Министерство образования и науки Российской Федерации

ФГАОУ ВО «КФУ им. В.И. Вернадского»

Физико-технический институт

|  |
| --- |
|  |

Кафедра компьютерной инженерии и моделирования

Михайлусов Анатолий Михайлович

отчет по практической работе №1  
по дисциплине **«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»**

Направление подготовки:

09.03.04 "Программная инженерия"

Оценка -



Симферополь, 2023

**Практическая работа №1.   
Тема: Использование программных конструкций C#**

**Цель работы:** Научиться создавать простейшие консольные и WPF приложения на языке C# в среде Visual Studio, изучить возможность создания самодокументируемых приложений.

Научиться преобразовывать различные типы данных в C#, познакомиться с типом данных Decimal, научиться грамотно использовать циклы для итерационных вычислений с контролем погрешности, обрабатывать события нажатия клавиш, научиться использовать классы String, StringBuilder, научиться создавать самодокументируемые XML справочные файлы.

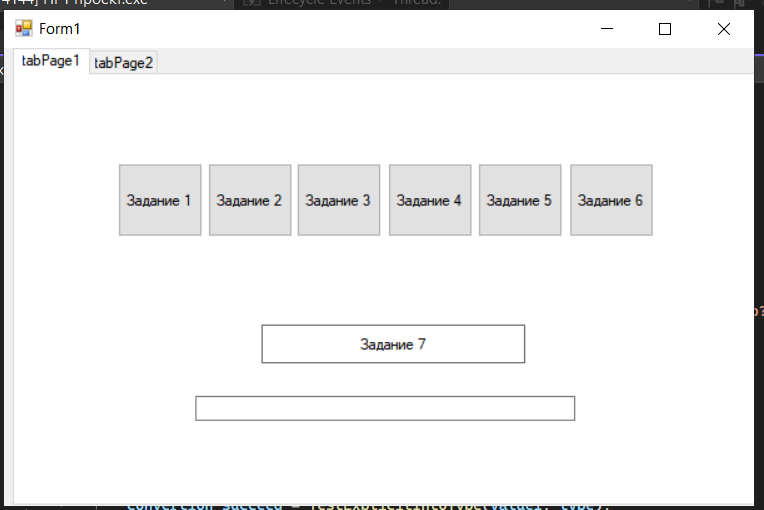
**Ключевые понятия:** интегрированная среда разработки IDE (Integrated Development Envirionment) Visual Studio, общеязыковая исполнительная среда CLR, библиотека классов FCL, общеязыковые спецификации CLS, решение (solution), проект (project), пространство имен (namespace), сборка (assembly), MSIL (Microsoft Intermediate Language, IL), управляемый код, двухэтапная компиляция, дизассемблер, обозреватель решений (Solution Explorer), Windows.Forms и WPF проекты, класс, статический тип, динамический тип, встроенные типы, типы-значения, ссылочные типы, фундаментальные типы: логический, символьный, целый, с плавающей точкой, void, указатели, ссылки, массивы, перечисления, структуры, классы, преобразования типов: упаковка Boxing, неявное преобразование, явное преобразование, класс Convert.

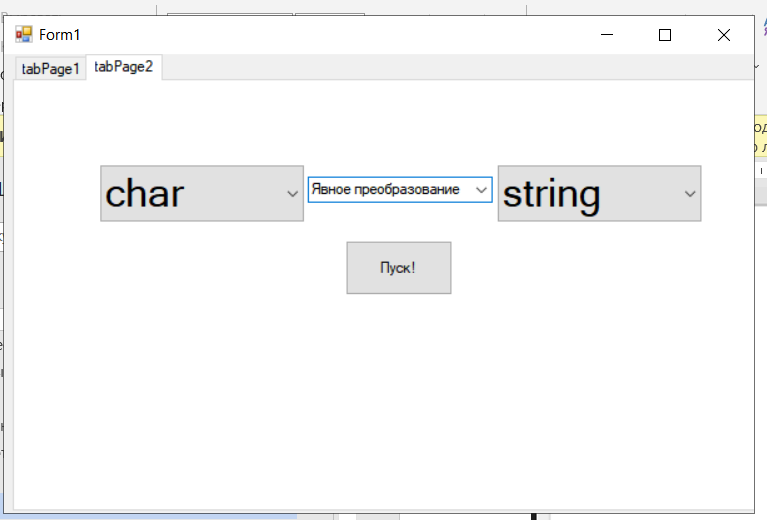
**Перед выполнением лабораторной работы изучена следующая литература:**

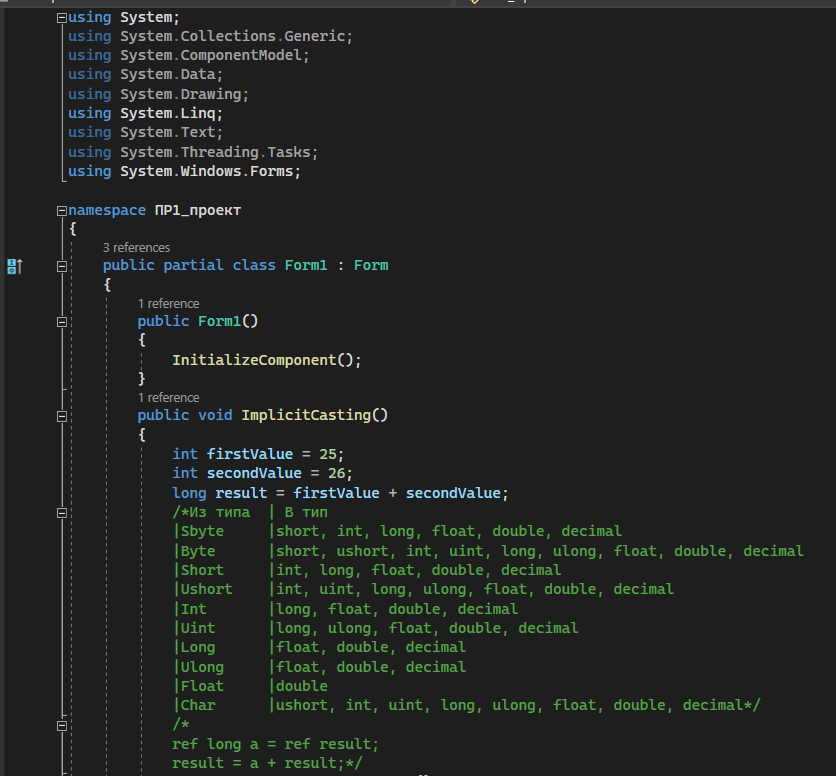
1. Презентация лектора курса: «Введение в C# и платформу .NET» (все материалы доступны в облаке на Mail.ru).
2. Прослушана видеолекция лектора курса.
3. Прослушано некоторое количество обучающих материалов C#.
4. Получены начальные сведения о спецификации языка C#.
5. Изучен туториал «Создание приложения Windows Forms на C# в Visual Studio» (<https://learn.microsoft.com/ru-ru/visualstudio/ide/create-csharp-winform-visual-studio?sa=X&ved=2ahUKEwjWg5vbyrroAhVXAJ0JHckQBzYQ9QF6BAgEEAI&view=vs-2022>)
6. Изучен Урок 5. Методы и функции в C# (<https://labs-org.ru/c-sharp5/>)
7. Раздел справочника по C# Ссылочные типы (<https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/reference-types>)
8. Статья «Объектно-ориентированное программирование. Наследование» (https://metanit.com/sharp/tutorial/3.7.php)

