**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«Национальный исследовательский университет ИТМО»**

**(Университет ИТМО)**

**Факультет: Инфокоммуникационных технологий**

**Образовательная программа: Инфокоммуникационные системы**

**Направление подготовки (специальность): Прикладное программирование в инфокоммуникационных системах**

О Т Ч Ё Т

по лабораторным работам

по дисциплине «Создание ПО инфокоммуникационных систем»

Выполнили: **студенты группы K34211**

**Дарзнек Лариса Викторовна**

**Логвинов Лев Анатольевич**

Проверил: **Осипов Никита Алексеевич**

Санкт-Петербург

2023

**Цель работы:**

Изучить методы разработки настольных клиентских Windows-приложений с использованием языка C# и Windows Forms.

**Ход работы:**

**Лабораторная работа №1 “Введение в разработку форм”**

1. Настроена прямоугольная форма Windows. (Рис. 1)

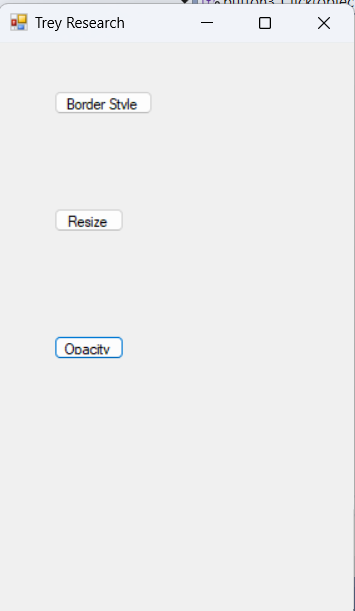


Рисунок 1 – Прямоугольная форма

1. Создана непрямоугольная форма. (Рис. 2)

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 2 – Непрямоугольная форма

1. Создана наследуемая форма. (Рис. 3)

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 3 – Наследуемая форма

1. Создано MDI-приложение. (Рис. 4)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4 - MDI-приложение

1. Создана овальная форма с возможностью закрытия во время выполнения. (Рис. 5)

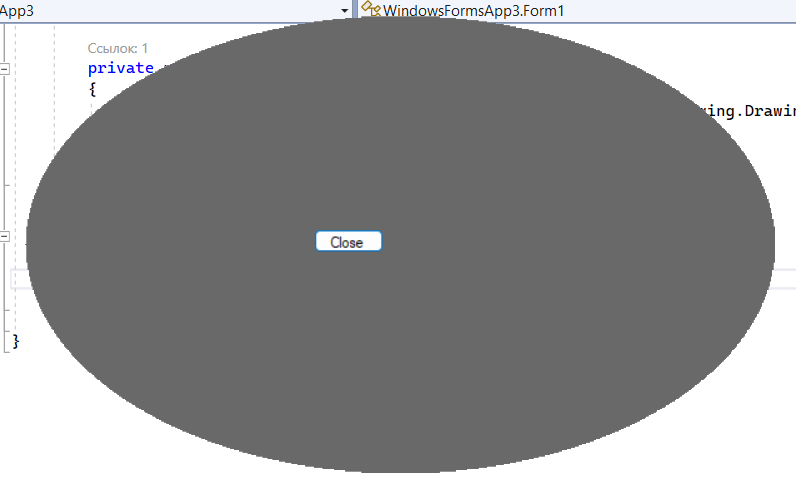


Рисунок 5 – Овальная форма

1. Создано приложение с двумя формами. (Рис. 6)

Изображение выглядит как текст, визитная карточка

Автоматически созданное описание

Рисунок 6 – Ромбовидная форма

**Лабораторная работа №2 “** **Работа с элементами управления”**

* + - 1. Создано простое приложение, отслеживающее события мыши, которые происходят у конкретного элемента управления. (Рис. 7)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 7 - Обработка событий Click и MouseMove

* + - 1. Создано приложения, использующее список. (Рис. 8)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 8 – Приложение со списком

* + - 1. Создан и использован элемент управления ToolStrip. (Рис. 9)

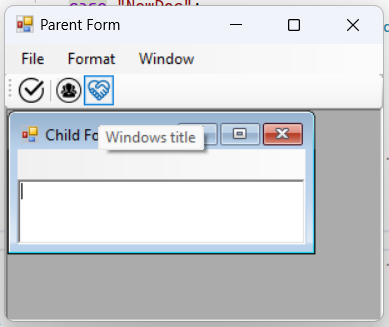


Рисунок 9 – Использование ToolStrip

* + - 1. Использован элемент StatusStrip. (Рис. 10)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 10 – Использование StatusaStrip

