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 №4  
по дисциплине **«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»**

Направление подготовки:

09.03.04 "Программная инженерия"

Оценка -



Симферополь, 2023

**Практическая работа №4.**

**Тема: Типы перечислений и структуры**

**Цель работы:** Научиться на практике создавать перечисления и структуры. Разобраться самостоятельно с эффективностью использования структур, недостатки и преимущества по сравнению с классами

**Ключевые понятия**: Enum, Struct, Nulable.

**Перед выполнением лабораторной работы рекомендуем:**

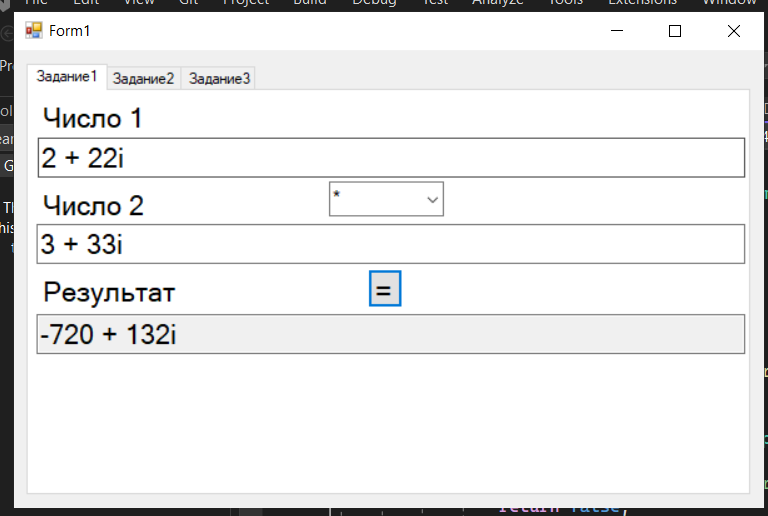
1. Изучить презентацию лектора курса: «Классы, структуры, конструкторы, модификаторы доступа в C# » (материалы доступны в облаке на mail.ru).
2. Сайт Metanit.com
3. Справочник по C#. Корпорация Microsoft.   
   http://msdn.microsoft.com/ru-ru/library/618ayhy6.aspx
4. Биллиг В.А. Основы программирования на C#. Интернет-университет информационных технологий. http://www.intuit.ru/studies/courses/2247/18/info
5. Павловская Т. Программирование на языке высокого уровня C#. http://www.intuit.ru/studies/courses/629/485/info
6. Руководство по программированию на C#. Корпорация Microsoft.   
   http://msdn.microsoft.com/ru-ru/library/67ef8sbd.aspx
7. Корпорация Microsoft. C#. Спецификация языка. (Приложение А, Комментарии к документации).