**Выполнены 4 задания, описанных в методических указания к выполнению лабораторных работ.**Задание 1: Приведение и преобразование типов

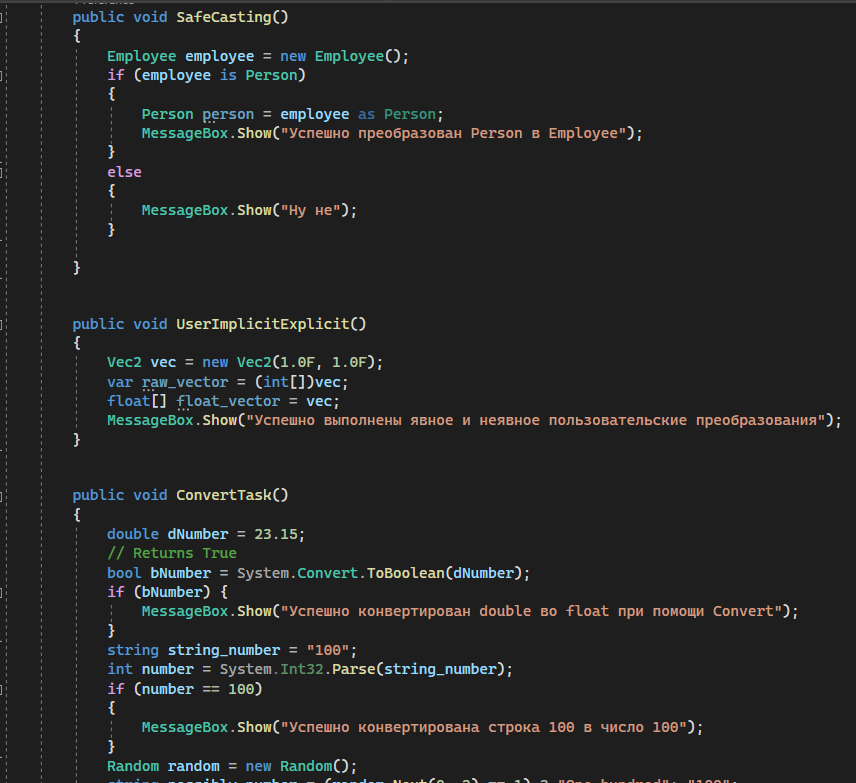
Скриншоты кода и интерфейса приведены ниже