* + - 1. Использованы контейнерные элементы управления. (Рис. 11)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 11 – Контейнерные элементы управления

* + - 1. Реализован простой интерфейс ввода-вывода. (Рис. 12)

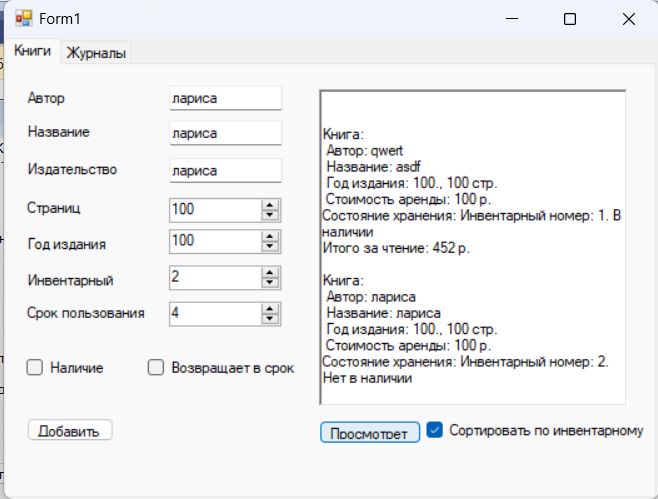


Рисунок 12 – Интерфейс ввода-вывода

1. Реализовано добавление и удаление элементов управления в режиме работы приложения (Рис. 13). Также было применено событие Validating (Рис. 14) и элемент ErrorProvider (Рис. 15).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 13 – Добавление элементов во время работы приложения

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 14 – Событие Validating

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 15 – ErrorProvider

8. Реализована вкладка для ввода журналов. (Рис. 16)

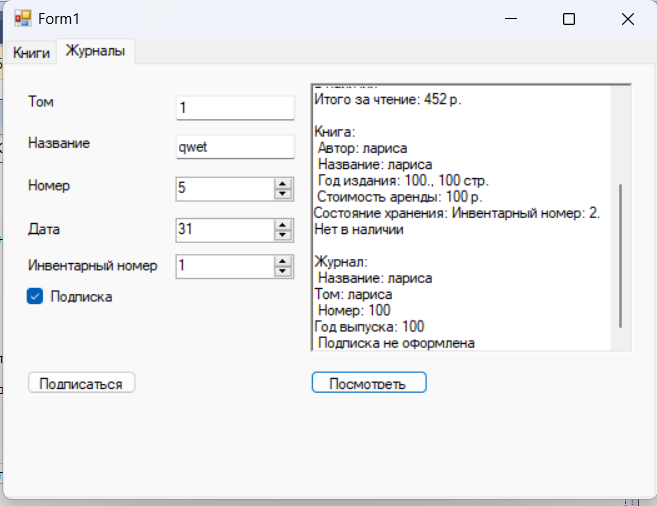


Рисунок 16 – Ввод журналов

**Лабораторная работа №3 “** **Создание элементов управления”**

* + - 1. Создан составного элемента управления UserControlTimer. (Рис. 17)

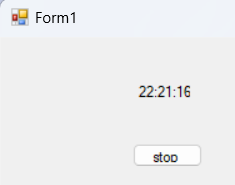


Рисунок 17 – UserControlTimer

* + - 1. Создан специализированного элемента управления UserControlTimer2. (Рис. 18)

Изображение выглядит как прямоугольный

Автоматически созданное описание

Рисунок 18 - UserControlTimer2

* + - 1. Создан расширенных элементов управления ClickButton. (Рис. 19)

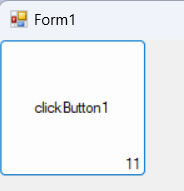


Рисунок 19 – ClickButton

* + - 1. Создана форма валидации данных. (Рис. 20)

