Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по учебной практике

УП 01.01

Выполнил: Епимахов Денис Леонидович

Группа: ПР-21

Преподаватель: Ватолина Т. А.

2025 г.

Содержание

**1. Практика\_2\_1. «Индивидуальные Файлы»** 3

[1. **Задание 1. Вариант – 65.** 3](#_Toc118960339)

[1.1 Постановка задачи 3](#_Toc118960340)

[1.2 Схема алгоритма 4](#_Toc118960341)

[1.3 Код программы 5](#_Toc118960342)

[1.4Тестовые случаи 6](#_Toc118960344)

[2. **Задание 2. Вариант – 47.** 7](file:///C:\Users\79126\Desktop\GAMES\C%23\колледж\УП_Ватолина\Отчеты_по_УП\Отчет_Практика_2_1_Индивидуальное_Файлы.docx#_Toc118960339)

[2.1 Постановка задачи 7](#_Toc118960340)

[2.2 Схема алгоритма 8](#_Toc118960341)

[2.3 Код программы 9](#_Toc118960342)

[2.4Тестовые случаи 9](#_Toc118960344)

**2.** [**ПРАКТИКА 2.2. «ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ 2»** 10](#_Toc118960339)

[1.1 Постановка задачи 10](#_Toc118960340)

[1.2 Схема алгоритма 11](#_Toc118960341)

[1.3 Код программы 13](#_Toc118960342)

1.3 Тестовые случаи 15

**3.** **Практика 7. «Парные картинки».** 17

1.1 Постановка задачи 17

1.2 Код программы 18

1.3 Тестовые случаи 31

1.4 Приложение 33

**4.** **Практика 8. «Игра пятнашки».** 36

1.1 Постановка задачи 36

1.2 Код программы 37

1.3 Тестовые случаи 49

1.4 Приложение 52

**5. Практика 22. «Нахождение НОП».** 54

1.1 Постановка задачи 54

1.2 Тест план 55

1.3Код программы 56

1.4Код методов для тестирования 61

1.5 Модульные тесты 65

1.6 Белый ящик 70

**6.** **Практика 5. «Разработка тестирующей программы».** 71

1.1 Постановка задачи 71

1.2 Код методов: организующих работу с checBox 72

1.3 Код методов: редактирования и удаления вопросов 73

1.4 Код методов: добавления при помощи OpenFileDialog 75

**7.** **Практика 12. «Работа с БД индивидуальное задание».** 76

1.1 Постановка задачи 76

1.2 Таблицы 76

1.3 Интерфейс приложения 77

1.4 Метод добавления 78

1.5 Метод изменения 79

1.6 Метод удаления 80

1.7 Метод: Поиск взрослых пациентов по роду деятельности 81

1.8 Метод: Найти детей, у которых флюорография не пройдена 82

1.9 Метод: Найти пациентов старше заданного возраста 83

**8.** **Практика 10-11. «Работа с БД общее задание».** 84

1.1 Постановка задачи 84

1.2 Таблицы 84

1.3 Код методов 85

**Практика 2.1 Индивидуальное Файлы**

# Вариант - 65

## 1.1 Постановка задачи

**Задача:** Дан строковый файл. Создать новый строковый файл, содержащий все строки исходного файла наибольшей длины (в обратном порядке).

## Входные данные:

**inputFilePath** – путь начального файла; строка, string **outputFilePath** – путь конечного файла; строка, string **maxLengthLines** – максимальная длина строки; целое, int

**Выходные данные:**

**long**\_– самая длинная строка, записанная в обратной последовательности; строка, string

* 1. **Схема алгоритма**

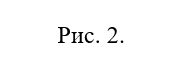
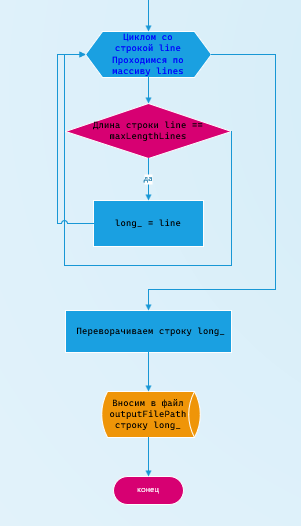
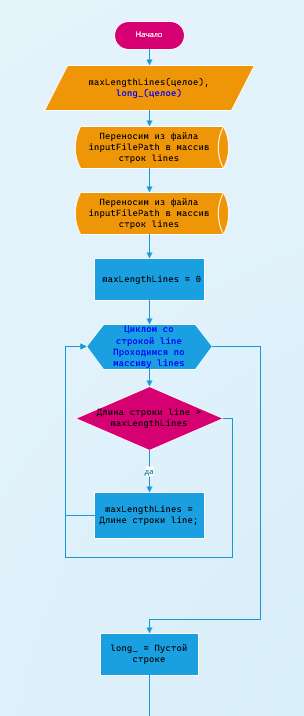
****

Рис. 1.

**1.3 Код программы**

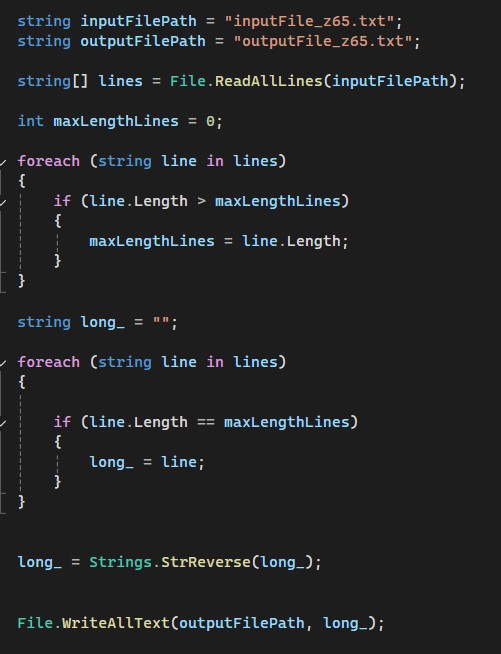
****

Рис. 3.

**1.4 Тестовые случаи**

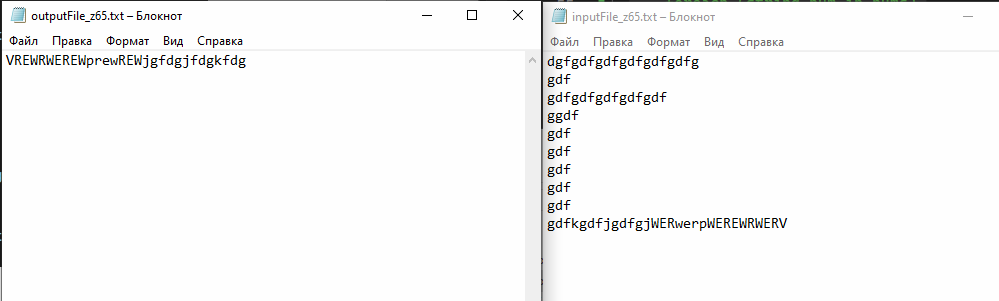
****

Рис. 4.

# Вариант - 47

## 2.1 Постановка задачи

**Задача:** Дан текстовый файл, каждая строка которого изображает целое или вещественное число, дополненное слева и справа несколькими пробелами (вещественные числа имеют ненулевую дробную часть). Вывести количество целых чисел и их сумму.

## Входные данные:

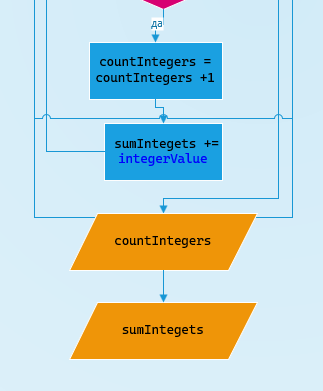
**countIntegers** – количество целых чисел; целое, int

**sumIntegets** – сумма целвх чисел, строка; string

**maxLengthLines** – максимальная длина строки, целое, int

**Выходные данные:**

**long**\_- самая длинная строка, записанная в обратной последовательности; строка, string

**2.2 Схема алгоритма**

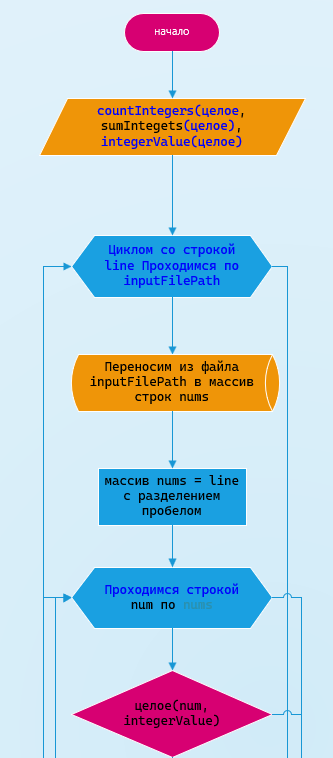
** **

Рис. 5.

**2.3 Код программы**

Рис. 7.

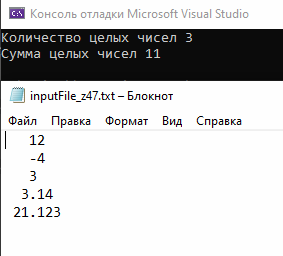
**2.4 Тестовые случаи**

Рис. 8.

# ПРАКТИКА 2.2 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ 2 Текстовые файлы.

Вариант - 10

# Постановка задачи

# Задача: Удалить все скобки и символы, заключенные между словами

# Входные данные:

pathFile – имя файла c данными; строковое, string

inputFile – имя файла, в который будет записаны изменения; строковое, string

# Выходные данные:

inputFile – имя файла, в который будет записаны изменения; строковое, string

modContent – переменная, с измененными данными; строковое, string

# Сообщения:

- «Введите имя файла»

- «Введите имя файла, в который будет записаны изменения»

- «Изменения сохранены в файл:»

# 1.2 Схема алгоритма

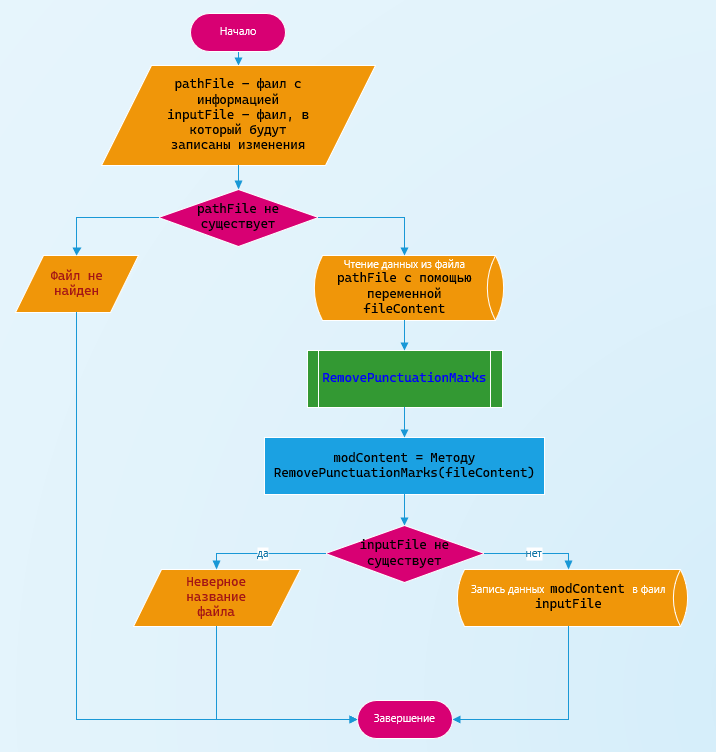
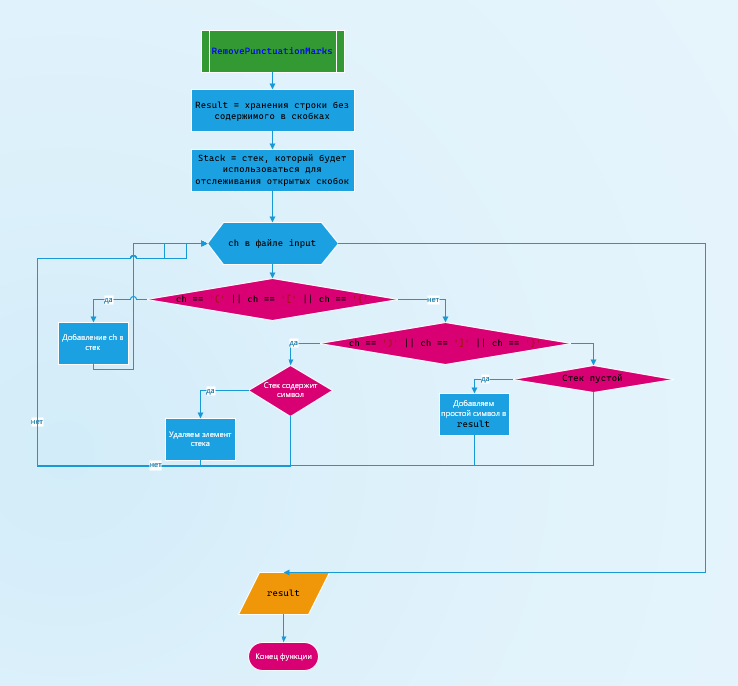
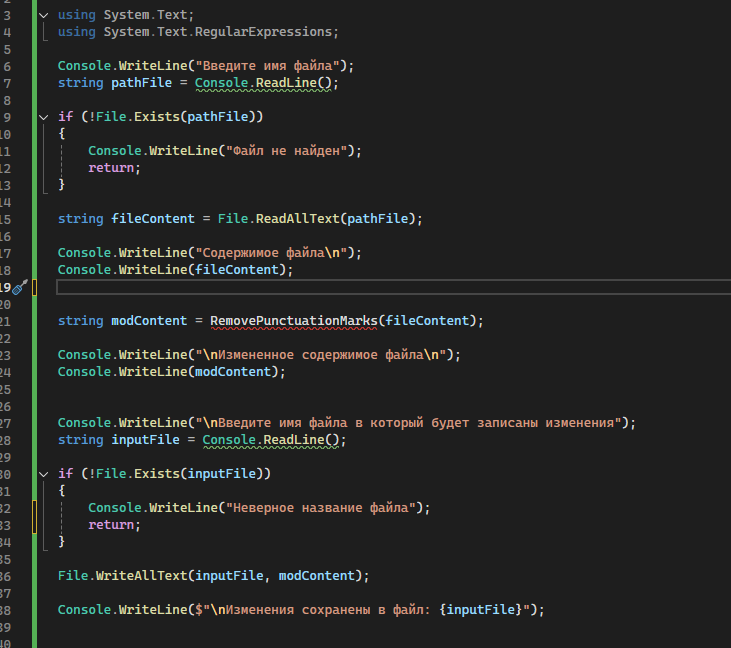
****

Рис.1.