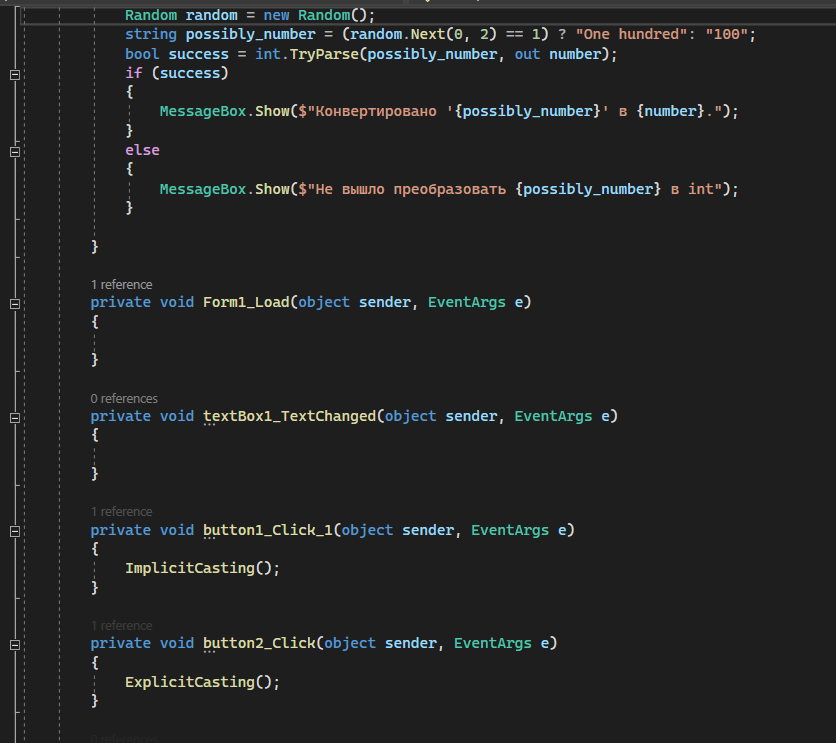


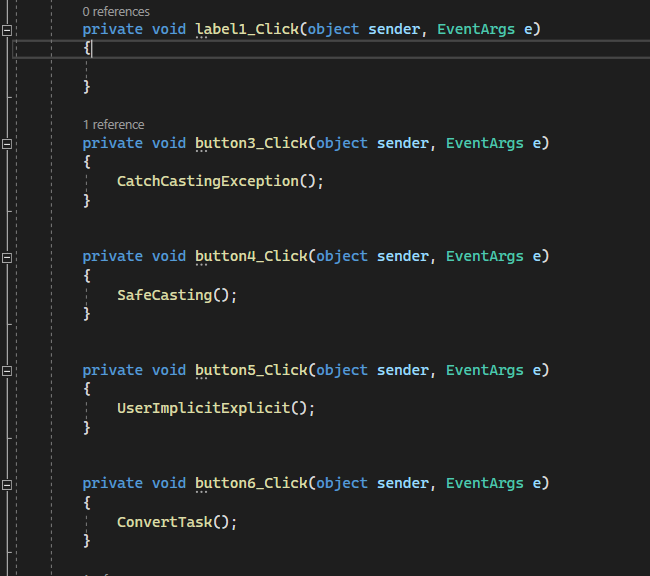




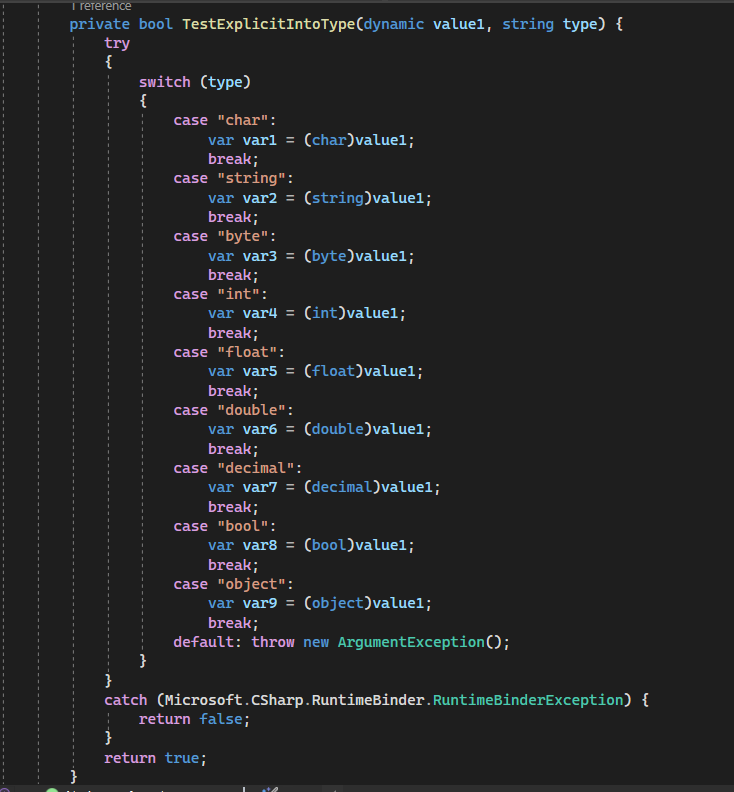


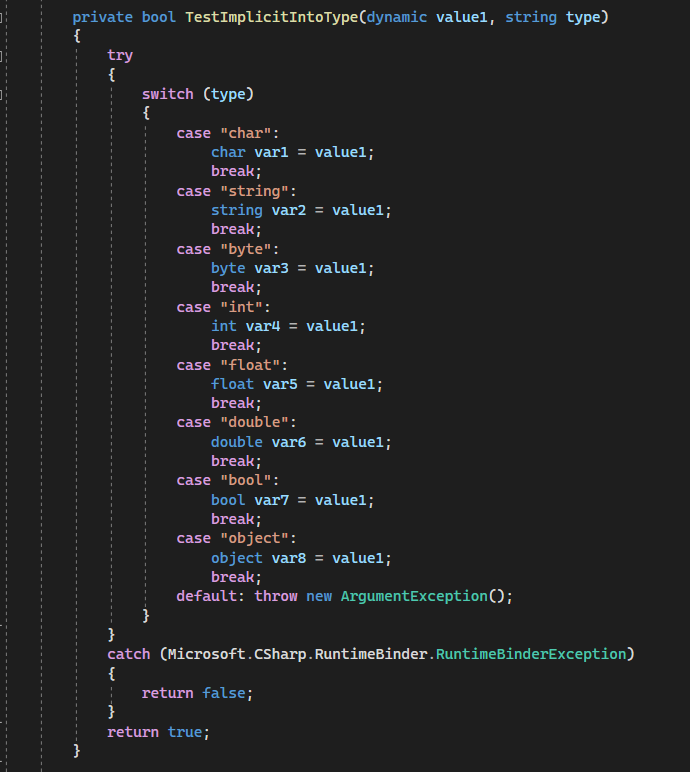


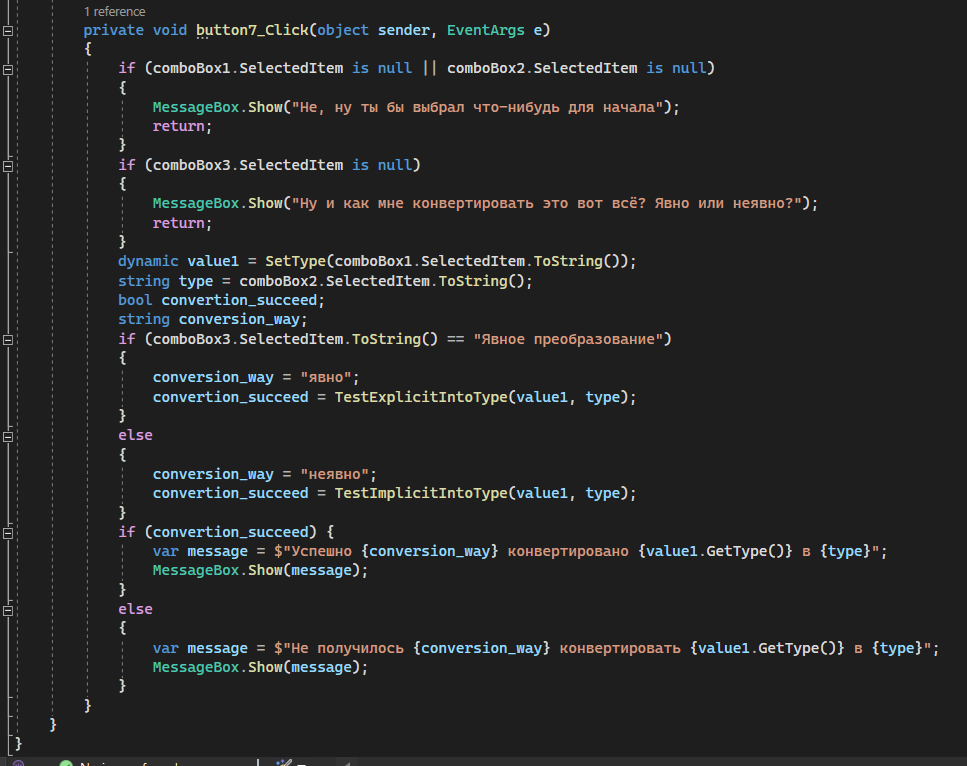




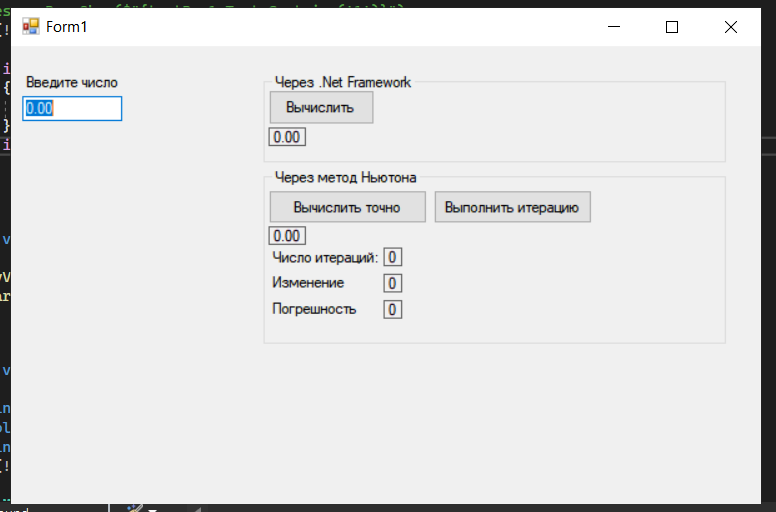