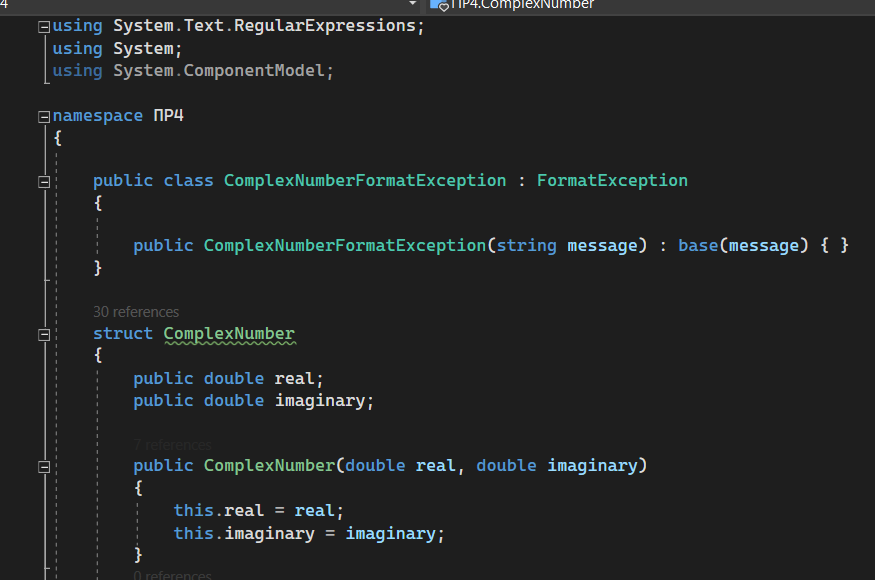
**Содержание отчета:**

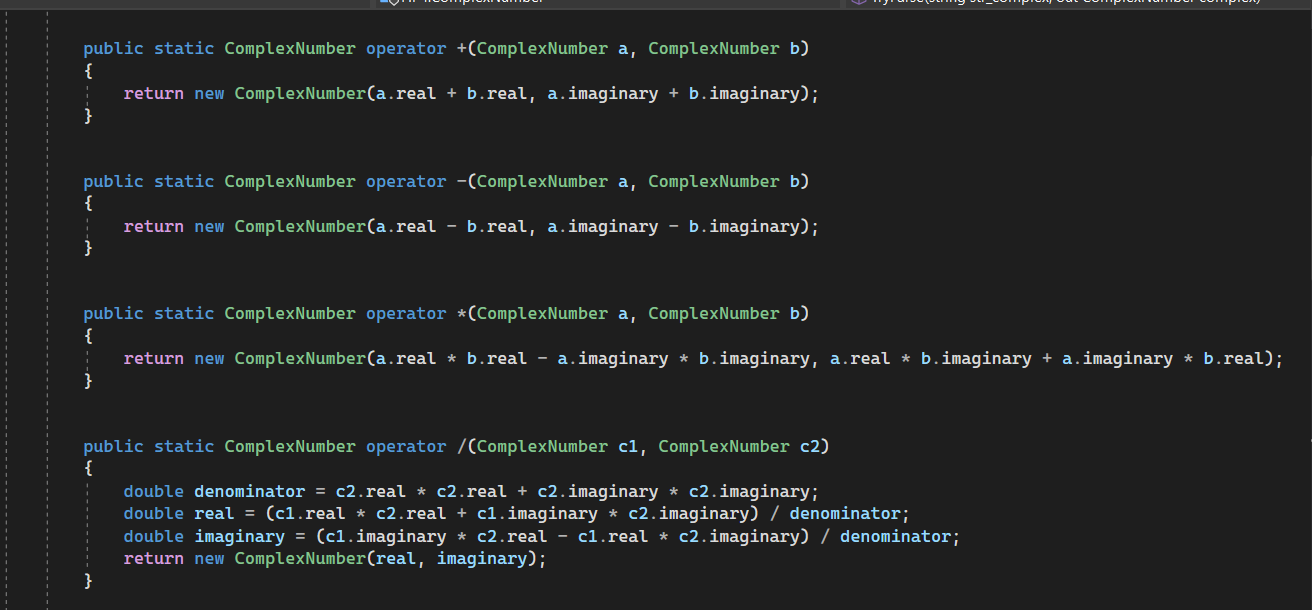
1. Описание и раскрытие ключевых понятий
2. Краткое описание изученной литературы

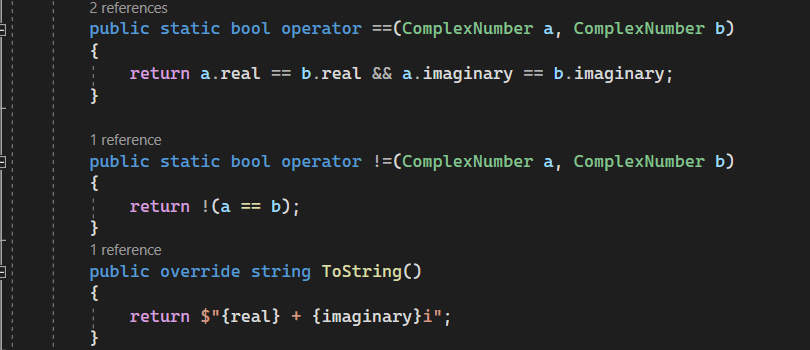
Программный код для каждого из заданий с подробными комментариями