 Рис.2.

* 1. **Код программы**

Рис.3.

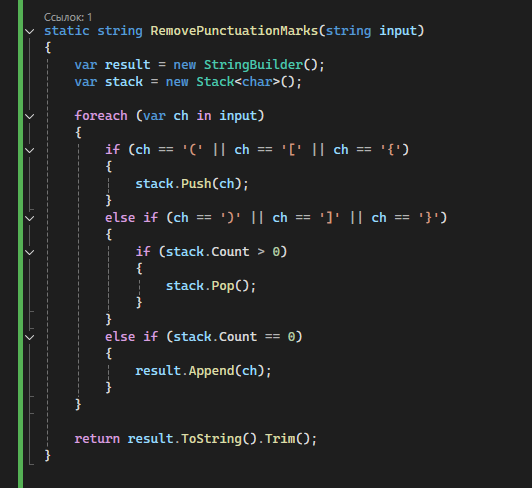
****

Рис.5.

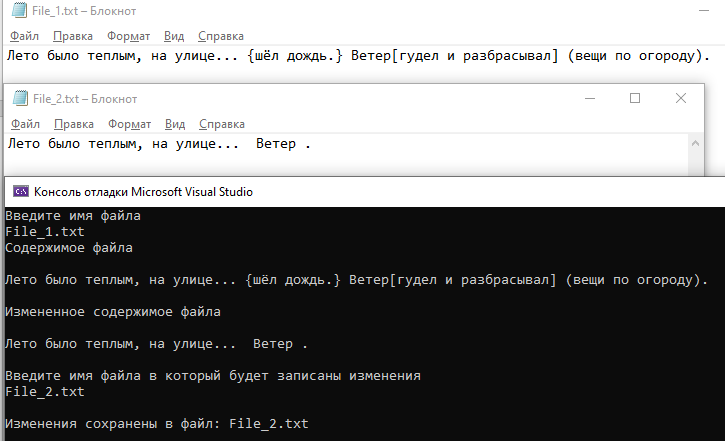
**1.4 Тестовые случаи**

Рис.6.

Практика 7. Парные картинки.

# 1.1 Постановка задачи

Разработать оконную программу в Windows Forms «Парные картинки».

**1.2 Код программы**

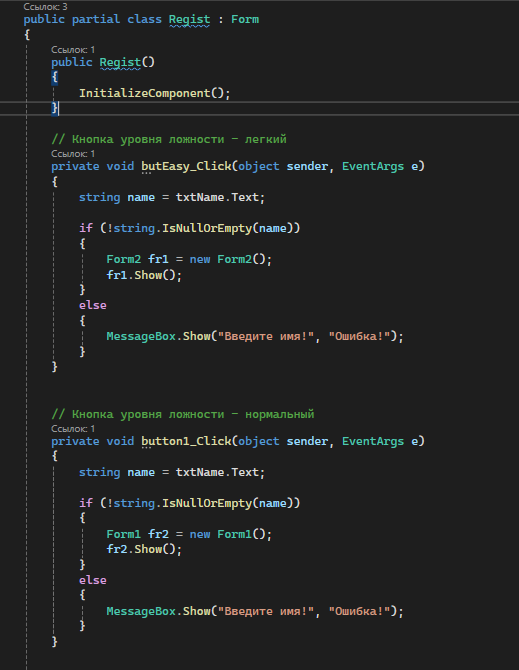
 Код класса – Regist

Рис.1.



Рис.2.

Код класса – Form1

Рис.3.

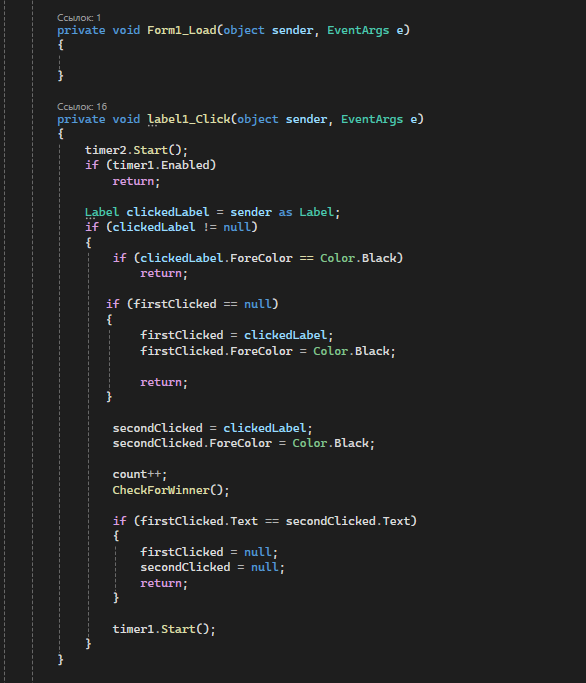
Рис.4.

Рис.5.

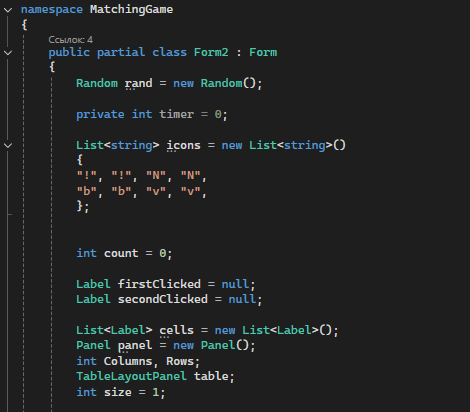
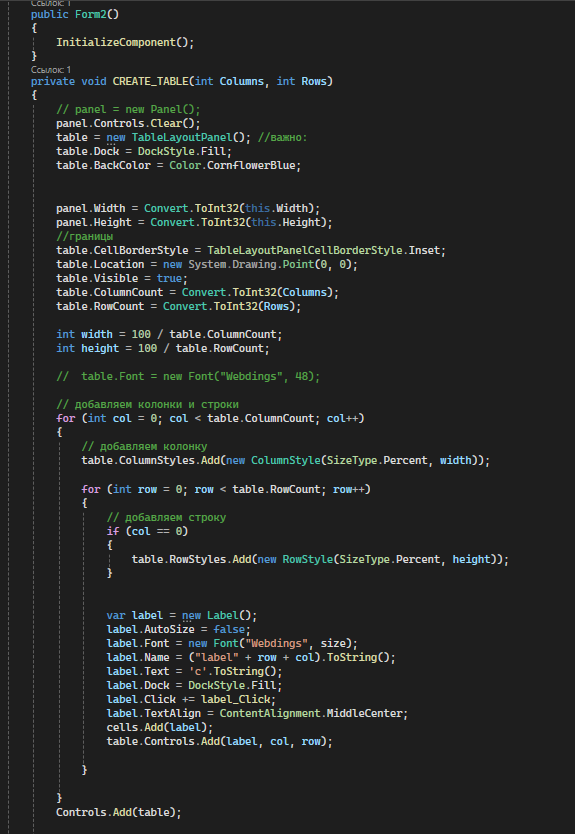
 Код класса – Form2

Рис.6.

 Рис.7.

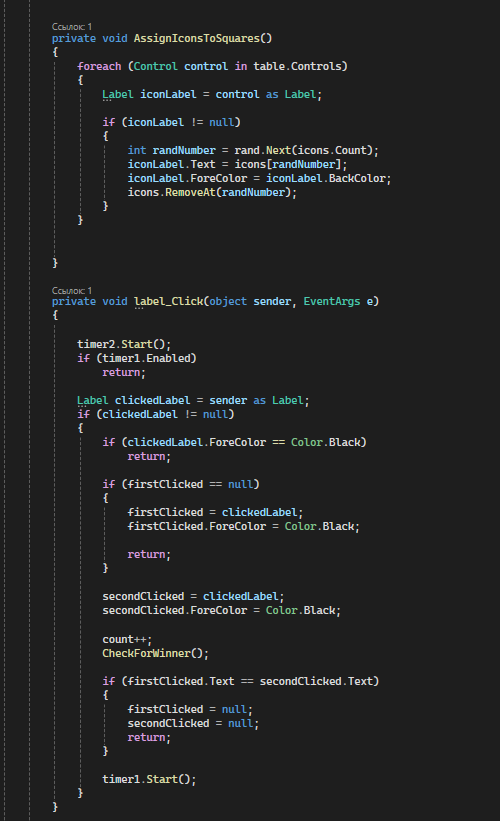
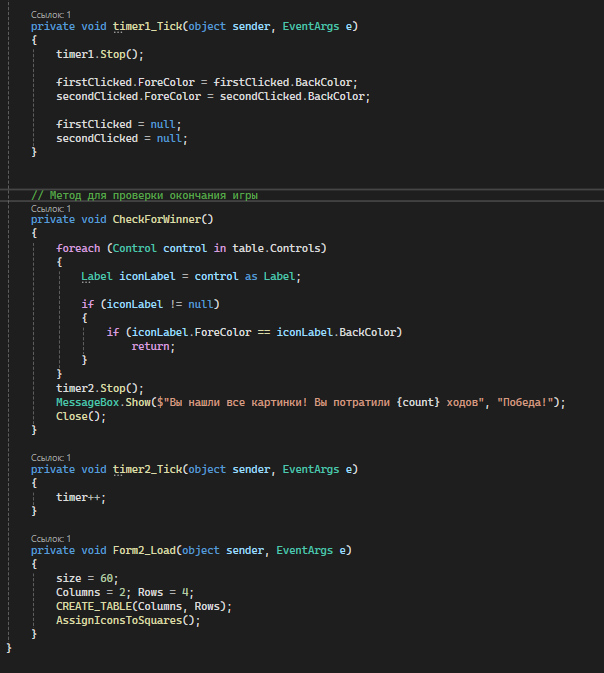


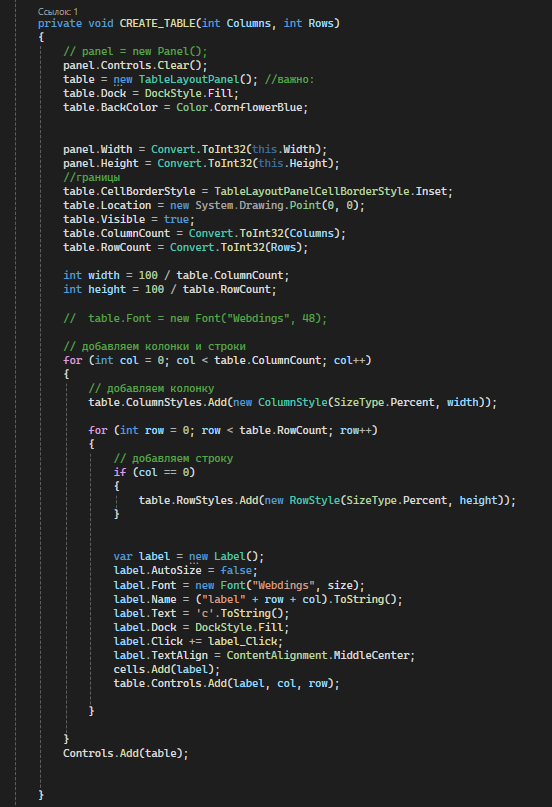
Рис.8.

Рис.9.