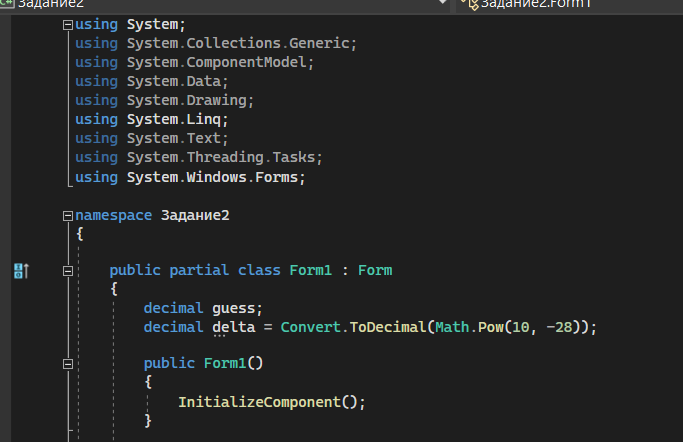


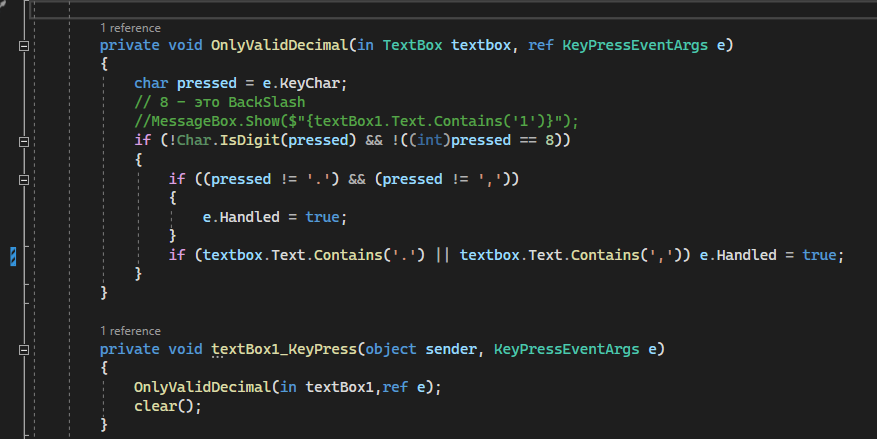


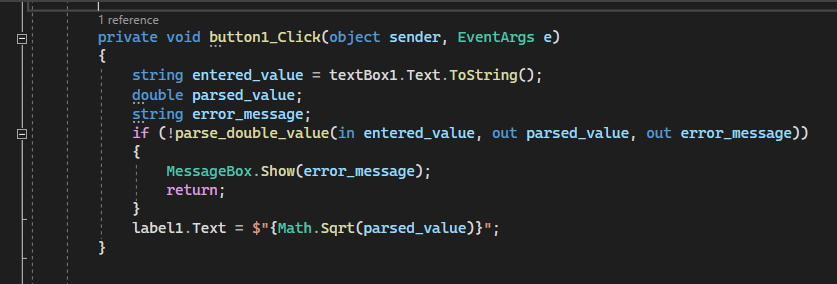
  
Задание 2: Вычислить квадратный корень с контролем точности

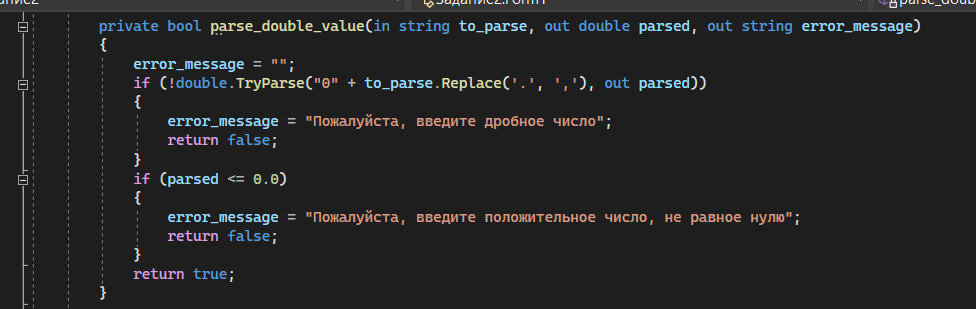
Скриншоты кода и интерфейса приведены ниже