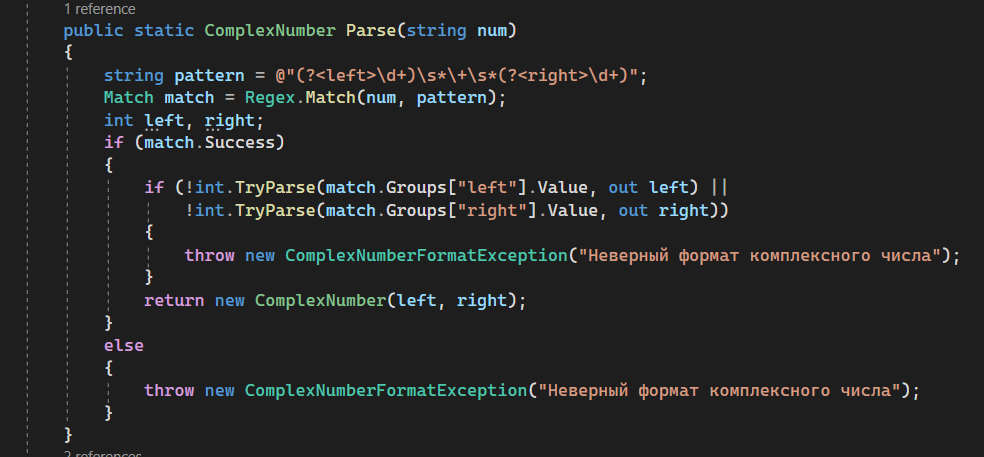
**Задание 1. Создание структур и перегрузка операторов.**