Код класса – Form3

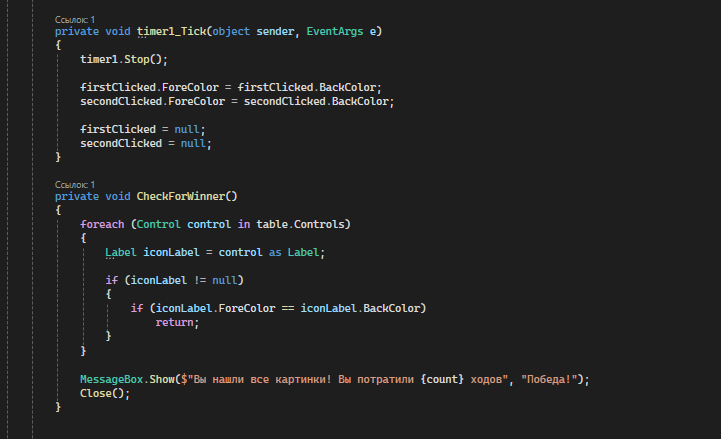


Рис.10.

****Рис.11.

****

Рис.12.

Рис.13.

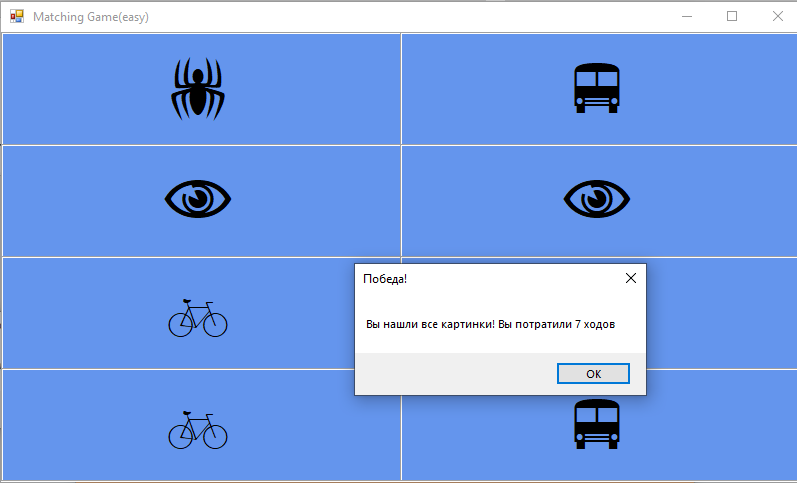
**1.3 Тестовые случаи**

Рис.14.

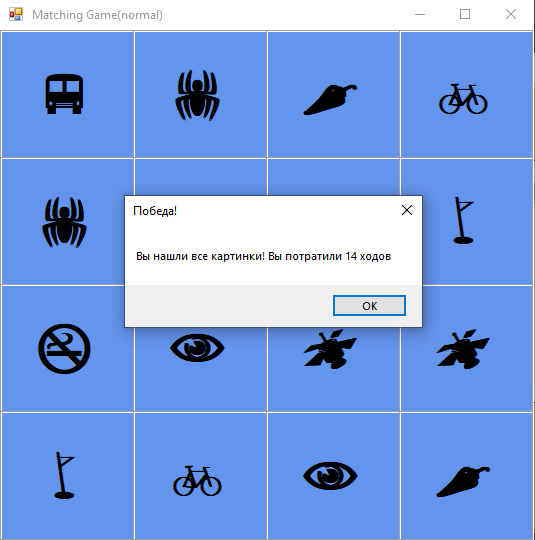
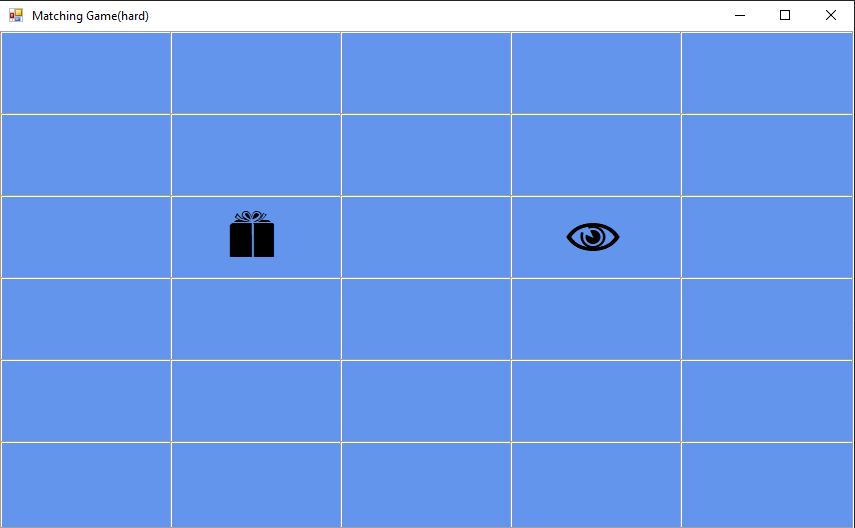


Рис.15.

 Рис.16.

* 1. **Приложение**

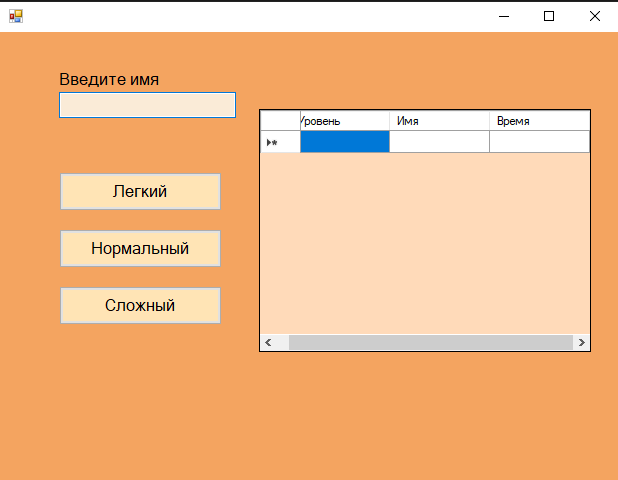
Regist [Конструктор]

Рис.17.

Практика 8. Игра пятнашки.

# 1.1 Постановка задачи

**Задача:** Разработать оконную программу в Windows Forms «Пятнашки».

Возможности:

* Выбора размера игрового поля (2×2 - 6×6)
* Генерации случайной начальной расстановки;
* Перемещения плиток по правилам игры;
* Автоматической проверки завершения игры;
* Визуального отображения игрового процесса;

**1.2 Код программы**

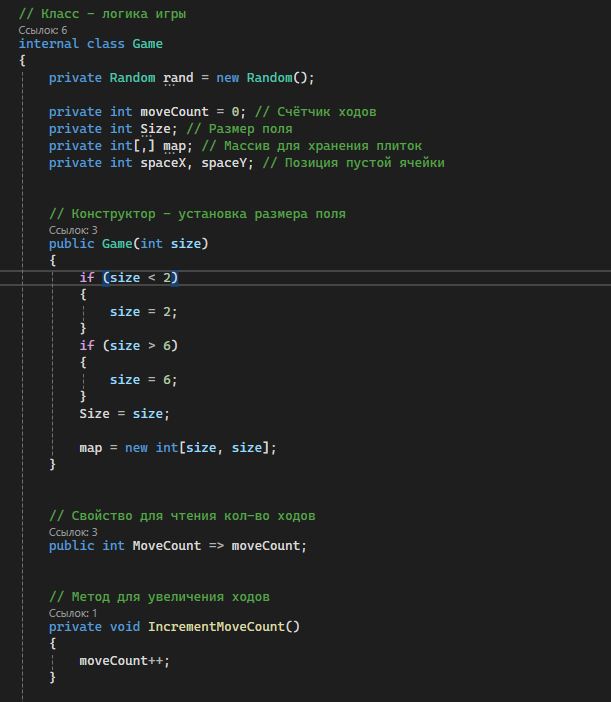
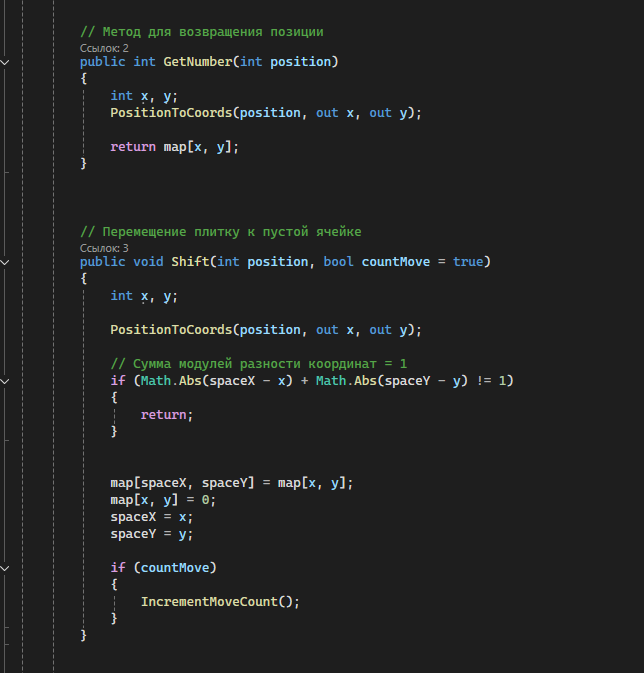
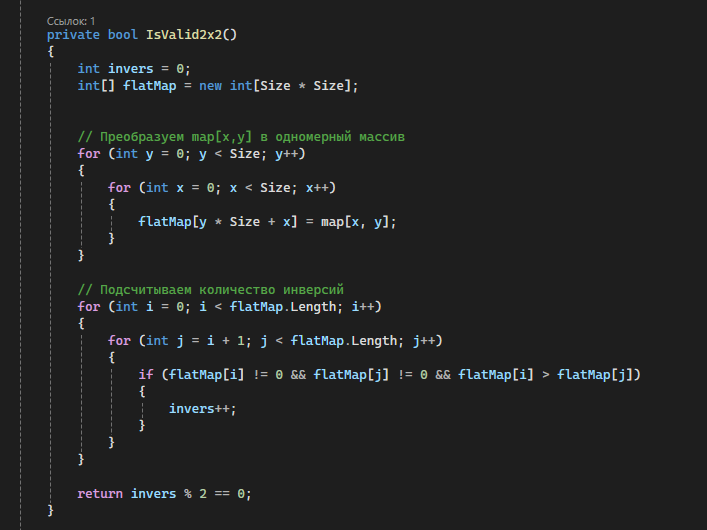
Код класса – Game.

Рис.1.

****Рис.2.

****Рис.3.

****Рис.4.

****Рис.5.

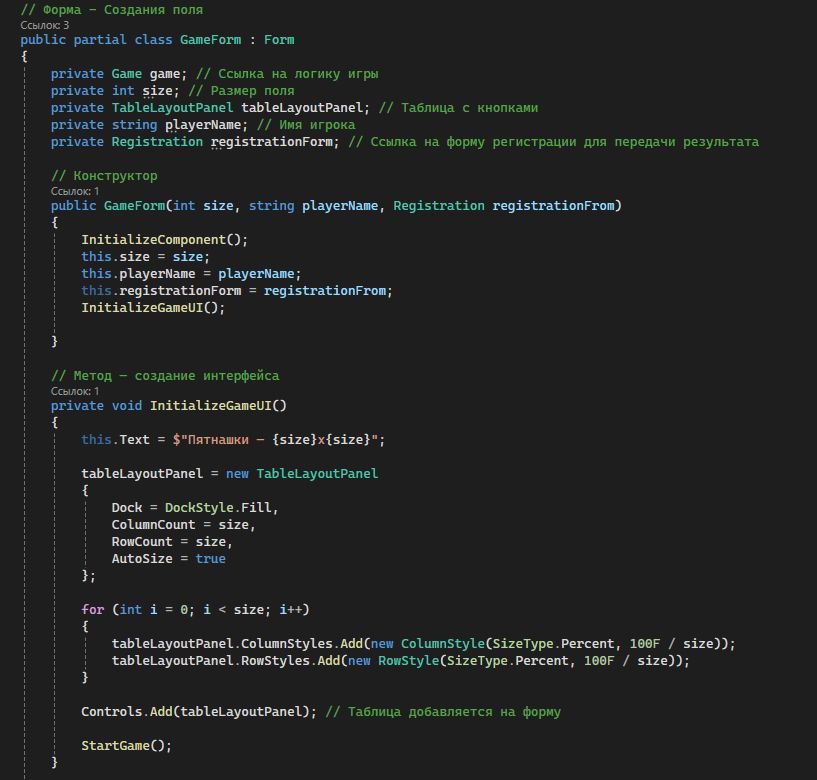
Код класса – GameForm

Рис.6.