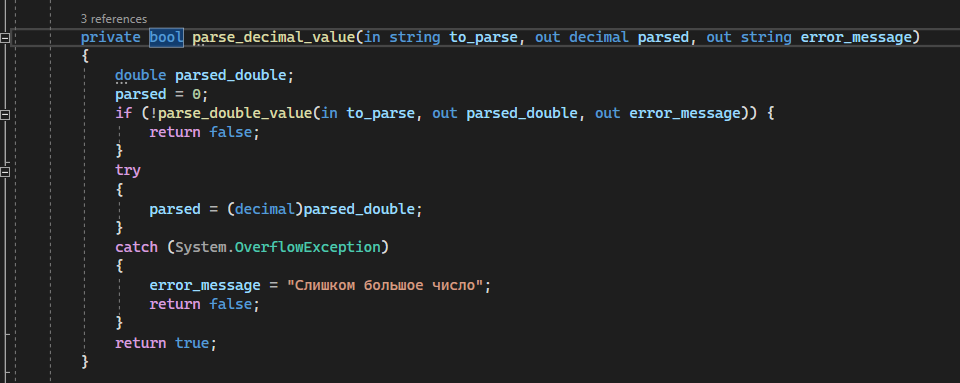


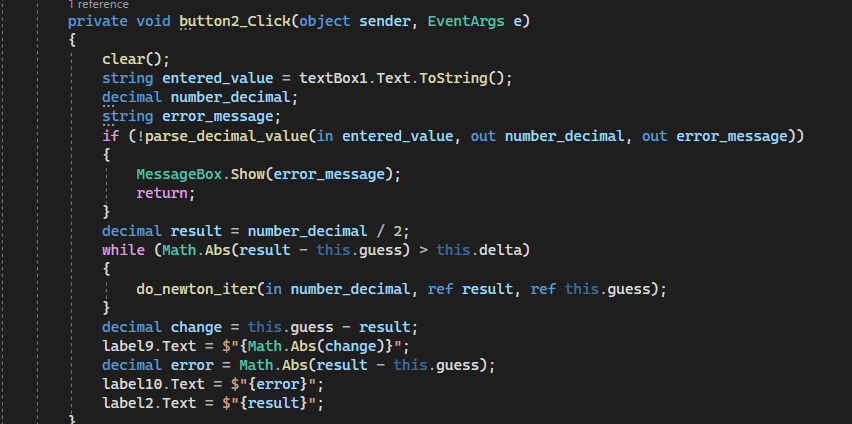


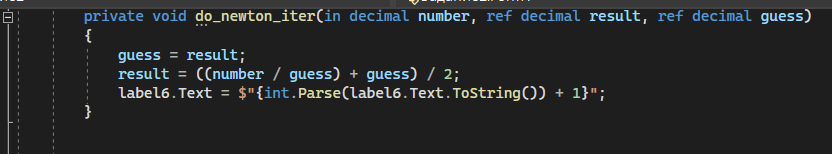


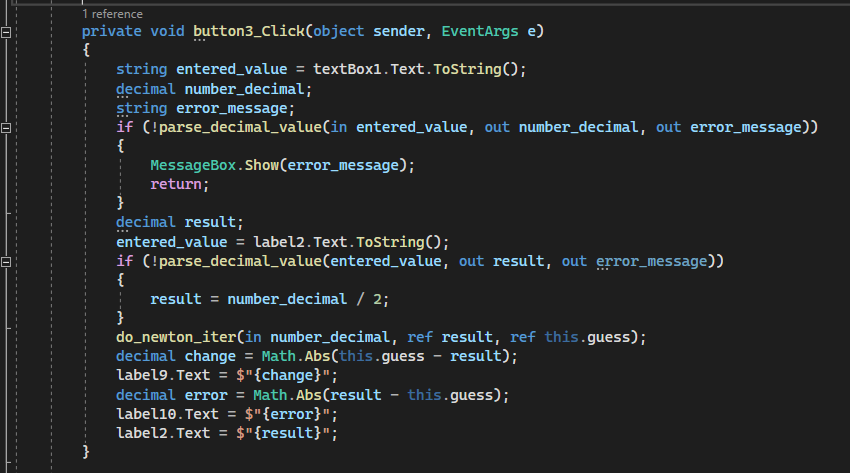


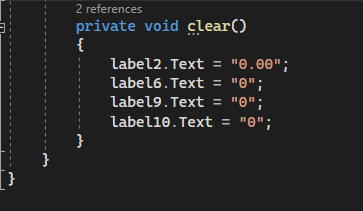




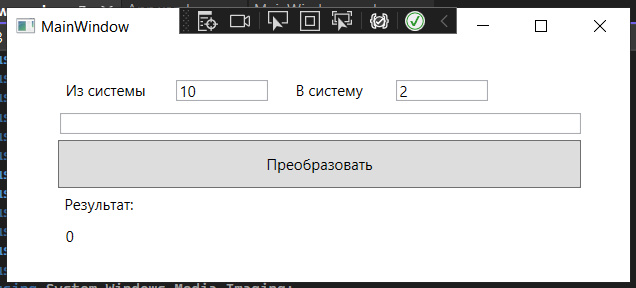