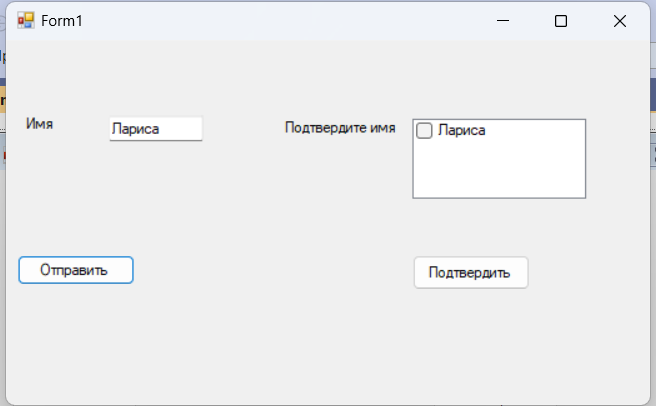


Рисунок 20 – Форма валидации данных

**Лабораторная работа №4 “Использование окон диалога”**

Использован компоне

Оглавление

нт SaveFileDialog. (Рис. 21)

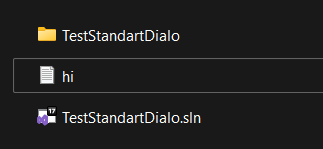


Рисунок 21 – Сохранённый файл

Использованы компоненты ColorDialog и FontDialog. (Рис. 22)

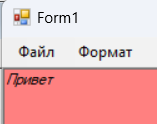


Рисунок 22 – Изменение цвета фона и шрифта

Использован компонент OpenFileDialog. (Рис. 23)

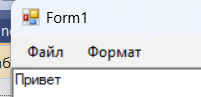


Рисунок 23 – Открытие файла из задания 1

Создано пользовательское модальное диалоговое окно. (Рис. 24)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 24 – Модальное окно

* + - 1. Реализовано взаимодействие компонентов формы и класса приложения. (Рис. 25)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 25 – Вывод списка

* + - 1. Создано немодальное окно диалога. (Рис. 26)

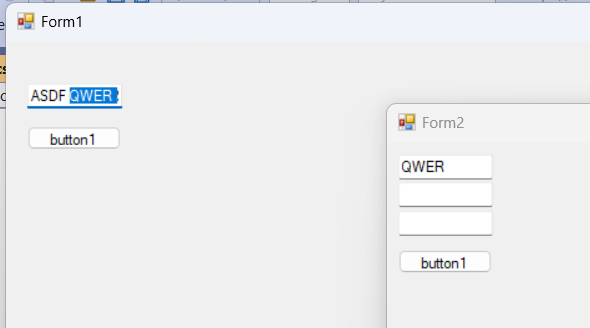


Рисунок 26 – Немодальное окно

**Лабораторная работа №5 “Взаимодействие управляемого и неуправляемого кода”**

Реализовано получение данных с использованием библиотеки объектов Microsoft Excel.

Реализовано использование COM-компонента для создания PDF-приложения.

Реализован вызов функции API. (Рис. 27)

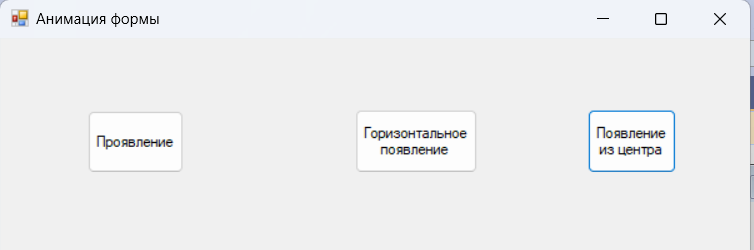


Рисунок 27 – Реализация анимации

Реализовано программное создание документа Microsoft Word.

**Лабораторная работа №6 “** **Графика и анимация в формах”**

1. Реализован вывод графика функции. (Рис. 28)

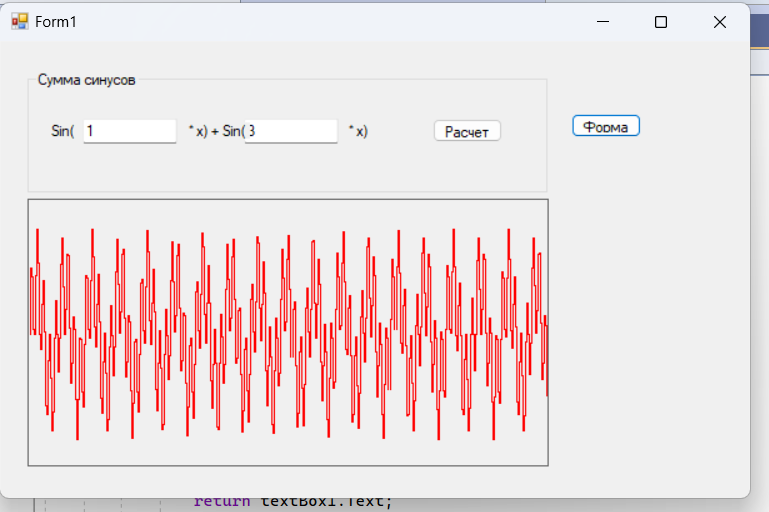


Рисунок 28 – График функции

2. Реализована анимация. (Рис. 29)