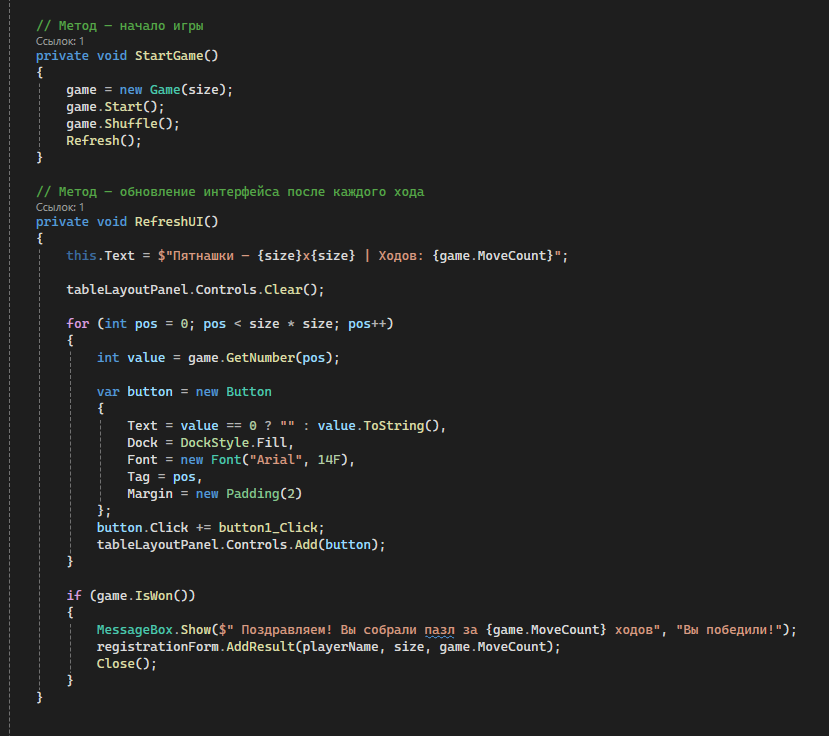
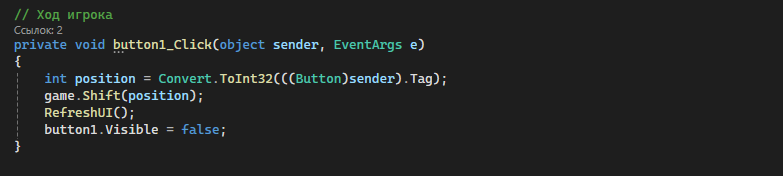
Рис.7.

Рис.8.

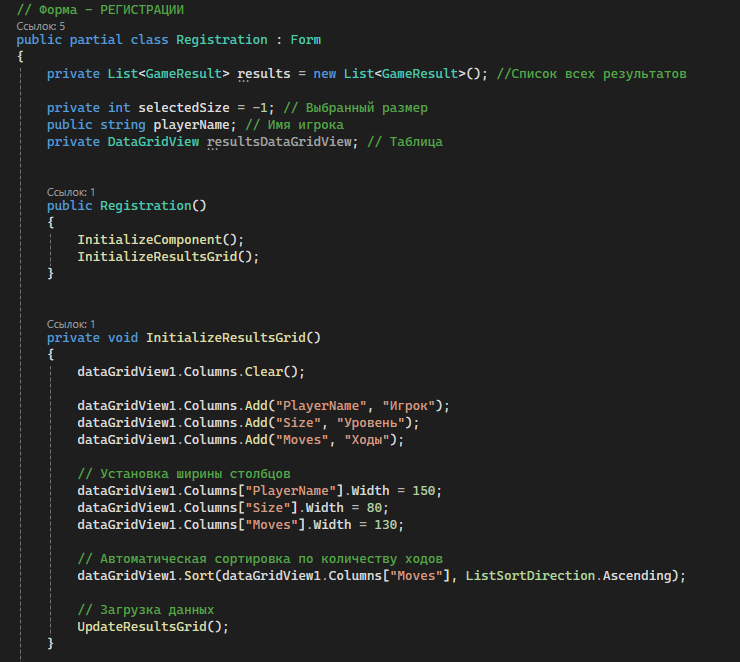
Код класса – Registration

Рис.9.

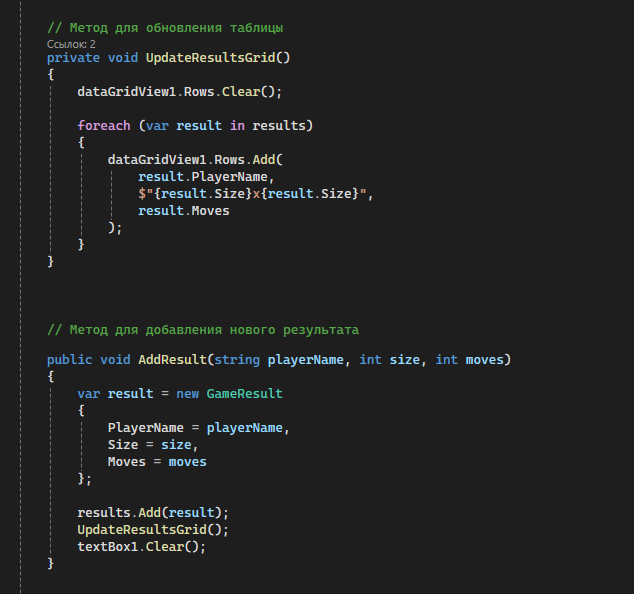
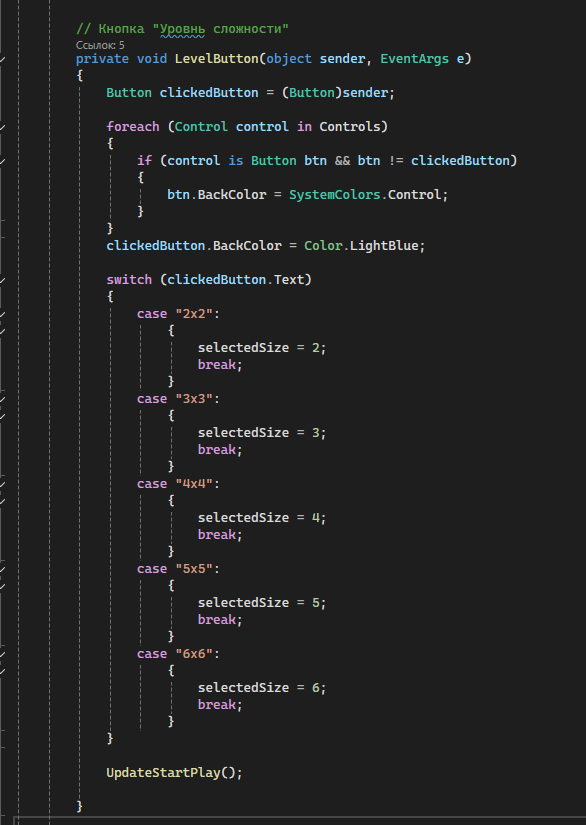
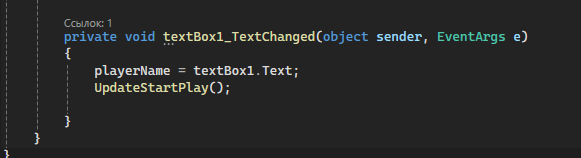
Рис.10.

Рис.11.

Рис.12.

Рис.13.

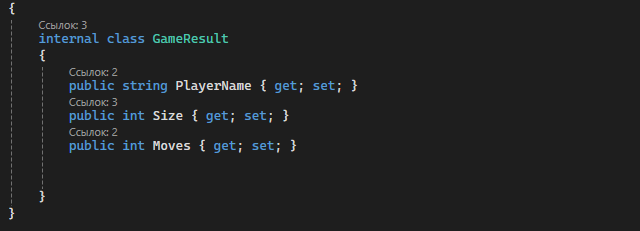
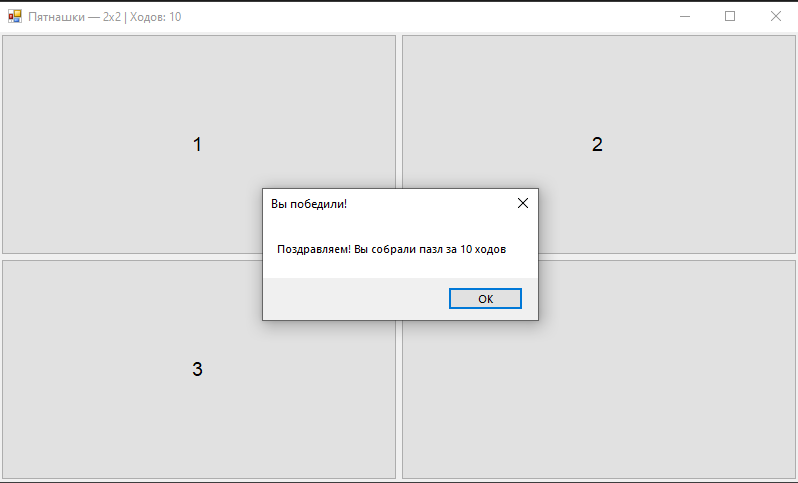


Рис.14.

**1.3 Тестовые случаи**

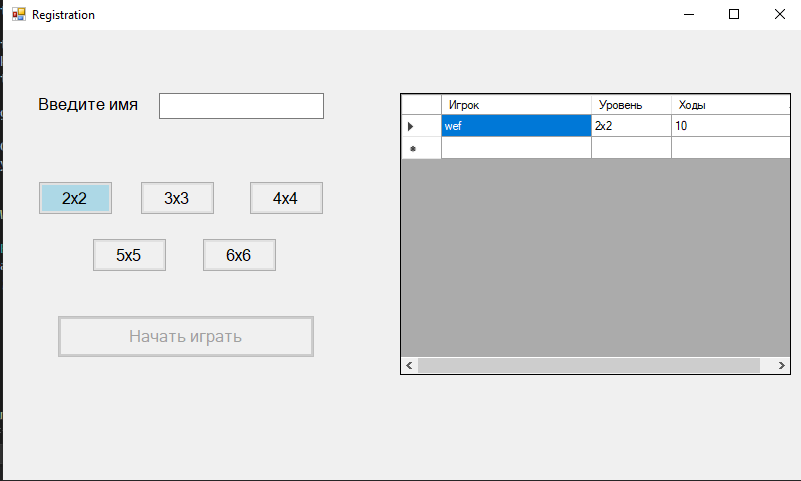
Рис.15.

Рис.16.

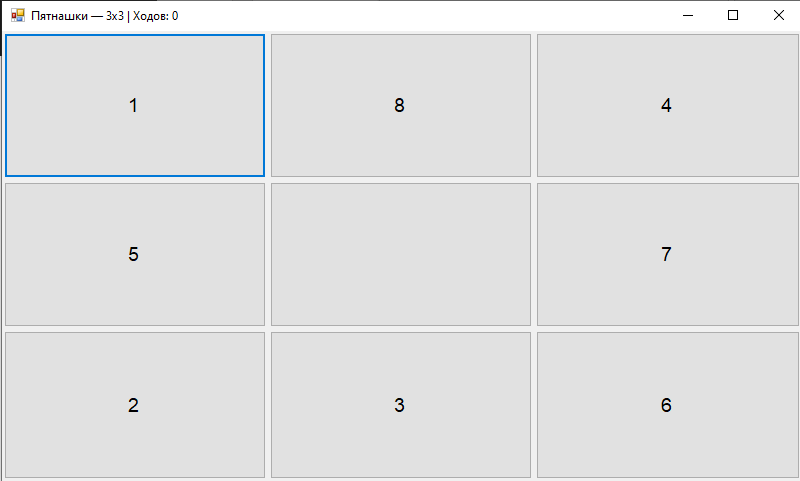
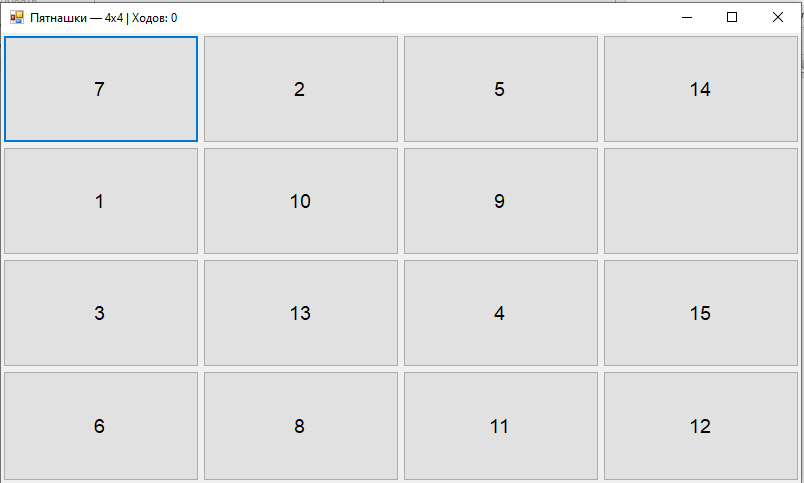
****Рис.17.

Рис.18.