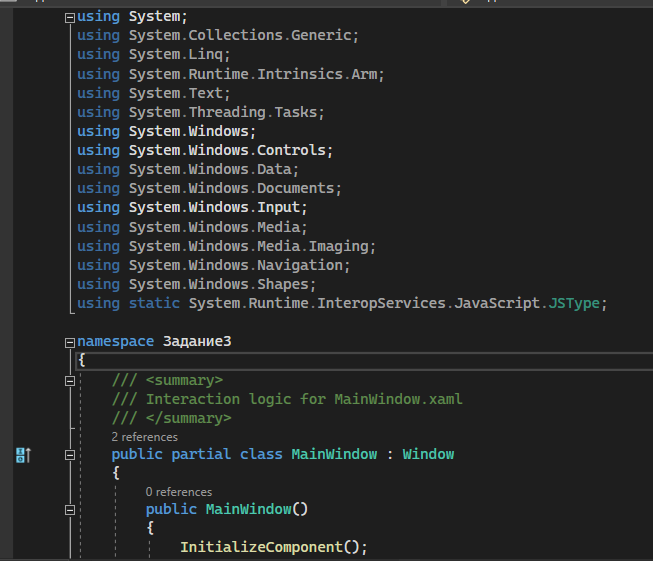


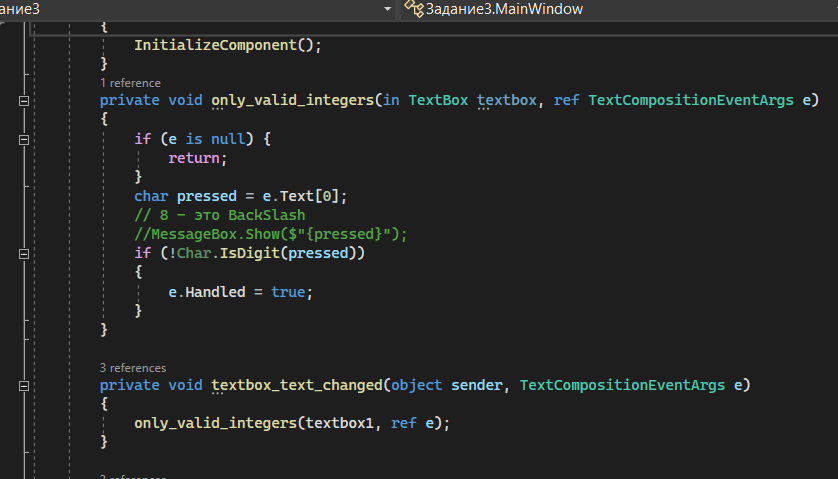


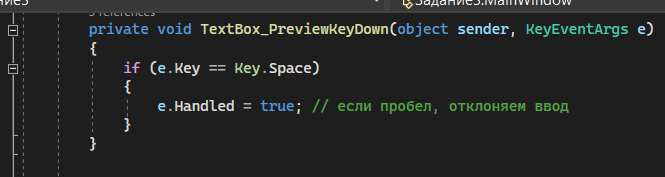
  
Задание 3: Конвертация целочисленных данных в бинарные.

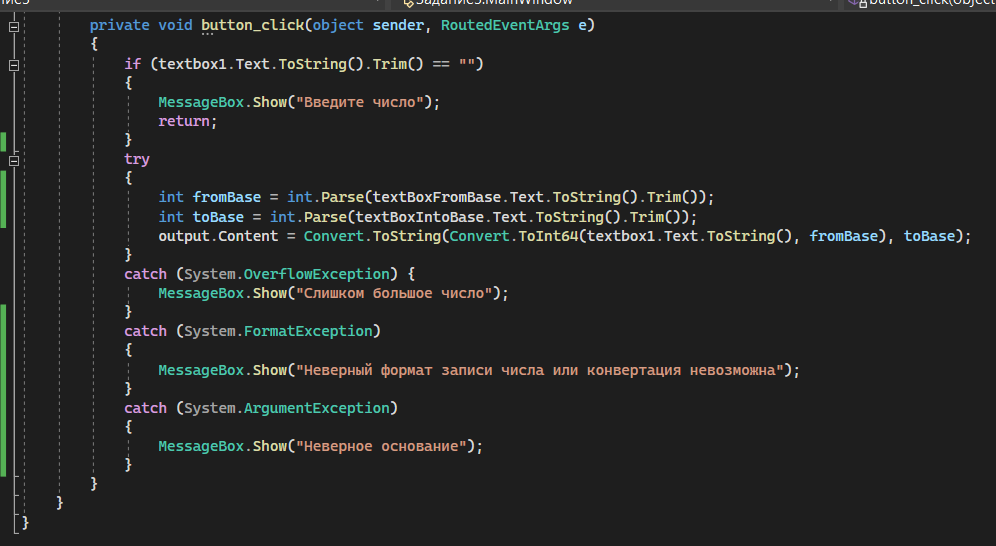
Скриншоты кода и интерфейса приведены ниже

