}{k_1 + k_2} \ln \frac{k_2 (C_a^0 + C_b^0) - C_a (k_1 + k_2)}{k_2 C_b^0 - k_1 C_a^0} = t$$

Проведём преобразования:

$$e^{-t(k_1+k_2)} = \frac{k_2 (C_a^0 + C_b^0) - C_a (k_1 + k_2)}{k_2 C_b^0 - k_1 C_a^0}$$

$$(k_2 C_b^0 - k_1 C_a^0) e^{-t(k_1+k_2)} = k_2 (C_a^0 + C_b^0) - C_a (k_1 + k_2)$$

$$C_a = \frac{k_2 (C_a^0 + C_b^0) - (k_2 C_b^0 - k_1 C_a^0) e^{-t(k_1+k_2)}}{k_1 + k_2}$$

В результате всех проведённых выше преобразований получим следующую формулу для расчёта концентрации вещества А в момент времени t для обратимой реакции.

$$C_a = \frac{k_2(C_a^0 + C_b^0)}{k_1 + k_2} - \frac{(k_2 C_b^0 - k_1 C_a^0)e^{-t(k_1+k_2)}}{k_1 + k_2}$$

Далее, обратимся к уравнению материального баланса и вспомним, что:

$$C_b = C_a^0 + C_b^0 - C_a$$

Отсюда следует, что:

$$C_b = C_a^0 + C_b^0 - \frac{k_2(C_a^0 + C_b^0)}{k_1 + k_2} + \frac{(k_2 C_b^0 - k_1 C_a^0)e^{-t(k_1+k_2)}}{k_1 + k_2}$$

Подставим значения $C_a^0 = 100$, $C_b^0 = 50$, $k_1 = 2$, $k_2 = 2$ и получим окончательную формулу для вычисления концентрации вещества А:

$$C_a = 25e^{-4t} + 75$$

Для вычисления концентрации вещества В:

$$C_b = 75 - 25e^{-4t}$$

1.3.2 Численный метод решения прямой задачи кинетики

Для решения поставленной задачи был сделан выбор в пользу метода Рунге-Кутты. Метод Рунге-Кутты используют для расчета стандартных моделей достаточно часто, так как при небольшом объеме вычислений он обладает точностью метода $O^4(h)$. Он использует промежуточные моменты времени для вычисления состояния в момент времени t_{k+1} с шагом h . Следующие формулы определяют алгоритм Рунге-Кутты четвертого порядка для системы из двух дифференциальных уравнений (с.1).

$$\text{Пусть } t = x_n, C_a = y_n, C_b = z_n, \frac{dC_a}{dt} = f(x_n, y_n, z_n), \frac{dC_b}{dt} = h(x_n, y_n, z_n).$$

Тогда (с.1) будет выглядеть так:

$$\begin{cases} f = -k_1 y_n + k_2 z_n \\ g = k_1 y_n - k_2 z_n \end{cases} \quad (\text{с. 2})$$

$$\text{Также } y_{n+1} = y_n + \frac{j_1 + 2j_2 + 2j_3 + j_4}{6} \quad (\text{ур. 3})$$

$$z_{n+1} = z_n + \frac{l_1 + 2l_2 + 2l_3 + l_4}{6} \quad (\text{ур. 4}),$$

где

$$j_1 = hf(x_n, y_n, z_n);$$

$$\begin{aligned}
l_1 &= hg(x_n, y_n, z_n); \\
j_2 &= hf(x_n + 0.5h, y_n + 0.5j_1, z_n + 0.5l_1); \\
l_2 &= hg(x_n + 0.5h, y_n + 0.5j_1, z_n + 0.5l_1); \\
j_3 &= hf(x_n + 0.5h, y_n + 0.5j_2, z_n + 0.5l_2); \\
l_3 &= hg(x_n + 0.5h, y_n + 0.5j_2, z_n + 0.5l_2); \\
j_4 &= hf(x_n + h, y_n + j_3, z_n + l_3); \\
l_4 &= hg(x_n + h, y_n + j_3, z_n + l_3);
\end{aligned}$$

То есть в начале нужно найти все коэффициенты $j_1, j_2, j_3, j_4, l_1, l_2, l_3, l_4$, подставив шаг h (изменение времени), концентрации (в первый раз начальные) $y_n = C_a^0 = 100, z_n = C_b^0 = 50$ и скорости реакций $k_1 = 2, k_2 = 2$. x_n – время t , однако, в с.2 от него зависимости нет, поэтому этот параметр в уравнении не учитывается. После найти первые изменённые концентрации в момент времени t ($t = h$ так как точкой отсчёта всегда будет 0), путём подстановки всех вычисленных коэффициентов, а также y_n и z_n в ур.3 и ур.4. Следующие концентрации ($t = 2h$) можно найти тем же путём, подставив вместо y_n и z_n предыдущие вычисленные концентрации.