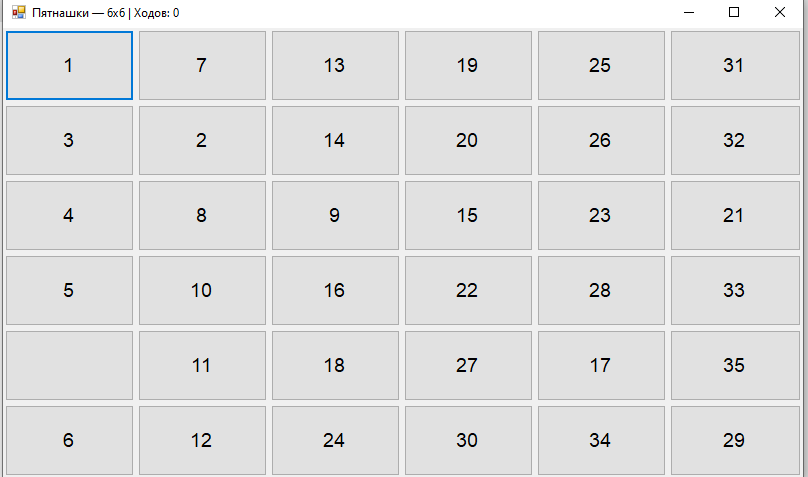
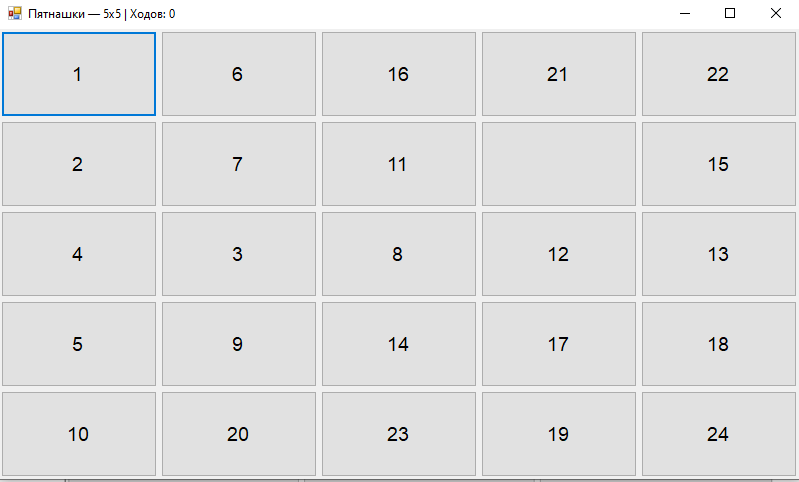
****Рис.19.

Рис.20.

* 1. **Приложение**

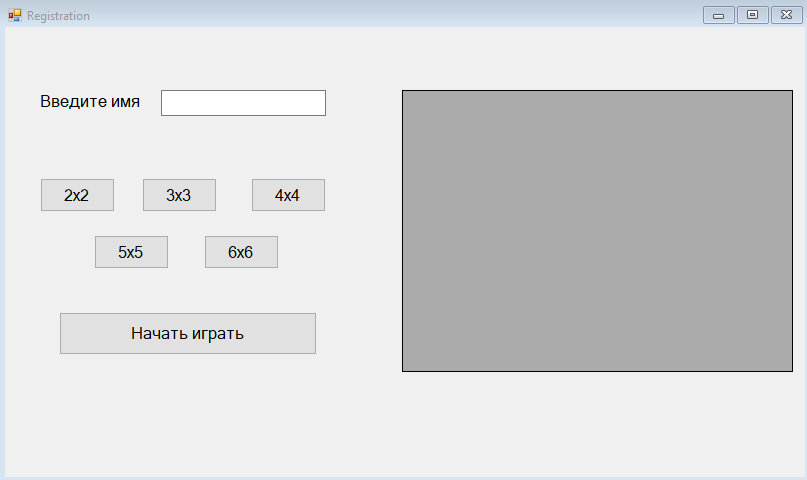
Registration [Конструктор]

Рис.21.

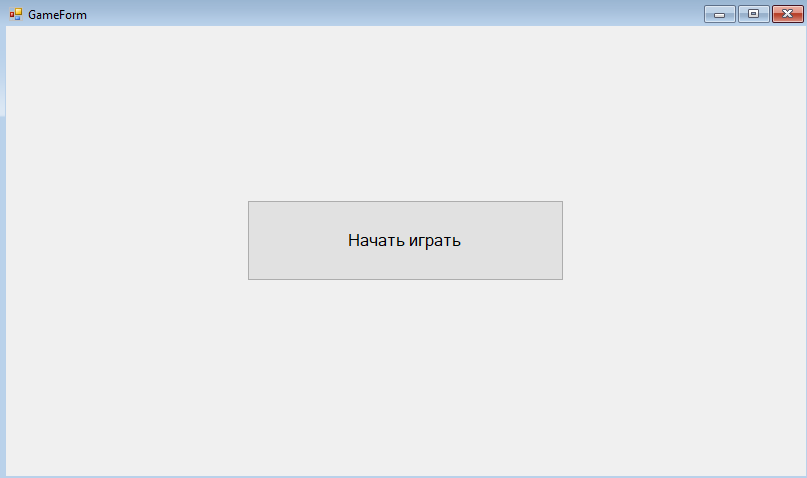
GameForm [Конструктор]

Рис.22.

Практика 22. Нахождение НОП.

## 1.1 Постановка задачи

**Задача:** Вычислить по М строкам самую длинную общую последовательность. Если их несколько, то наиболее «прижатую» к началу.

**Краткая характеристика области применения**

Необходимо реализовать программу, позволяющую находить НОП для M строк, с приоритетом на самую длинную и, при равной длине, наиболее «прижатую к началу» последовательность. Данная программа предназначена для практической работы по ПиТПМ.

## 1.2 Тест план

Рис. 1

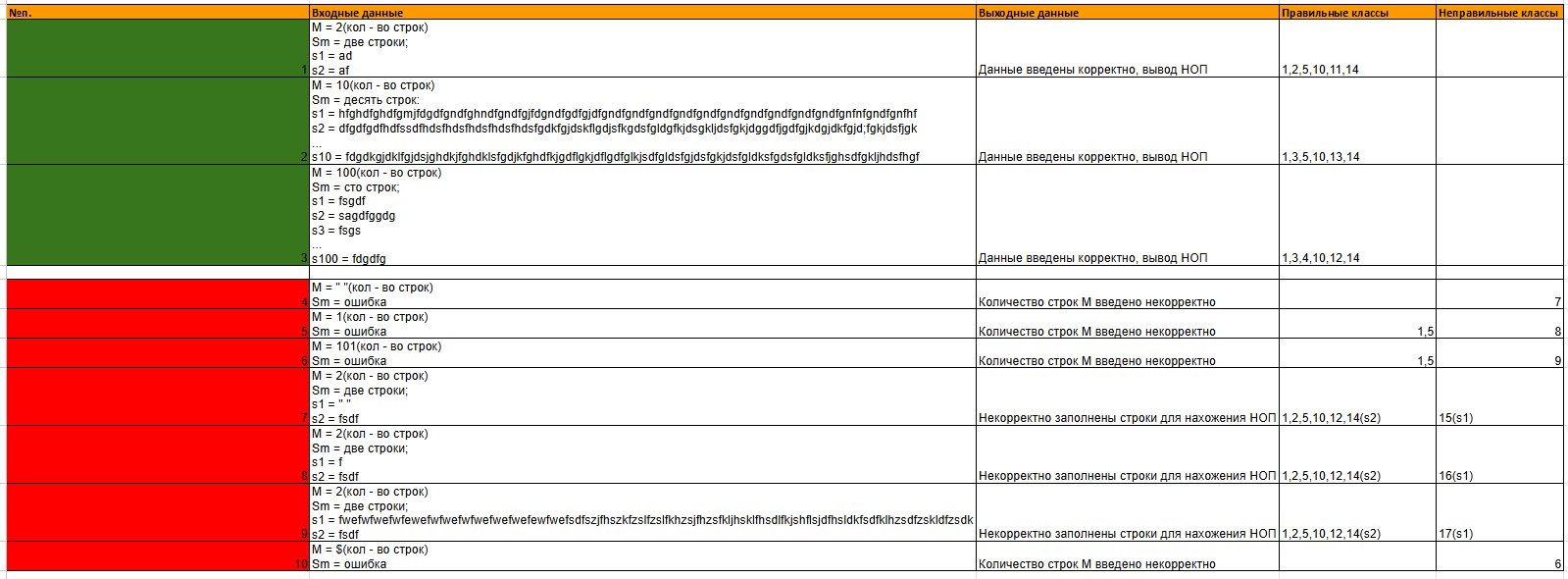
Рис. 2.

Рис. 3.

**1.3 Код программы**

Код класса – Form1

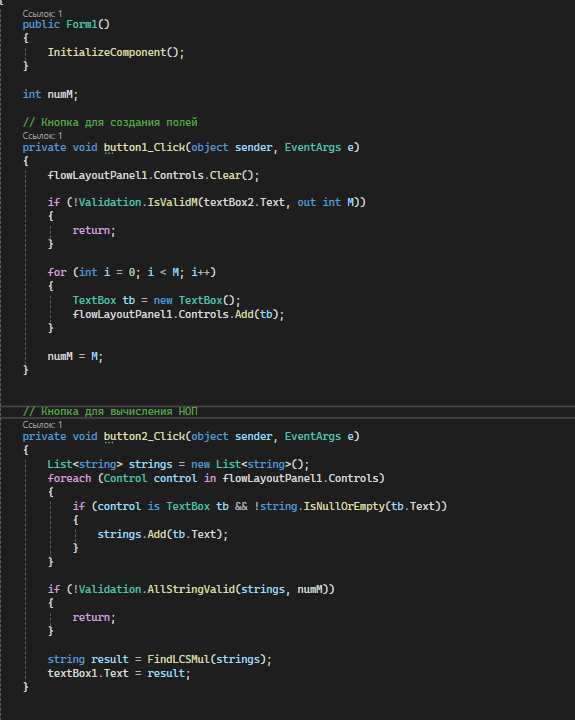
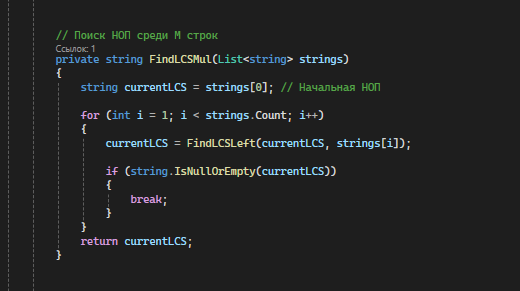


Рис. 4.

****

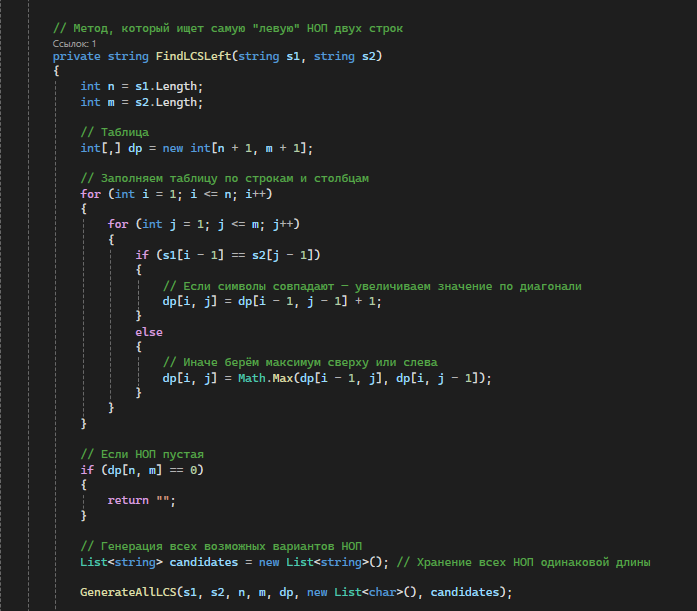
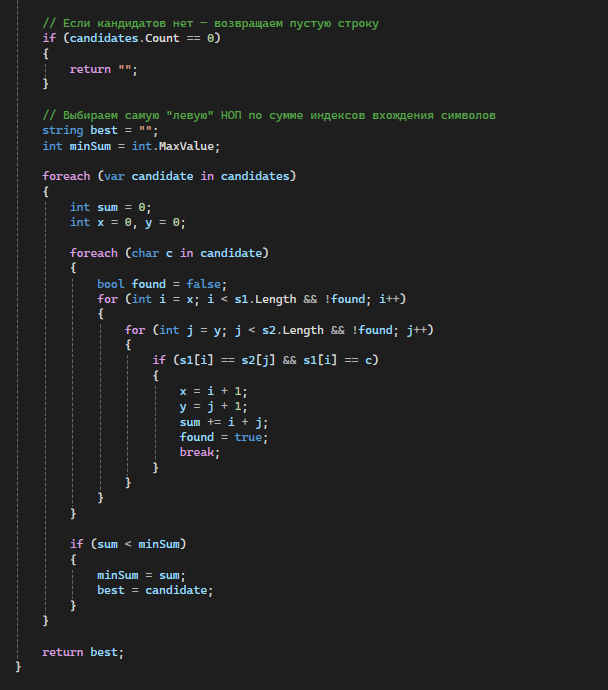
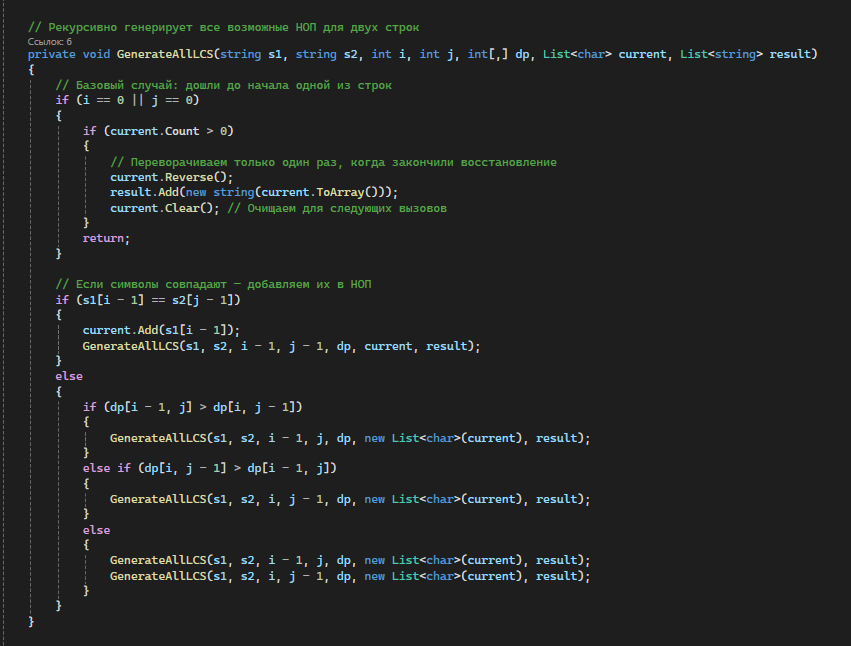
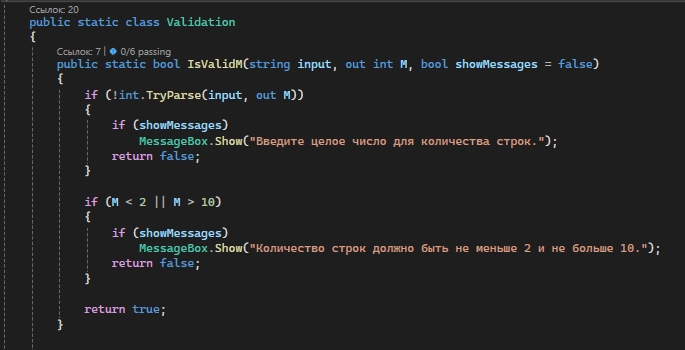
Рис. 5.