****

****

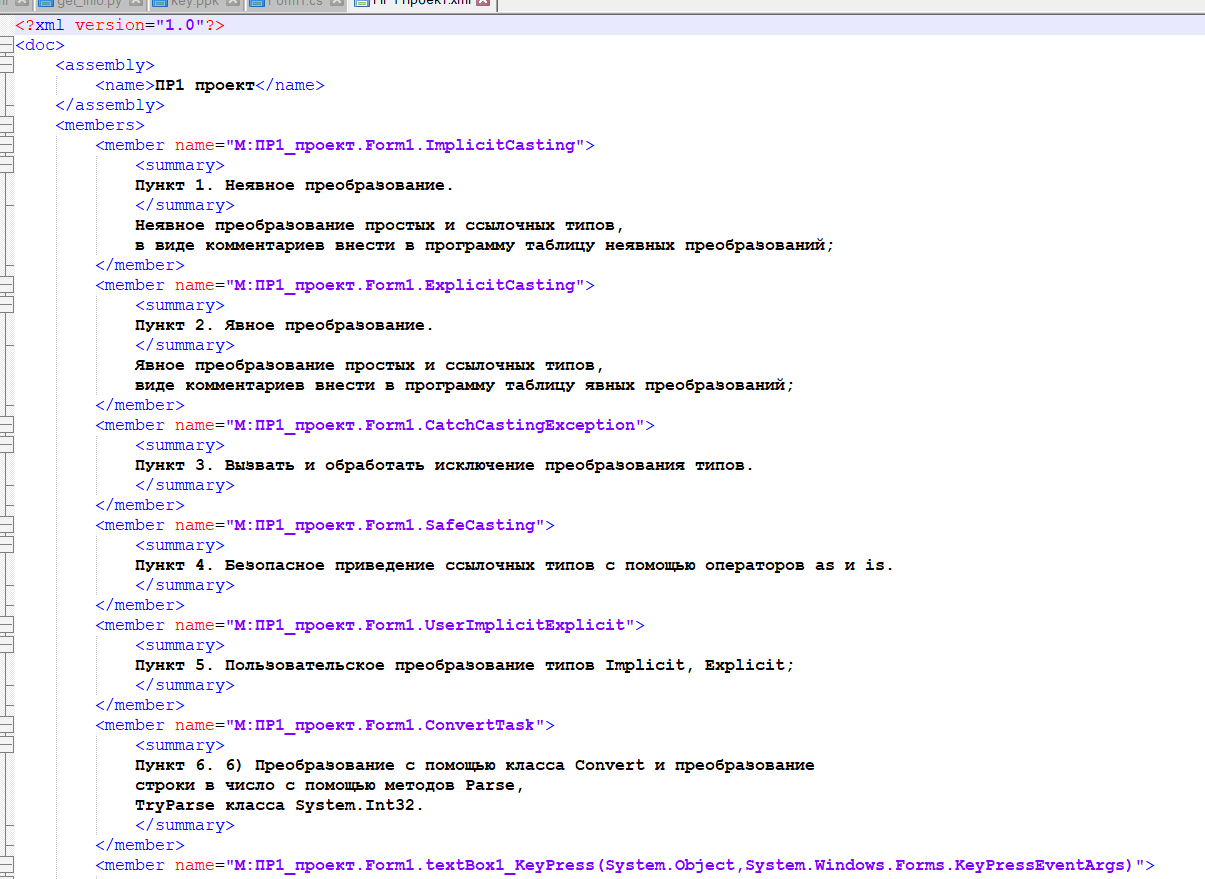
****

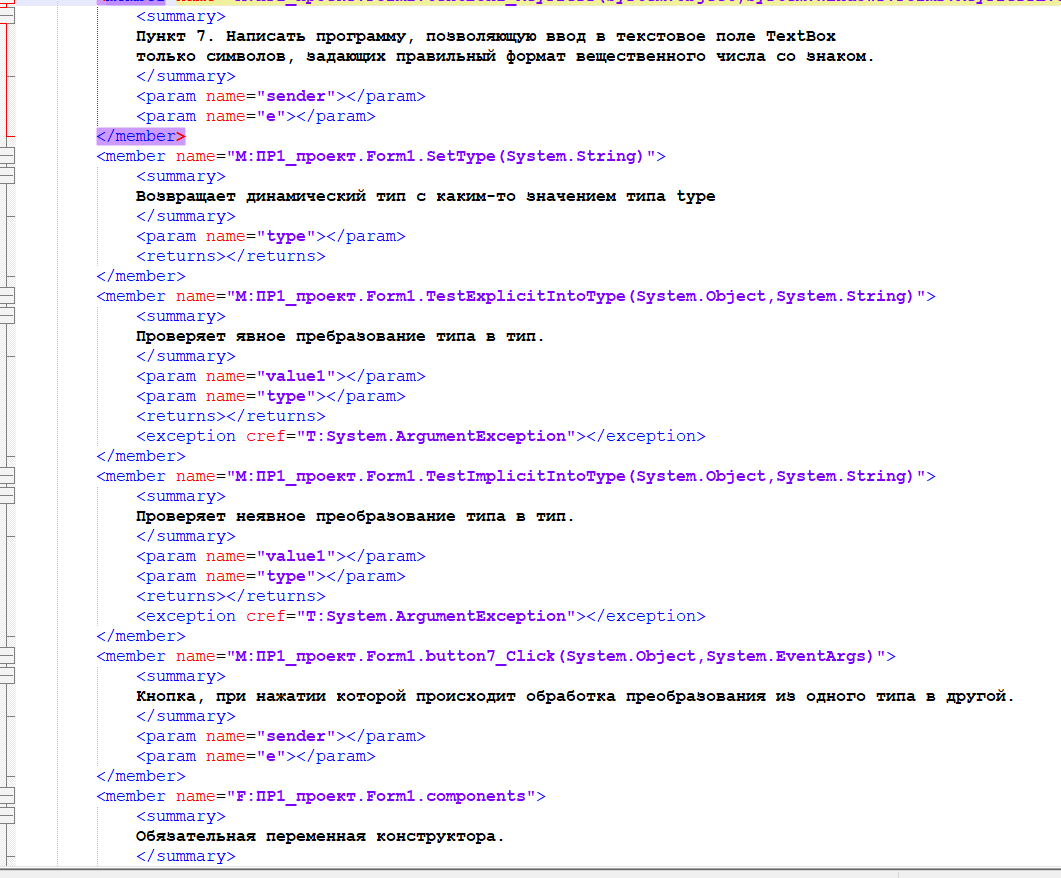
****

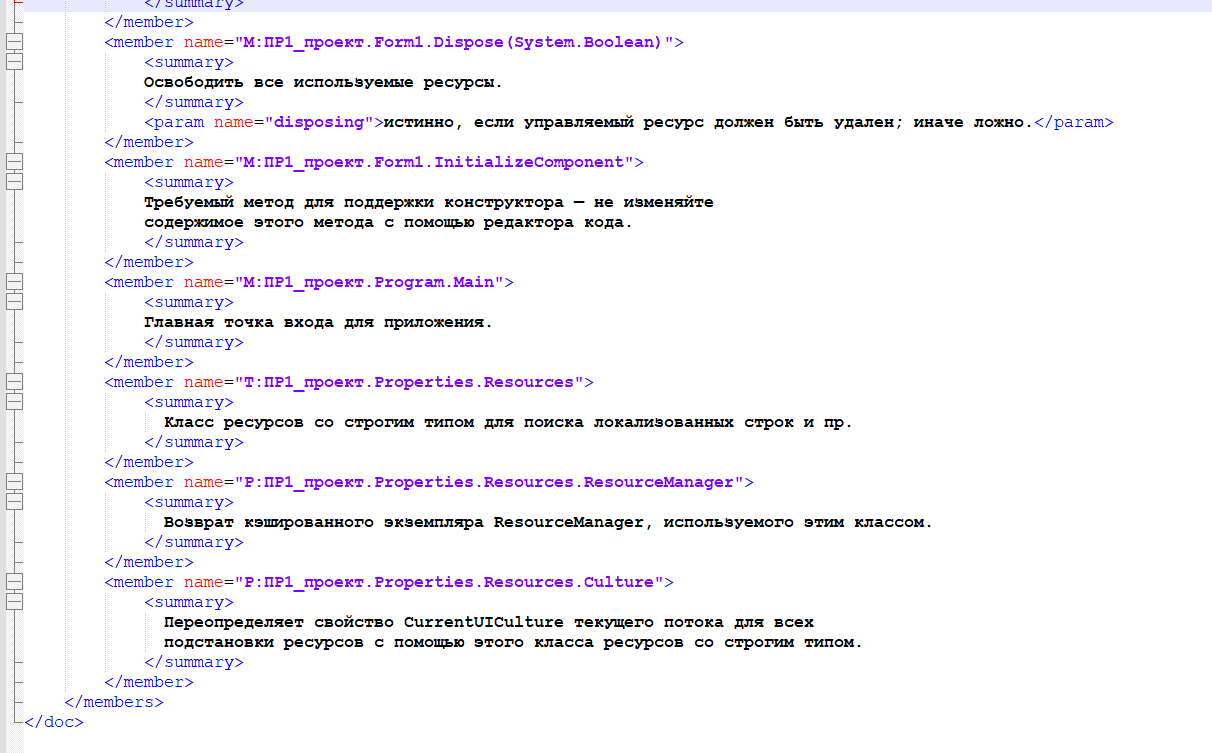
****

**Задание 4: Создание документации для приложения**

**XML файл:**

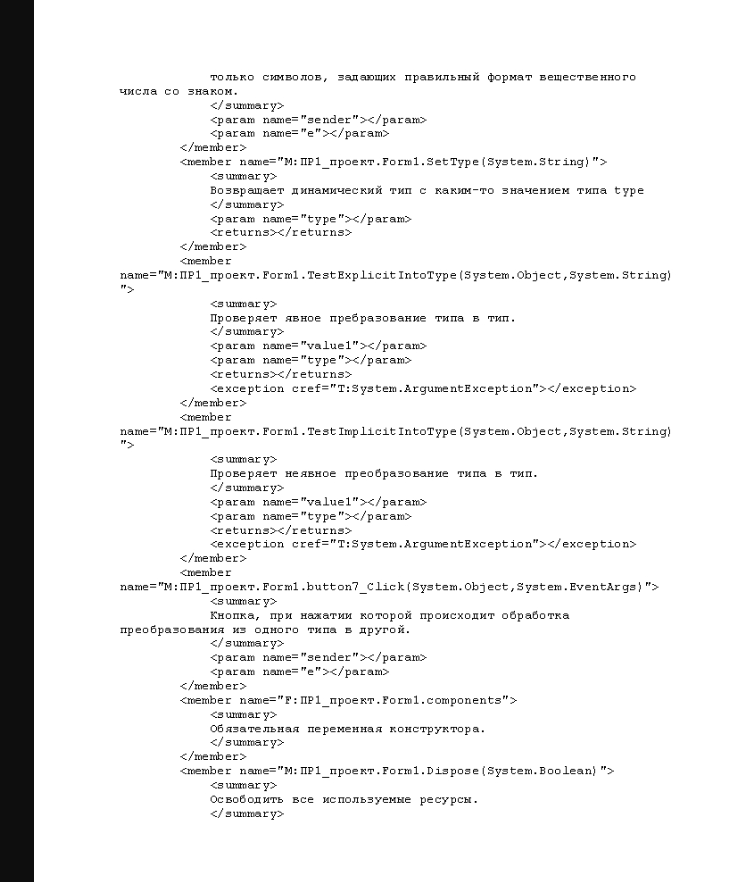
****

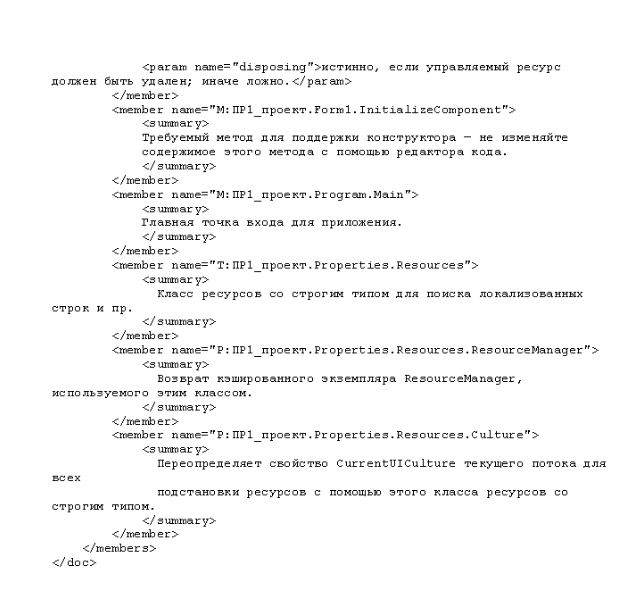
****

****

**CHM файлы:**

****

****

****

**Представлены 4 проекта, реализованных в Visual Studio Community 2022.  
Проекты представлены преподавателю в электронной форме, продемонстрирована их работоспособность, разъяснены детали программного кода.**

**Вопросы, заданные преподавателем:**

* 1. Что такое IDE и какие её преимущества?

IDE – Интегрированная среда разработки. Преимущества использования IDE заключаются в большем удобстве написания кода и разработки приложений, чем в обычных текстовых редакторах. Например, для приложений Dot Net IDE позволяет создавать формы через конструктор. В IDE можно гибко настраивать сборку проекта, производить быстро и легко рефакторинг огромных частей кода. Также, IDE предоставляет удобную навигацию по проекту, загрузку библиотек и многое другое.

* 1. Основные различия между WPF и Windows Forms.

Отличия WPF от Windows Forms:

* Она более новая и, соответственно, отвечает современным стандартам разработки
* Microsoft использует ее во многих своих приложениях, например Visual Studio
* Это более гибкая система, Вы можете сделать больше, без написания или покупки готовых элементов управления