1.3.3 Вывод по аналитическому и численному методам

Как было показано выше, аналитический метод предполагает интегрирование, последующие преобразования и упрощение уравнений. Конечно, формула, полученная путём такого решения, предельно точна, в ней нет места погрешности, однако её нахождение громоздко и требует знание основ интегрирования. Также при решении более сложных дифференциальных задач, знаний в области математики может не хватить и можно очень легко сделать ошибку, что неизбежно приведёт к неверным результатам. Как раз в этом случае прибегают к численным методам решения ДУ. С одной стороны, основным минусом этого решения является наличие погрешности в вычислениях, так как значения, полученные с его помощью, являются лишь близкими, аппроксимированными к значениям, полученным с помощью аналитического решения. С другой стороны, численные методы

значительно упрощают и ускоряют выполнение поставленной задачи, так как в них нет необходимости что-либо интегрировать или преобразовывать.

Наглядно разницу между этими двумя методами решения продемонстрирует Таблица 1, в которой представлены результаты, полученные при решении задачи двумя способами с исходными данными, соответствующие варианту.

Таблица 1 – Сравнение двух методов решения ДУ

t	Аналитическое решение		Численный метод Рунге-Кутты	
	C_a	C_b	C_a	C_b
0	100	50	100	50
0,1	91,76	58,24	91,76	58,24
0,2	86,23	63,77	86,24	63,76
0,3	82,53	67,47	82,53	67,47
0,4	80,05	69,95	80,05	69,95
0,5	78,38	71,62	78,39	71,61
0,6	77,27	72,73	77,27	72,73
0,7	76,52	73,48	76,52	73,48
0,8	76,02	73,98	76,02	73,98
0,9	75,68	74,32	75,68	74,32
1	75,46	74,54	75,46	74,54
1,1	75,31	74,69	75,31	74,69
1,2	75,21	74,79	75,21	74,79
1,3	75,14	74,86	75,14	74,86
1,4	75,09	74,91	75,09	74,91
1,5	75,06	74,94	75,06	74,94
1,6	75,04	74,96	75,04	74,96
1,7	75,03	74,97	75,03	74,97
1,8	75,02	74,98	75,02	74,98
1,9	75,01	74,99	75,01	74,99
2	75,01	74,99	75,01	74,99
2,1	75,01	74,99	75,01	74,99
2,2	75,00	75,00	75,00	75,00

Заметим, что использование метода Рунге-Кутты даёт довольно точные значения.

2. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

2.1 Краткое описание и блок-схема алгоритма

Алгоритм вычисления концентраций веществ А и В в момент времени t представлен на Рисунке 1. Более подробно он изложен в пункте описания числового метода решения. В качестве условия прекращения расчётов было решено использовать разницу между текущим и предыдущим концентрациями. То есть, если разница между ними становится больше некоторого значения (в этой программе было решено использовать число 0,005, так как при дальнейшем тестировании программы оно обеспечивало необходимую точность) для обоих веществ, то вычисление результатов прекращается.