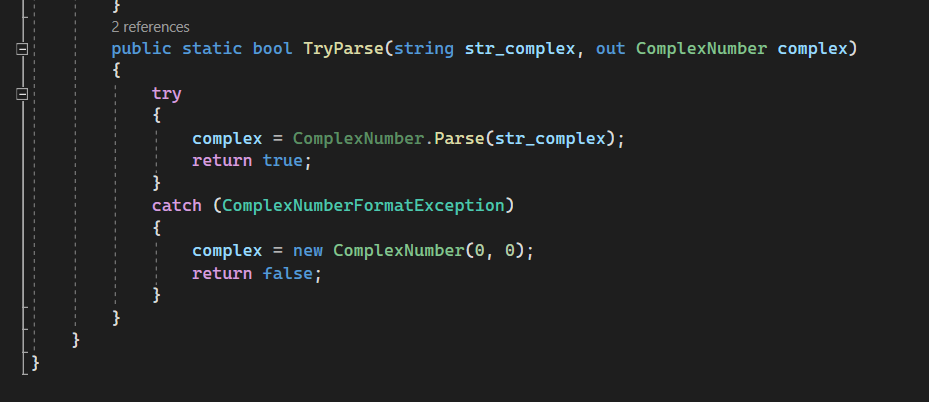




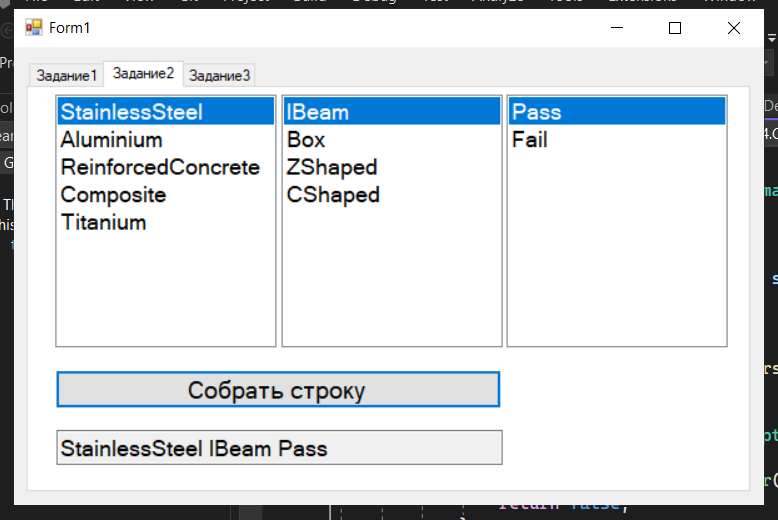


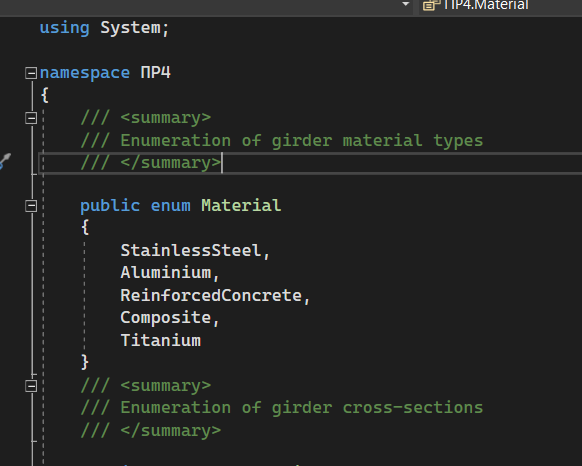