Рис. 6.

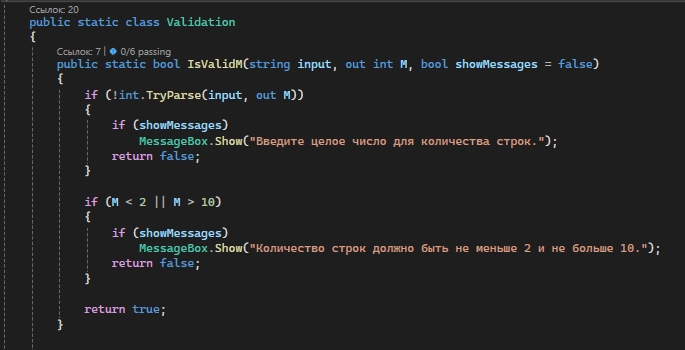
Рис. 7.

****Рис. 8.

****Код класса – Validation

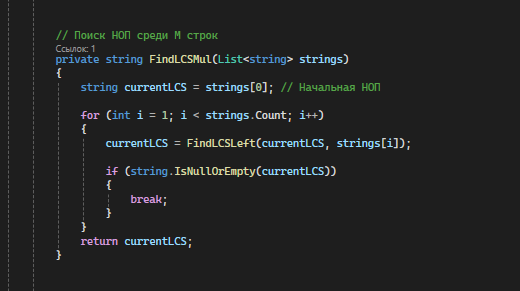
****Рис. 9.

Рис. 10.

**1.4 Код методов для тестирования**

****Рис. 11.

Рис. 12.

****

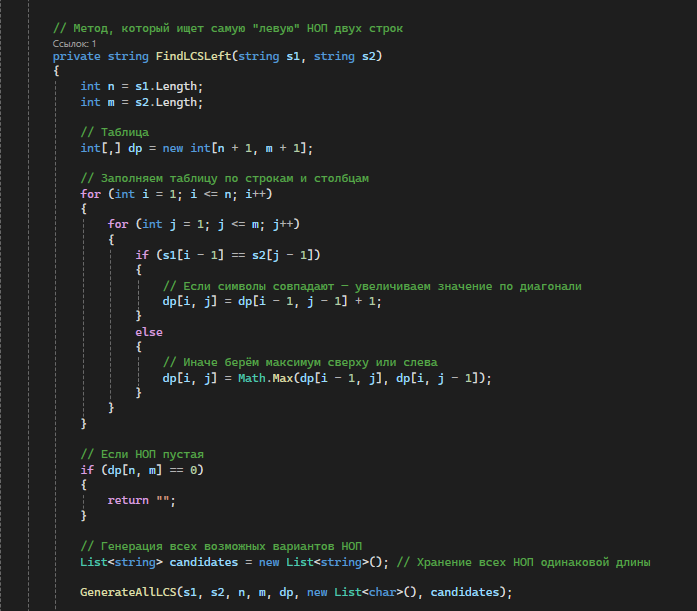
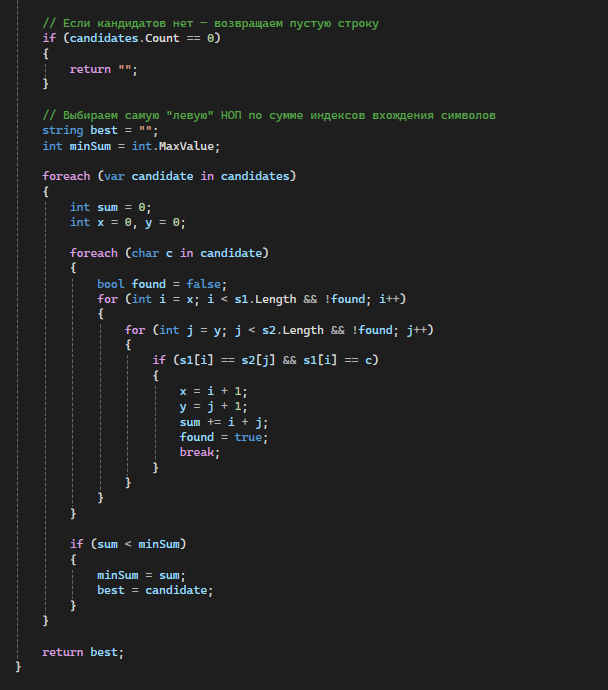
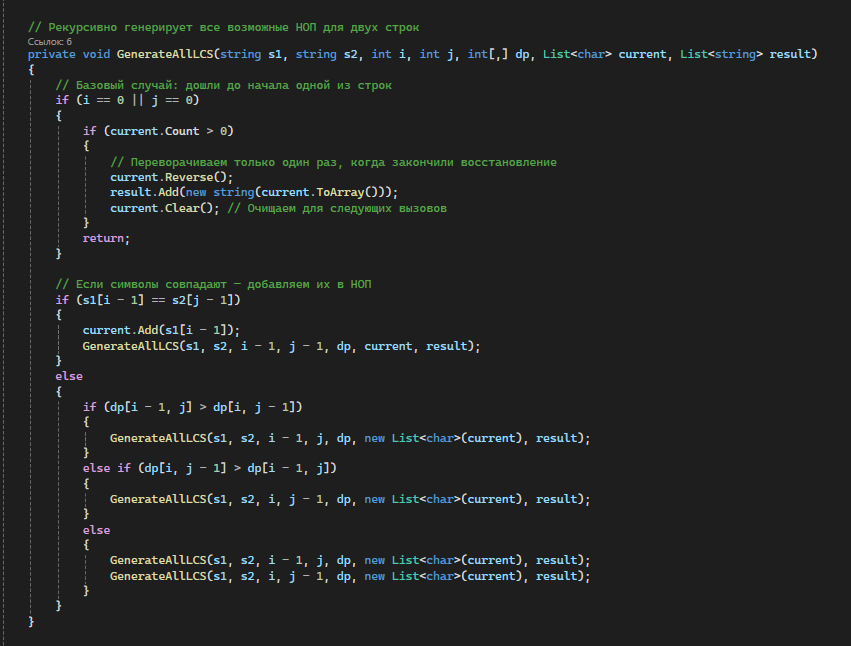
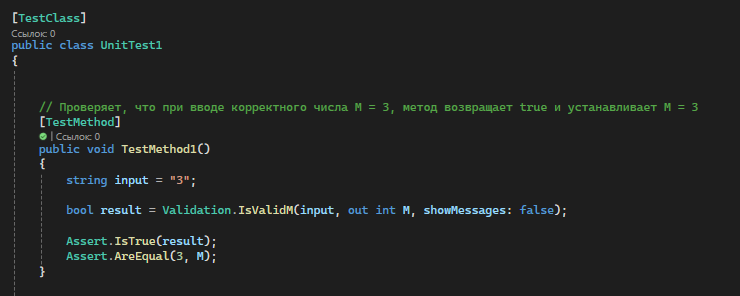
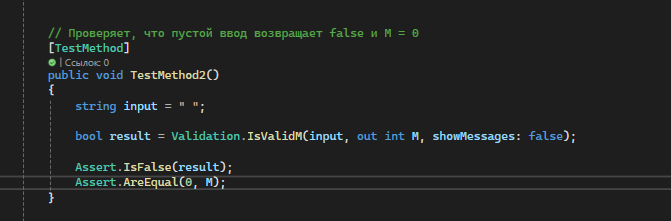
Рис. 13.

Рис. 14.

Рис. 15.

****Рис. 16.

**1.5 Модульные тесты**

Рис. 17.

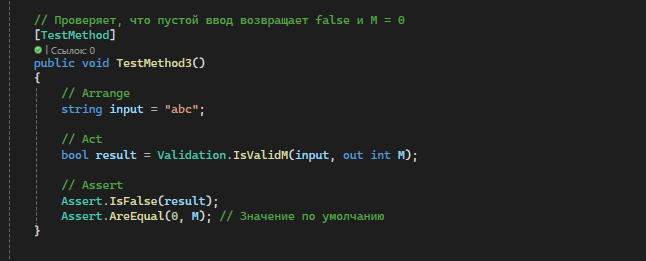
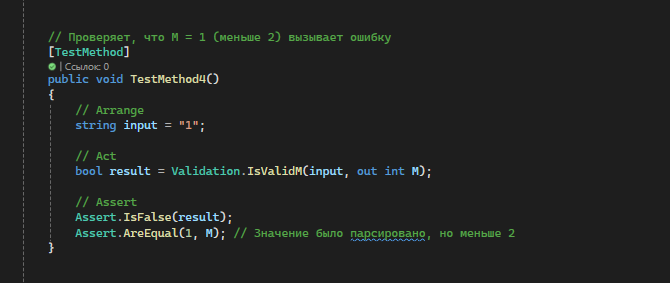
Рис. 18.

Рис. 19.

****Рис. 20.

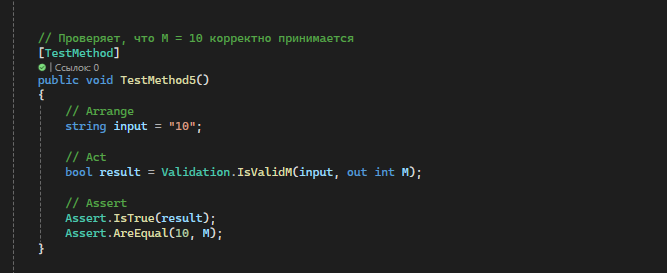
****

Рис. 21.

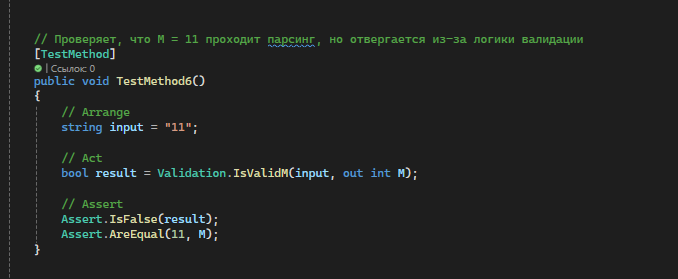


Рис. 22.

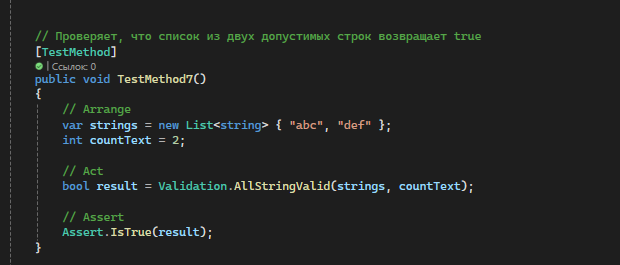
Рис. 23.