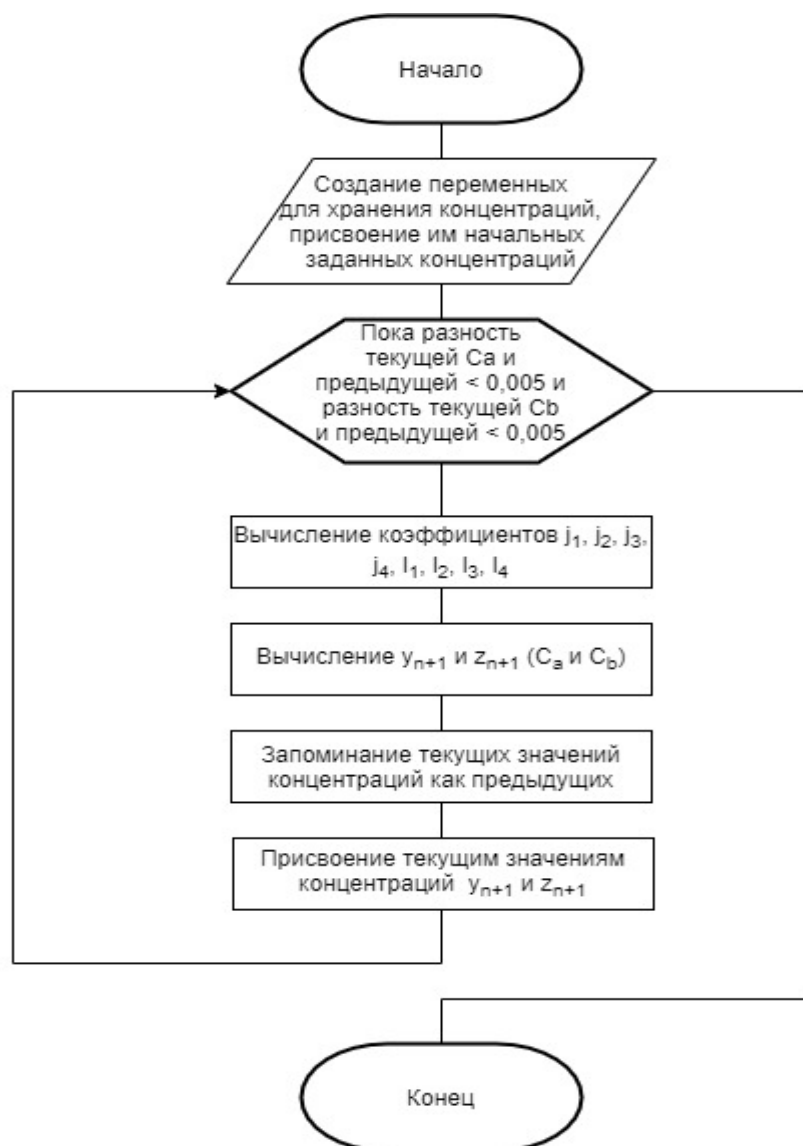


Рисунок 1 – Блок-схема работы программы

2.2 Обзор интерфейса программы

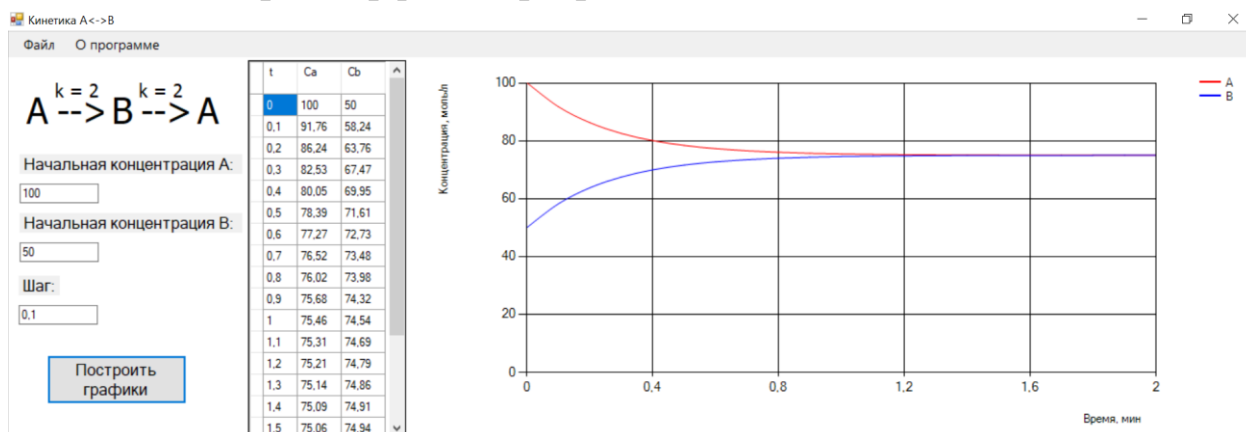


Рисунок 2 – Интерфейс программы

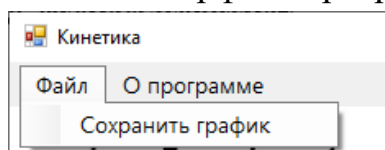


Рисунок 3 – Сохранение графика

В интерфейсе реализовано:

- Введение начальных концентраций веществ A и B
- Введение шага с ограничениями
- Построение графика изменения концентрации веществ
- Составление таблицы с расчетами
- Сохранения графика в файл

2.3 Тестирование программного комплекса, анализ результатов

Для проведения теста возьмём значения, данные из задания к практике, а именно концентрация вещества $A = 100, B = 50$. Возьмём шаг в 0.5 и получим следующее:

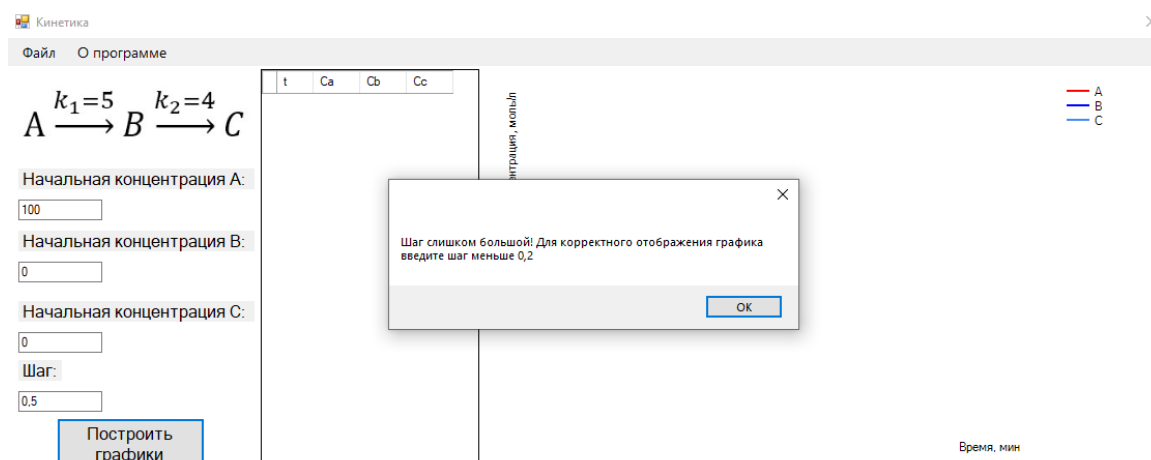


Рисунок 4 – Ошибка при построении графика

Так мы можем заметить, что для корректного отображения графика необходимо ввести шаг, значение которого будет меньше 0,2.

Произведем второе тестирование, где концентрация $A = 150$, $B = 100$, шаг 0,1. Так как данные были корректны, то программа успешно произведет все расчеты и выведет результаты на экран.