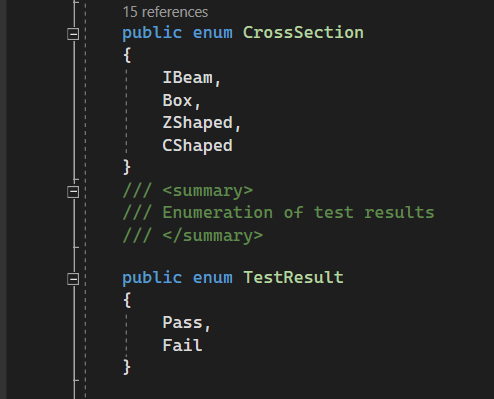


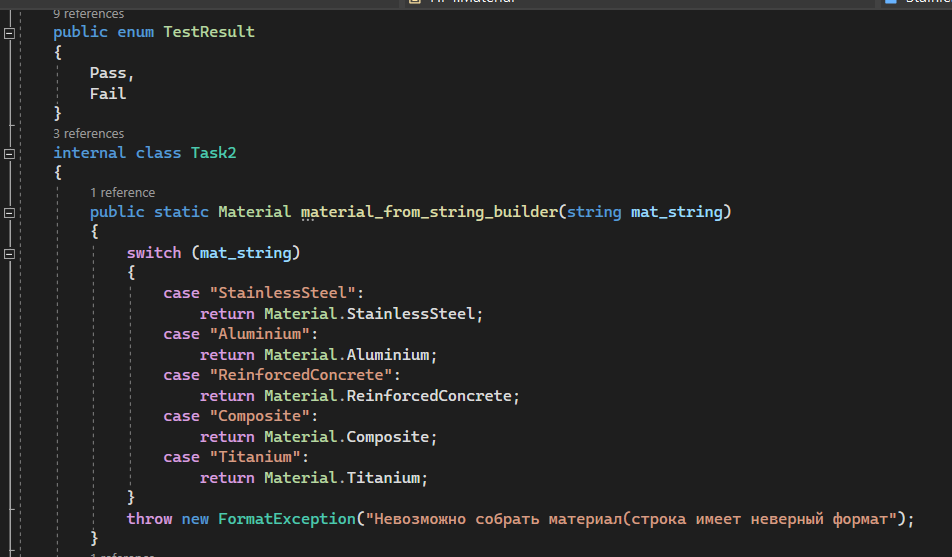


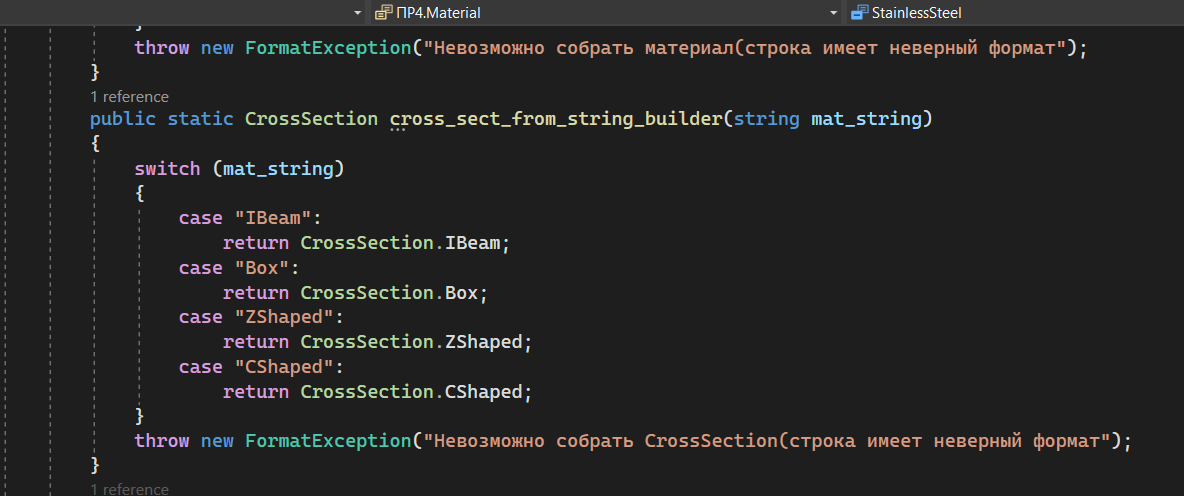
**Задание 2. Использование перечислений.**