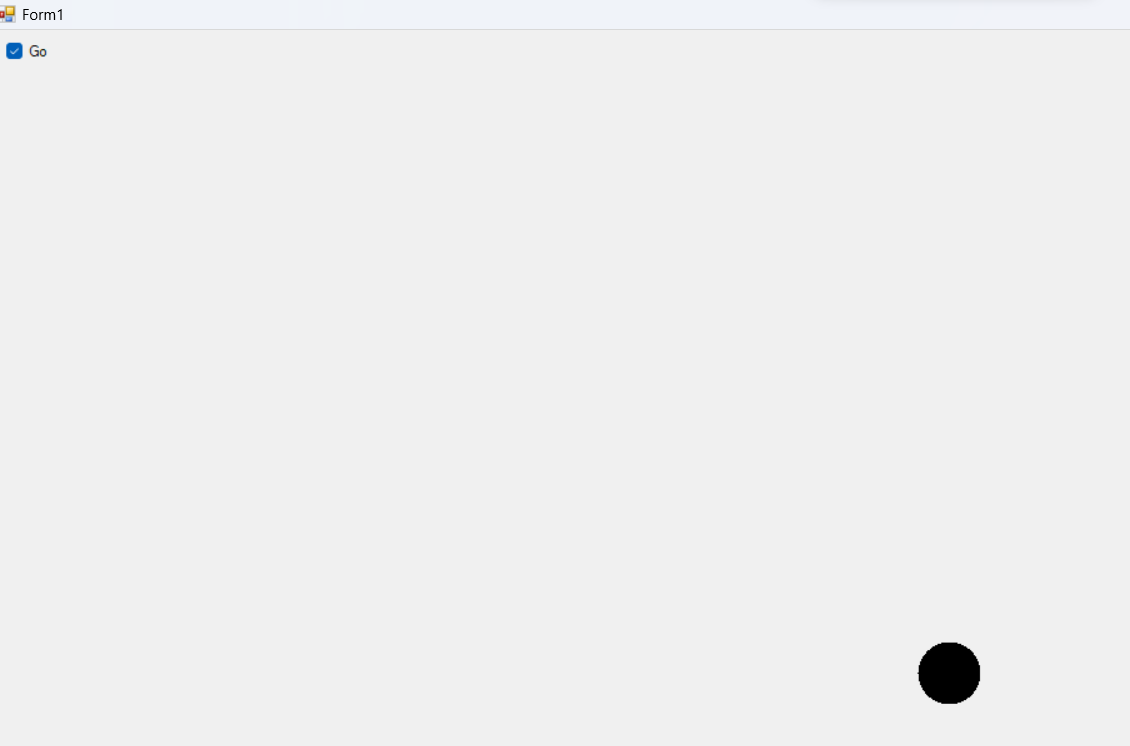


Рисунок 29 – Анимация

1. Использован компонент Chart. (Рис. 30)

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 30 – Отображение данных

**Лабораторная работа №7 “Асинхронное программирование”**

1. Реализована работа с компонентом BackgroundWorker. (Рис. 31)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 31 – Фоновый процесс

1. Использованы делегаты. (Рис. 32)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 32 – Фоновый процесс

1. Реализован асинхронный запуск произвольного метода. (Рис. 33)

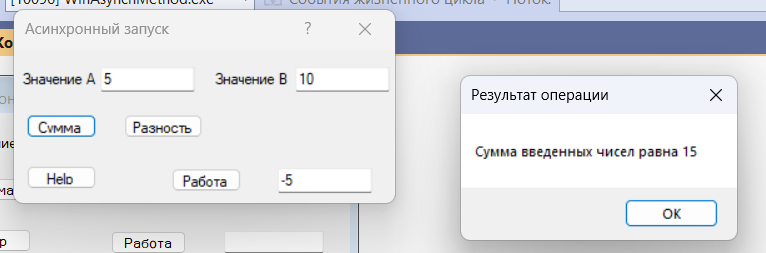


Рисунок 33 – Выполнение метода с использованием делегата

1. Реализован алгоритм во вторичном потоке. (Рис. 34)