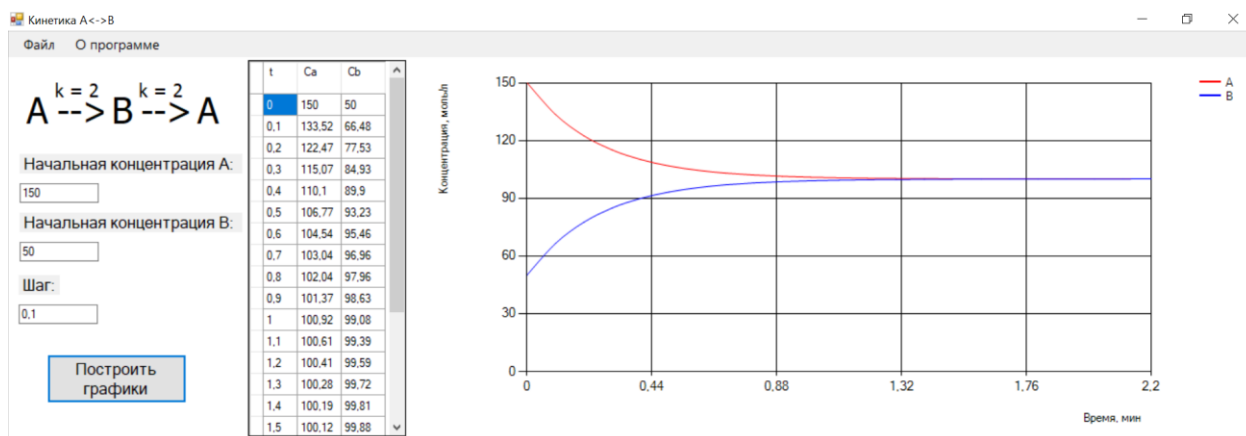


Рисунок 5 – Успешная работа программы

ВЫВОДЫ

Практическая работа была выполнена успешно в соответствии с полученным вариантом задания. Аналитическая часть работы содержит теоретическую информацию, необходимую для реализации задания, а также анализ различных языков программирования и сред разработок. Практическая часть работы содержит реализацию программы для решения задачи кинетики в соответствии с вариантом.

В результате прохождения учебной практики были приобретены новые знания и умения, развились навыки по поиску необходимой информации и ее обработке.

ПРИЛОЖЕНИЕ А

[--- Начало программы ---]

```
// Program.cs
using System;
using System.Windows.Forms;

namespace Kinetics
{
    static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new App());
        }
    }
}
```

```
// Configure.cs
using System;

namespace Kinetics
{
    class Configure
    {
        internal static double ToFixed(double number, uint decimals) //
Округления до заданного числа
        {
            string str = number.ToString("N" + decimals);
            return Convert.ToDouble(str);
        }
    }
}
```



```

// AllTests.cs
using System;
using System.Windows.Forms;

namespace Kinetics
{
    class AllTests
    {
        internal static bool AllChecks(double conA, double conB, double
conC, double step) // Все проверки значений
        {
            if (ValidInput() && AllFieldsFilled() && ValidValues(conA, conB,
conC, step))
            {
                return true;
            }
            return false;
        }

        internal static bool AllFieldsFilled() // Проверка заполненности
всех полей
        {
            for (int i = 1; i < (int)App.TextBoxes.step; i++)
            {
                string tempStr = Enum.GetName(typeof(App.TextBoxes), i) +
"_tb";
                if (Form.ActiveForm.Controls[tempStr].Text.Length == 0)
                {
                    MessageBox.Show("Заполните все поля!");
                    return false;
                }
            }
            return true;
        }

        internal static bool ValidValues(double conA, double conB, double
conC, double step) // Проверка корректности значений
        {
            if (step <= 0)
            {
                MessageBox.Show("Некорректное значение шага! Введите шаг
больше 0");
                return false;
            }

            if (step < 0.01)
            {
                MessageBox.Show("Шаг слишком маленький! Введите шаг больше
0,01");
                return false;
            }
        }
    }
}

```

```

    }

    if (step > 0.2)
    {
        MessageBox.Show("Шаг слишком большой! Для корректного
отображения графика введите шаг меньше 0,2");
        return false;
    }

    if (conA < 0 || conB < 0 || conC < 0)
    {
        MessageBox.Show("Концентрация не может быть отрицательной!");
        return false;
    }

    return true;
}

internal static bool ValidInput()
{
    for (int i = 1; i < (int)App.TextBoxes.step; i++)
    {
        string tempStr = Enum.GetName(typeof(App.TextBoxes), i) +
        "_tb";
        double.TryParse(Form.ActiveForm.Controls[tempStr].Text, out
double num);
        if (num == 0 && Form.ActiveForm.Controls[tempStr].Text != "0")
        {
            MessageBox.Show("Некорректные значения!");
            return false;
        }
    }
    return true;
}
}
}
}

```

```

// App.cs
using System;
using System.Linq;
using System.Windows.Forms;
using System.Windows.Forms.DataVisualization.Charting;

namespace Kinetics
{
    public partial class App : Form
    {
        internal enum Symbols
        {
            Backspace = 8
        }

        internal enum TextBoxes
        {
            conA = 1,
            conB,
            step
        }

        public App()
        {
            InitializeComponent();
        }

        private double func(double conA, double conB, int num) // расчёт
        производных f1,f2
        {
            double res = 0;
            switch(num){
                case 1:
                    res = -2 * conA + 2 * conB;
                    break;
                case 2:
                    res = 2 * conA - 2 * conB;
                    break;
            }
            return res;
        }

        private void build_button_MouseClick(object sender, MouseEventArgs
e)
        {
            dataTable.Rows.Clear();

            ConChart.Series[0].Points.Clear();
            ConChart.Series[1].Points.Clear();
        }
    }
}