* Если вдруг Вы решите воспользоваться готовыми решениями из сообщества, скорее всего сторонние разработчики будут сфокусированы именно на WPF, поскольку это более новая система
* С помощью XAML можно легко создавать и редактировать GUI, позволяя разделить работу дизайнера (XAML) и программиста (C#, VB.NET и др.)
* Привязка данных позволяет Вам еще лучше разделить данные и GUI
* Для лучшей производительности можно использовать аппаратное ускорение при отрисовке GUI
* Позволяет создавать GUI как для Windows приложений, так и для приложений Web (Silverlight/XBAP)
  1. Зачем нужен XML файл документации?
  2. Что такое Intelligence Sense в Visual Studio?
  3. Как осуществляется перенаправление ввода в командной строке?

Может ли быть потеряна точность в неявных преобразованиях из int, uint, long, или ulong в float и из long или ulong в double.

Точность может быть потеряна при преборазовании типа с большим диапазоном в тип с меньшим.

* 1. Как осуществить неявное преобразование в тип char.

Такого пребразования не существует.

* 1. Как осуществить неявные преобразования между типами с плавающей запятой и типом decimal?

Неявного преобразования из типов с плавающей запятой и типом decimal не существует.

Float можно неявно преобразовать в Double.

* 1. Всегда ли можно явно преобразовать int в sbyte, byte, short, ushort, uint или ulong?
  2. Как контролируется точность при вычислении квадратного корня методом Ньютона?
  3. Зачем в работе используется класс StringBuilder?

Ссылка на github (Исходный код всех работ)

<https://github.com/CyberTea0X/OOP.git>