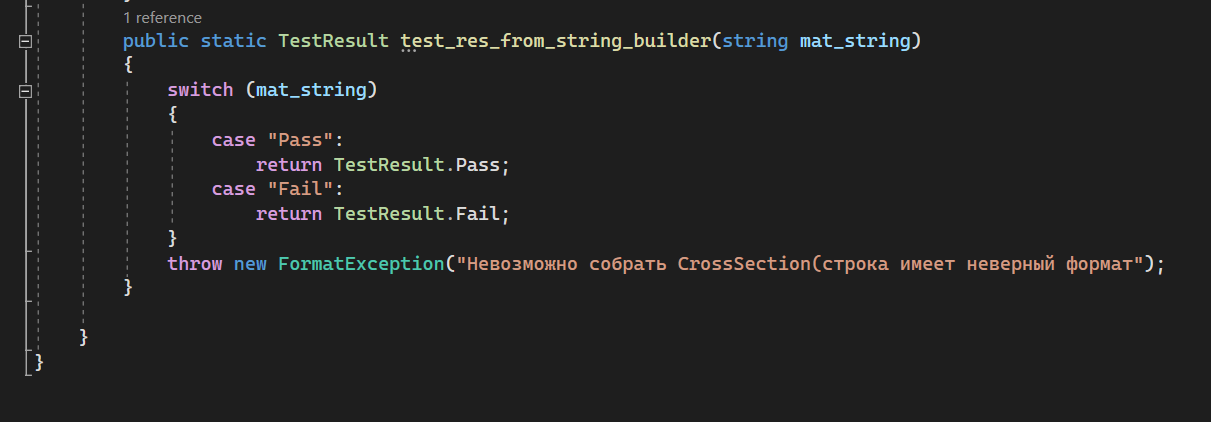




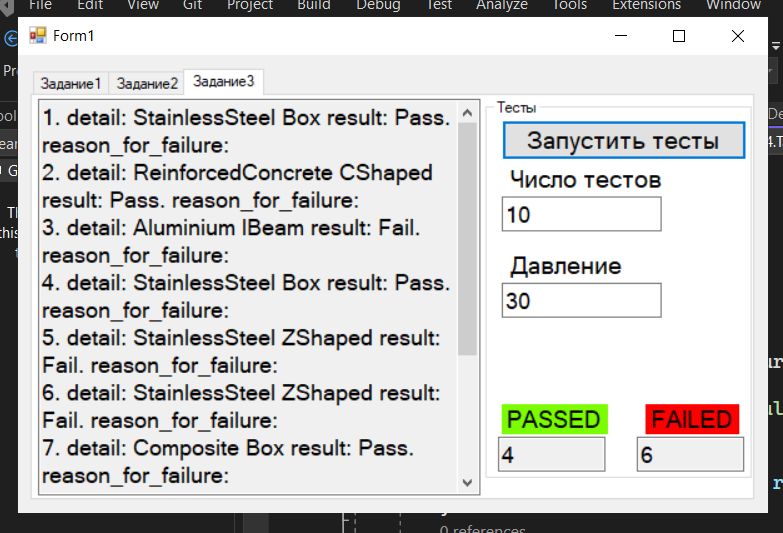


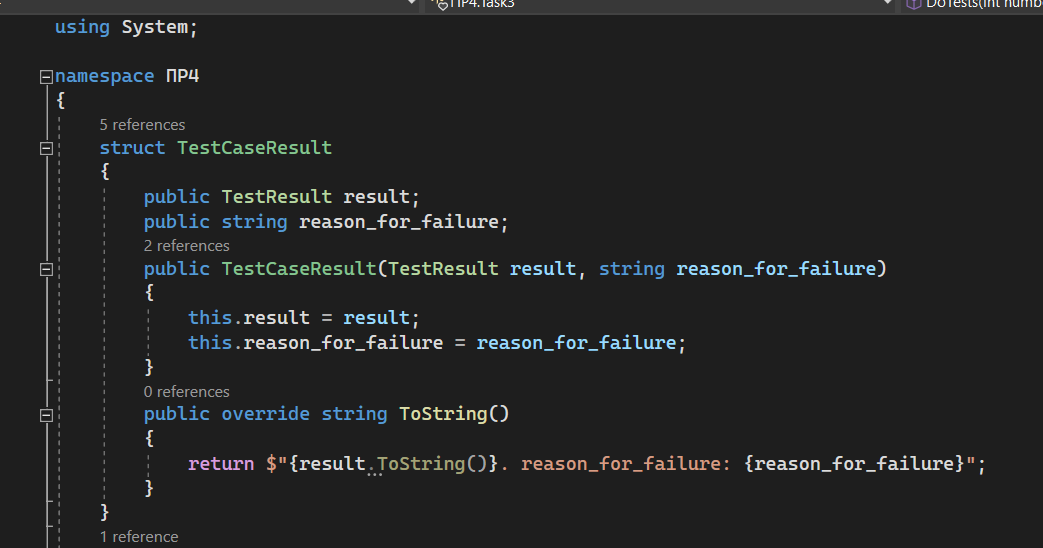


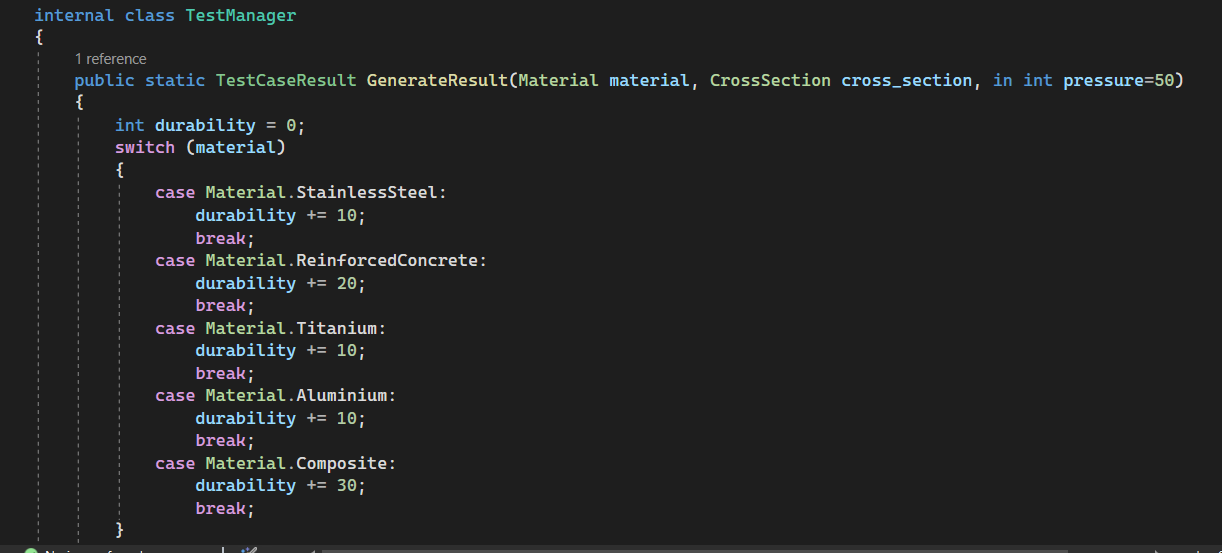


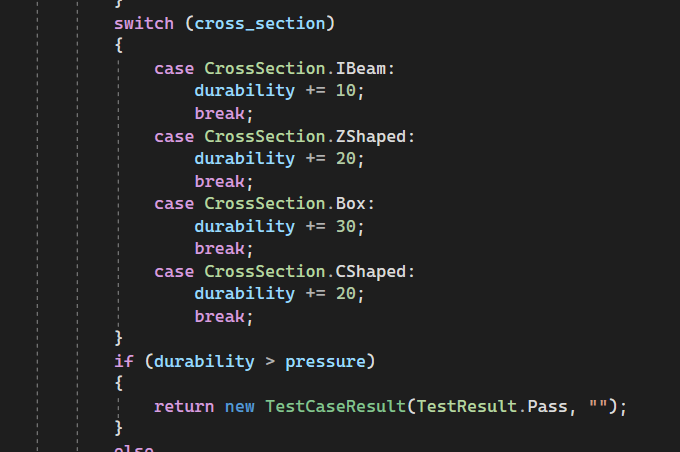


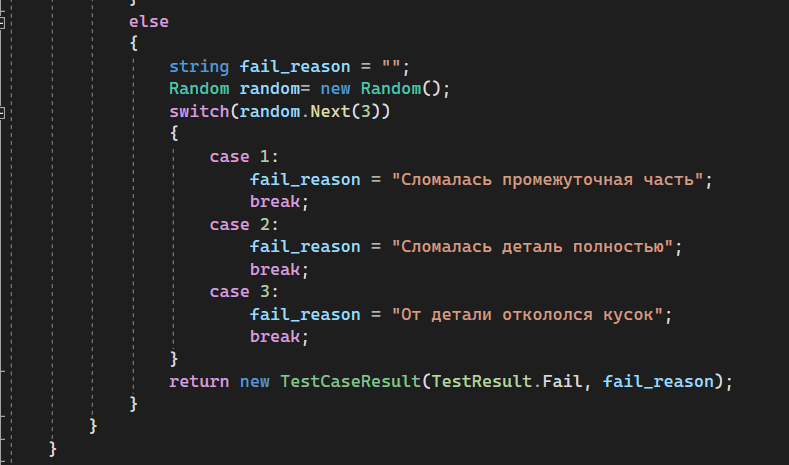
**Задание 3. Использование структур**

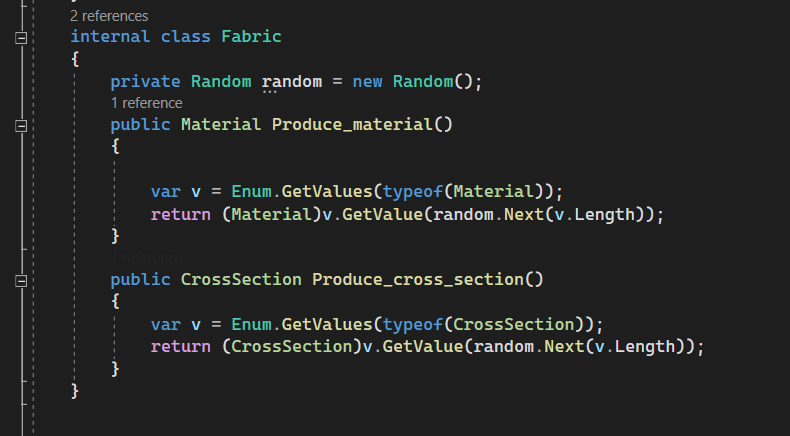


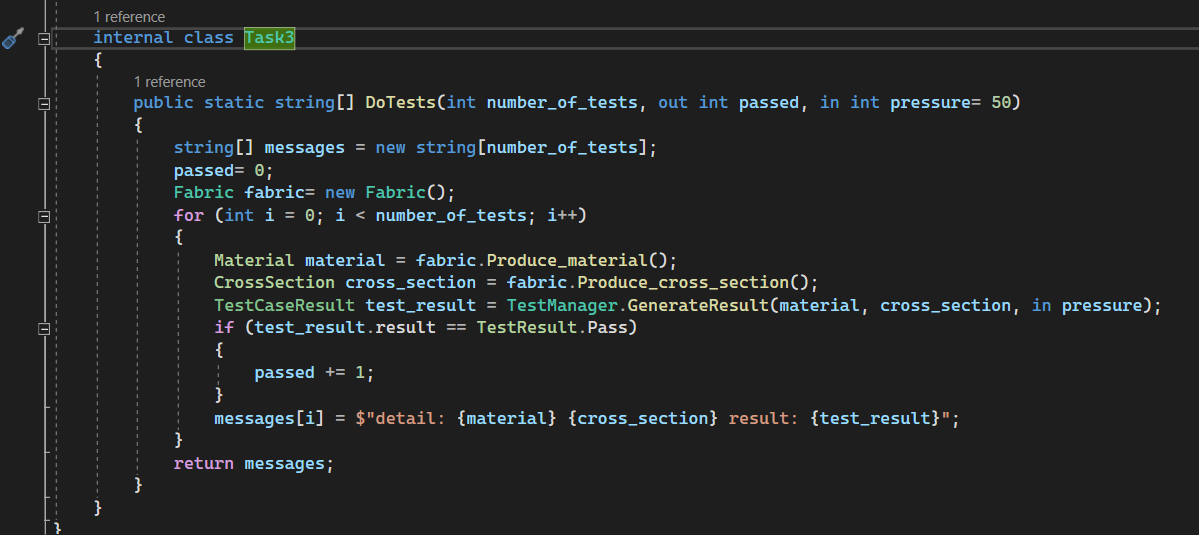












**Представлены 3 проекта, реализованных в Visual Studio Community 2022.  
Проекты представлены преподавателю в электронной форме, продемонстрирована их работоспособность, разъяснены детали программного кода.**

**Вопросы, заданные преподавателем:**

1. Что такое перечисление (Enum) в C# и для чего его используют?