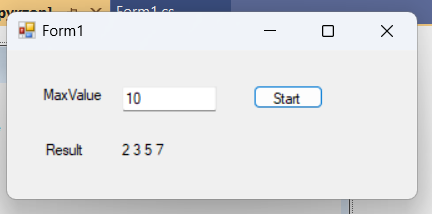


Рисунок 34 – Вывод последовательности чисел по алгоритму Эратосфена

**Лабораторная работа №8 “** **Повышение удобства использования приложений”**

1. Использована сериализация для сохранения коллекции объектов. (Рис. 35, 36)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 35 – Содержимое файла PersonCollection.xml.

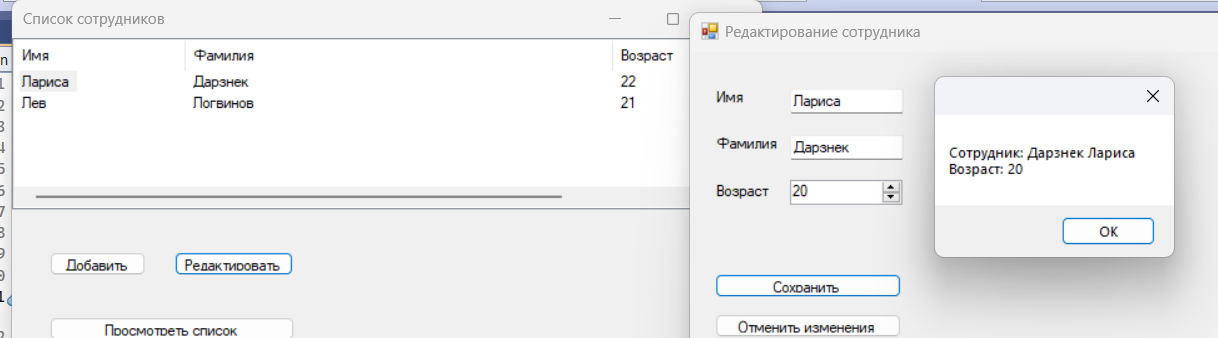


Рисунок 36 – Значение Age не сохранено

1. Создана контекстная справка. (Рис. 37)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 37 – Контекстная справка

1. Реализован автоматический выбор языка при запуске приложения. (Рис. 38)

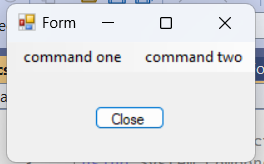


Рисунок 38 – Автоматическое определение языка

**Итоговое контрольное задание**

Реализован калькулятор в соответствие с требованиями, перечисленными в лабораторной работе. (Рис. 39, 40, 41)

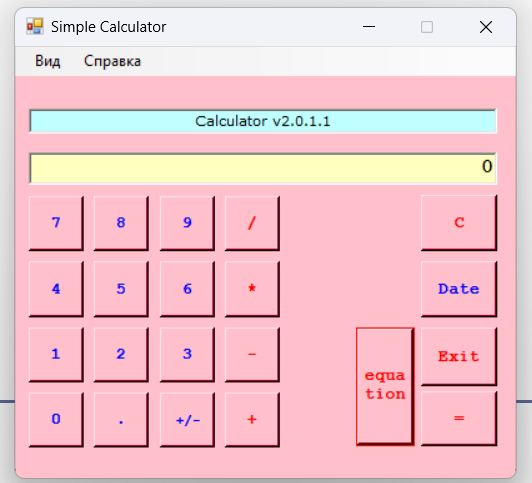


Рисунок 39 – Обычный калькулятор

Изображение выглядит как текст, электроника

Автоматически созданное описание

Рисунок 40 – Инженерный калькулятор

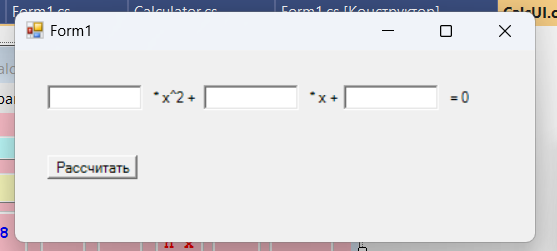


Рисунок 41 – Ввод коэффициентов квадратного уравнения

**Вывод:** в результате выполнения лабораторных работ были изучены методы разработки настольных клиентских Windows-приложений с использованием языка C# и Windows Forms.