```

```

try
{
    double.TryParse(conA_tb.Text, out double conA);
    double.TryParse(conB_tb.Text, out double conB);
    double.TryParse(step_tb.Text, out double step);

    if (AllTests.AllChecks(conA, conB, step))
    {

        ConChart.ChartAreas[0].AxisY.Minimum = 0;
        ConChart.ChartAreas[0].AxisY.Maximum = conA >= conB ? conA :
conB;

        double x = 0, ya = conA, yb = conB, ka1, ka2, ka3, ka4, kb1,
kb2, kb3, kb4, conAChck,
            conBChck, t = step;

        ConChart.Series[0].Points.AddXY(0, conA);
        ConChart.Series[1].Points.AddXY(0, conB);

        ConChart.Series[0].IsVisibleInLegend = true;
        ConChart.Series[1].IsVisibleInLegend = true;

        ConChart.Titles[0].Visible = true;
        ConChart.Titles[1].Visible = true;

        dataTable.Rows.Insert(0,
            0,
            Configurate.ToFixed(conA, 2),
            Configurate.ToFixed(conB, 2)
        );

        do
        {
            x = t;
            ka1 = step * func(conA, conB, 1);
            kb1 = step * func(conA, conB, 2);

            ka2 = step * func(conA + ka1 / 2, conB + kb1 / 2, 1);
            kb2 = step * func(conA + ka1 / 2, conB + kb1 / 2, 2);

            ka3 = step * func(conA + ka2 / 2, conB + kb2 / 2, 1);
            kb3 = step * func(conA + ka2 / 2, conB + kb2 / 2, 2);

            ka4 = step * func(conA + ka3, conB + kb3, 1);
            kb4 = step * func(conA + ka3, conB + kb3, 2);

            conAChck = conA;
            conBChck = conB;

```

```

        conA = conA + (ka1 + 2 * ka2 + 2 * ka3 + ka4) / 6;
        conB = conB + (kb1 + 2 * kb2 + 2 * kb3 + kb4) / 6;

        dataTable.Rows.Add(Configurate.ToFixed(x, 2),
            Configurate.ToFixed(conA, 2),
            Configurate.ToFixed(conB, 2)
        );

        t += step;
    } while (Math.Abs(conAChck - conA) > 0.005 &&
Math.Abs(conBChck - conB) > 0.005);

    ConChart.ChartAreas[0].AxisX.Minimum = 0;
    ConChart.ChartAreas[0].AxisX.Maximum = x;

    for (int i = 0; i < dataTable.RowCount; i++)
    {
        ConChart.Series[0].Points.AddXY(dataTable[0, i].Value,
dataTable[1, i].Value);
        ConChart.Series[1].Points.AddXY(dataTable[0, i].Value,
dataTable[2, i].Value);
    }
    ConChart.Visible = true;
}
}
catch
{
    MessageBox.Show("Введены некорректные значения!");
}
}

private void step_tb_KeyPress(object sender, KeyPressEventArgs
e)// Проверка ввода для шага
{
    char ch = e.KeyChar;
    if (!Char.IsDigit(ch) && ch != (int)Symbols.Backspace && ch !=
',')
    {
        e.Handled = true;
        MessageBox.Show("Введите шаг в формате 0,1");
    }
}

private void оПрограммеToolStripMenuItem_Click(object sender,
EventArgs e)
{
    MessageBox.Show("Данная программа предоставляет решения прямой
задачи кинетики обратимой реакции A<->B.\n©Зобнин Илья 485 гр.");
}

```

```

        private void сохранитьToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            if (ConChart.Visible)
            {
                SaveFileDialog saveFileDialog = new SaveFileDialog
                {
                    Filter = "PNG Image|*.png",
                    Title = "Сохранить как png",
                    FileName = "Sample.png"
                };

                DialogResult result = saveFileDialog.ShowDialog();
                saveFileDialog.RestoreDirectory = true;

                ConChart.SaveImage(saveFileDialog.FileName,
ChartImageFormat.Png);
            }
            else
            {
                MessageBox.Show("Сперва необходимо построить график!");
            }
        }
    }
}
[--- Конец программы ---]

```