Ответ: Перечисление (Enum) - это тип данных, который позволяет определить набор именованных констант. Он используется для удобства работы с константами, которые имеют определенный порядок или связь между собой.

1. Как создать перечисление в C#?

Ответ: Для создания перечисления в C# необходимо использовать ключевое слово enum, после которого указываются имена констант.

1. Что такое структура (Struct) в C# и в чем ее отличие от класса?

Ответ: Структура (Struct) - это тип данных, который содержит набор полей и методов. Отличие структуры от класса заключается в том, что структура передается по значению, а класс - по ссылке.

1. Как создать структуру в C#?

Ответ: Для создания структуры в C# необходимо использовать ключевое слово struct, после которого указываются поля и методы.

1. Какие преимущества и недостатки имеют структуры по сравнению с классами в C#?

Ответ: Преимущества структур в C# - это более эффективное использование памяти и передача по значению. Недостатки - ограниченный функционал и невозможность наследования.

1. Что такое Nullable в C# и для чего его используют?

Ответ: Nullable - это тип данных, который позволяет переменной принимать значение null. Он используется для работы с базами данных и другими источниками данных, где значения могут быть неопределенными.

1. Какие методы и свойства доступны для работы со структурами в C#?

Ответ: Для работы со структурами в C# доступны методы и свойства, такие как ToString(), Equals(), GetHashCode(), GetType() и другие.

Ссылка на github (Исходный код всех работ)

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