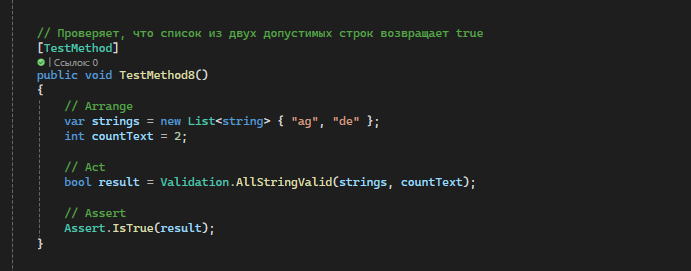
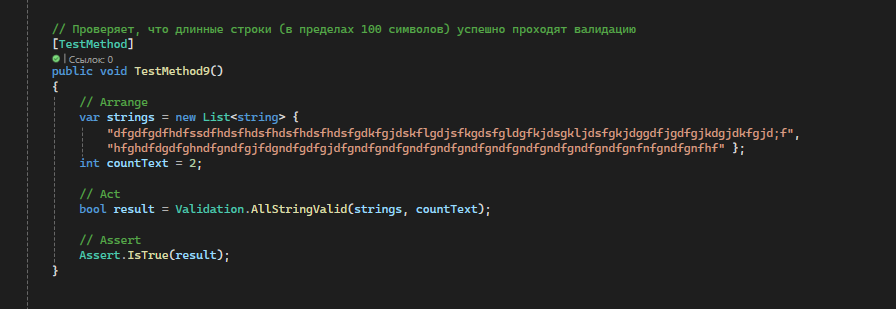
Рис. 24.

Рис. 25.

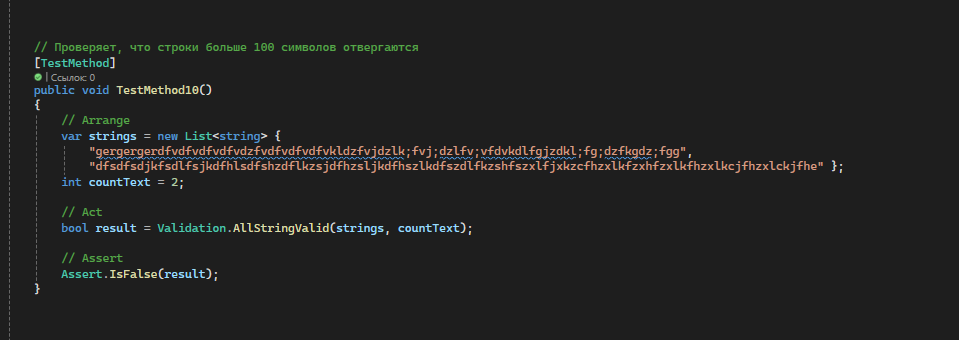
Рис. 26.

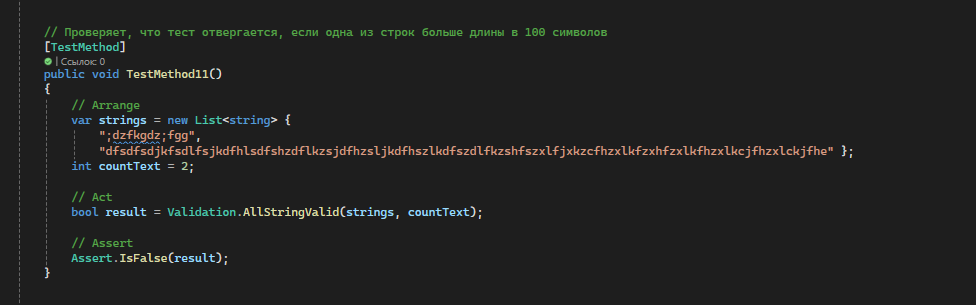
Рис. 27.



Рис. 28.

Рис. 29.



Рис. 30.

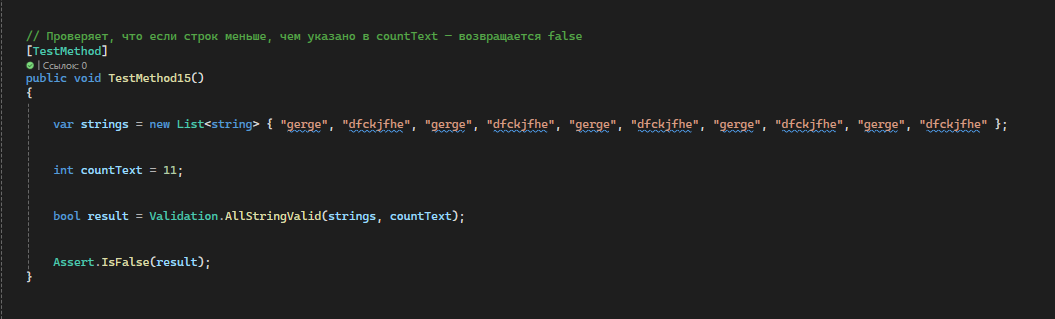


Рис. 31.

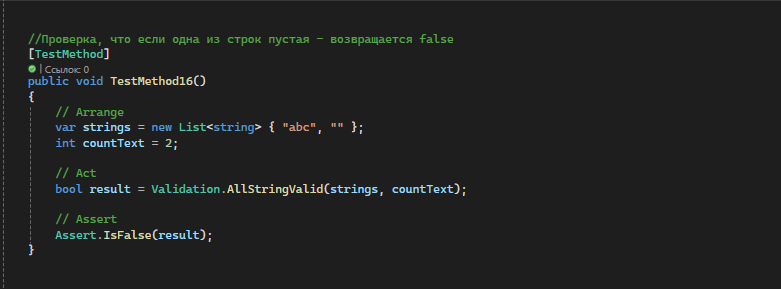
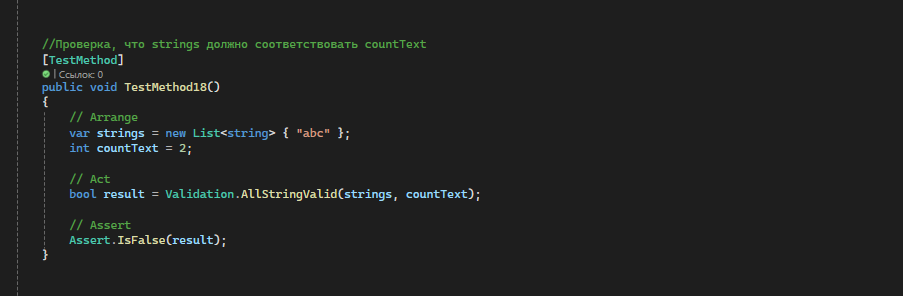


Рис. 32.



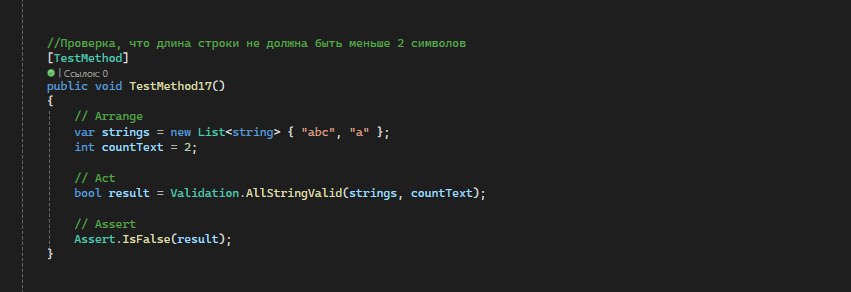
Рис. 33.

Рис. 34.

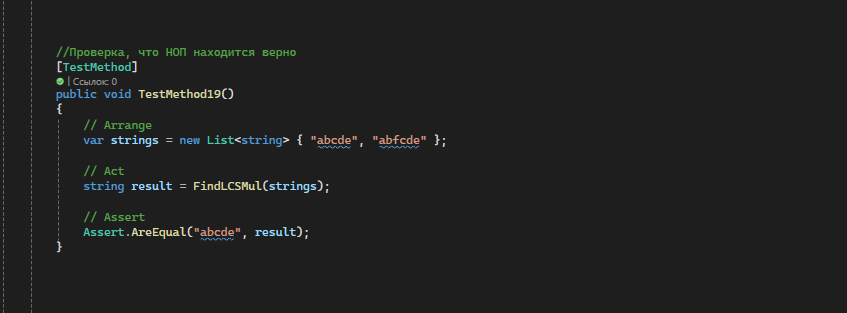


Рис. 35.

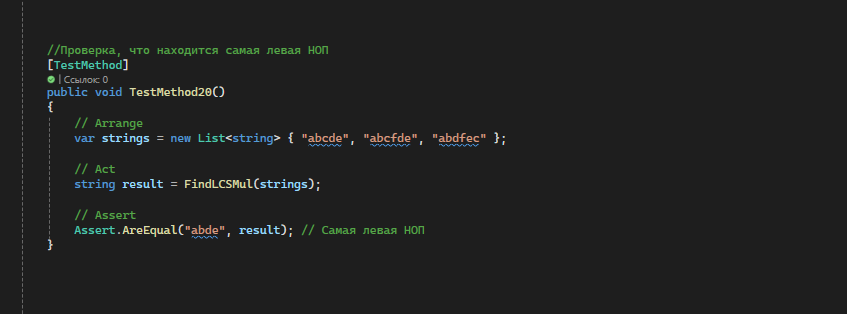


Рис. 36.

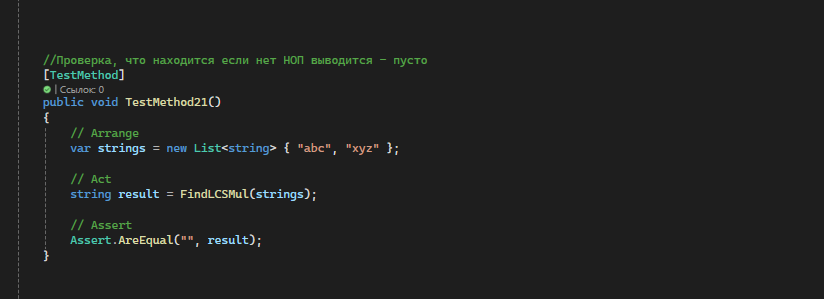


Рис. 37.

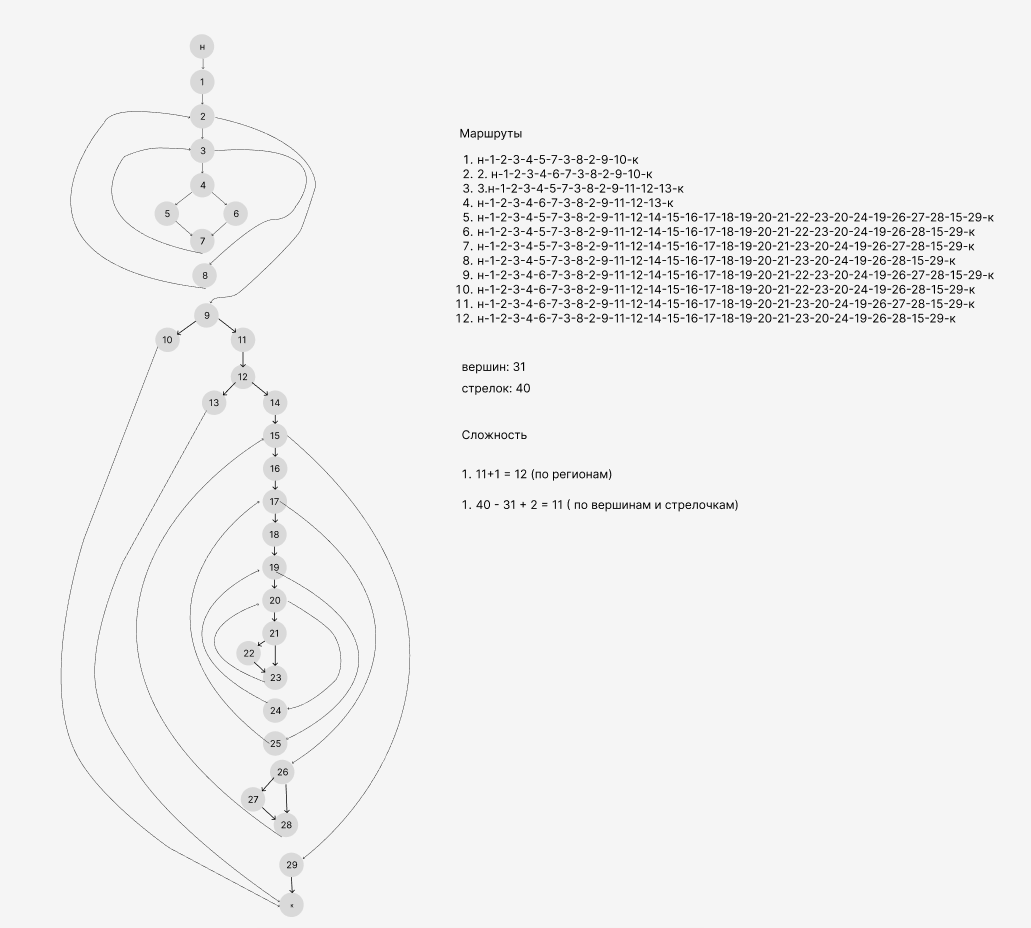
**1.6 Белый ящик**

Рис. 38.

**Практика 5. «Разработка тестирующей программы»**

## 1.1 Постановка задачи

**Задача:** Разработать объектно-ориентированное приложение, проводящее тестирование ученика, на основе информации, хранимой в файле.

## Входные данные:

**StudentName**– имя студента; строка, string

**StudentGroup**– группа студента; строка, string

**questions.txt**– файл с тестом; файл, txt

**Выходные данные:**

**results.txt**– файл с результатом теста; файл, txt

**1.2 Код методов: организующих работу с checBox**

CheckedListBox – нужен для заполнения вариантами ответов на вопросы.

Данный метод показывает вопросы и включает в себя: обновление текущего вопроса; заполнение CheckedListBox вариантами ответов; очистка выбранных ответов; показ изображения.

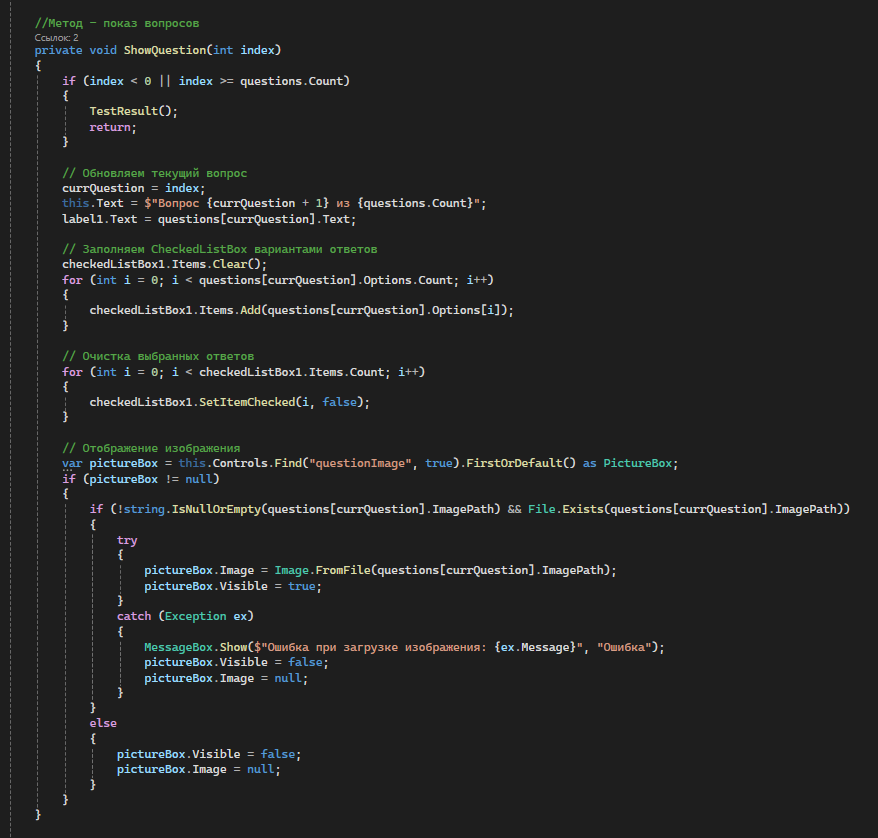


Рис. 1.

* 1. **Код методов: редактирования и удаления вопросов**

Кнопка – добавить новый вопрос включает в себя: текст вопроса; варианты ответов (через запятую); checkedListBox1 с правильными ответам; наличие изображения

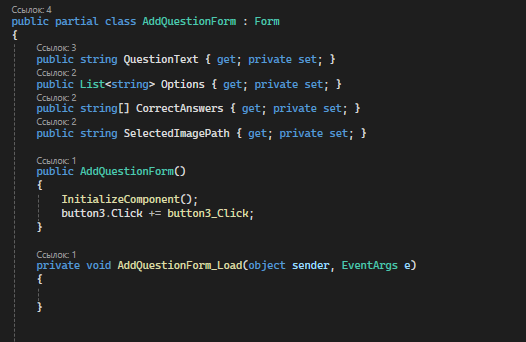


Рис.2.



Рис.3.

Рис.4.

Кнопка – удаление вопроса включает в себя: выбор вопроса в listBox1 и его удаление по индексу.

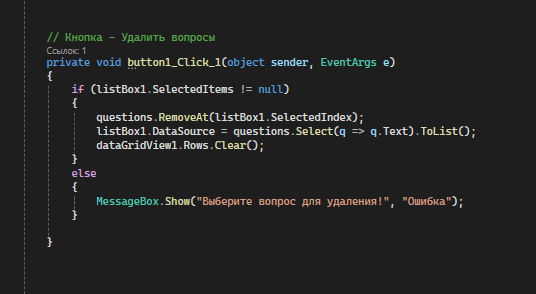


Рис.5.

**1.4 Код методов: добавления при помощи OpenFileDialog**

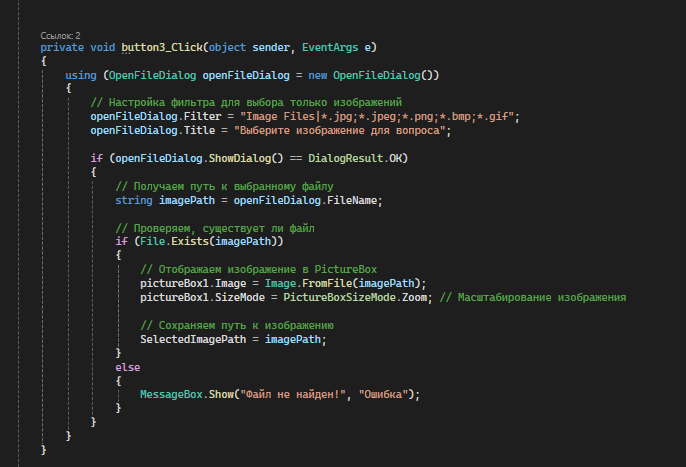
OpenFileDialog – служит для выбора изображения через проводник.

Рис.6.

**Практика 12. «Работа с БД индивидуальное задание».**

**1.1 Постановка задачи**

Спроектировать таблицы для практики 20.

Удаление/Редактирование.

Выборка по запросу.

**1.2 Таблицы**

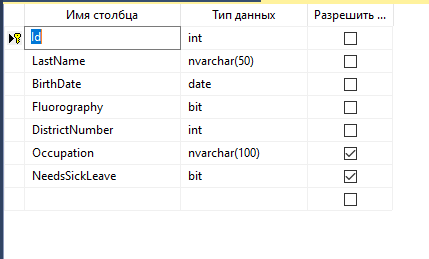


Рис. 1. Таблица «Взрослый пациент» в режиме Проекта.

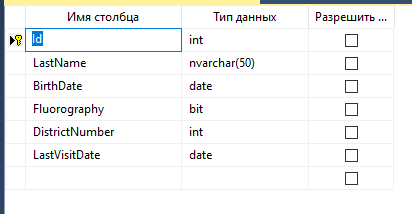
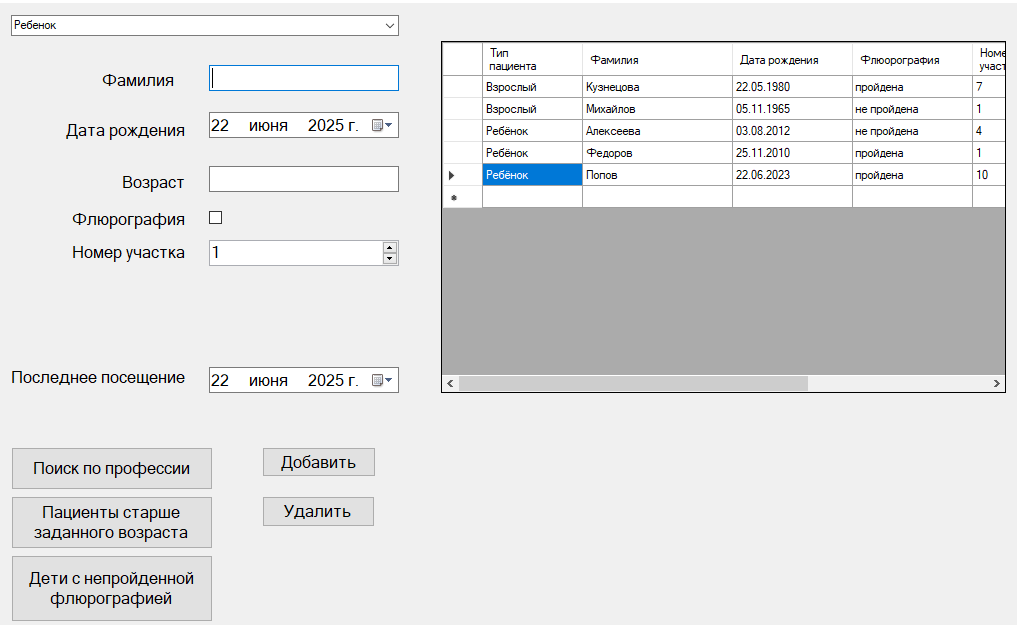
****

Рис. 2. Таблица «Ребенок пациент» в режиме Проекта.

**1.3 Интерфейс приложения**

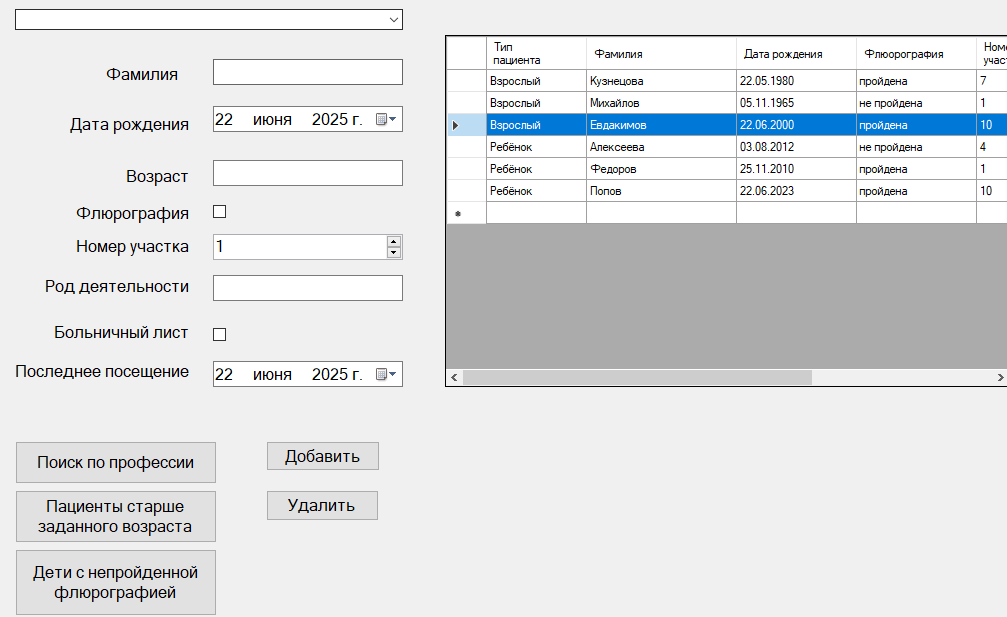
Рис. 3. Интерфейс добавления «Пациента ребенка»

Рис. 4. Интерфейс добавления «Взрослого пациента»

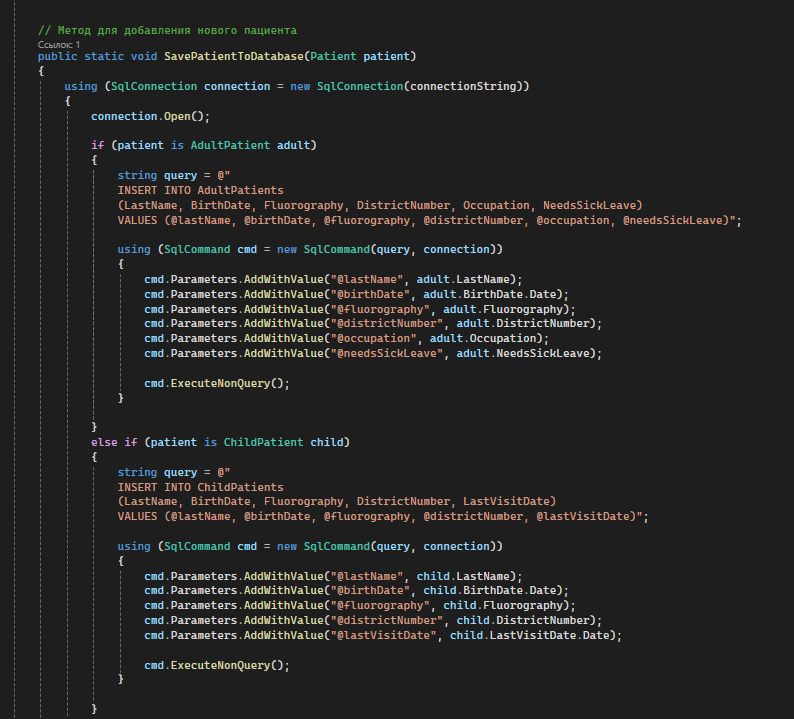
**1.4 Метод добавления**

Рис. 5.

**1.5 Метод изменения**

Рис. 6.

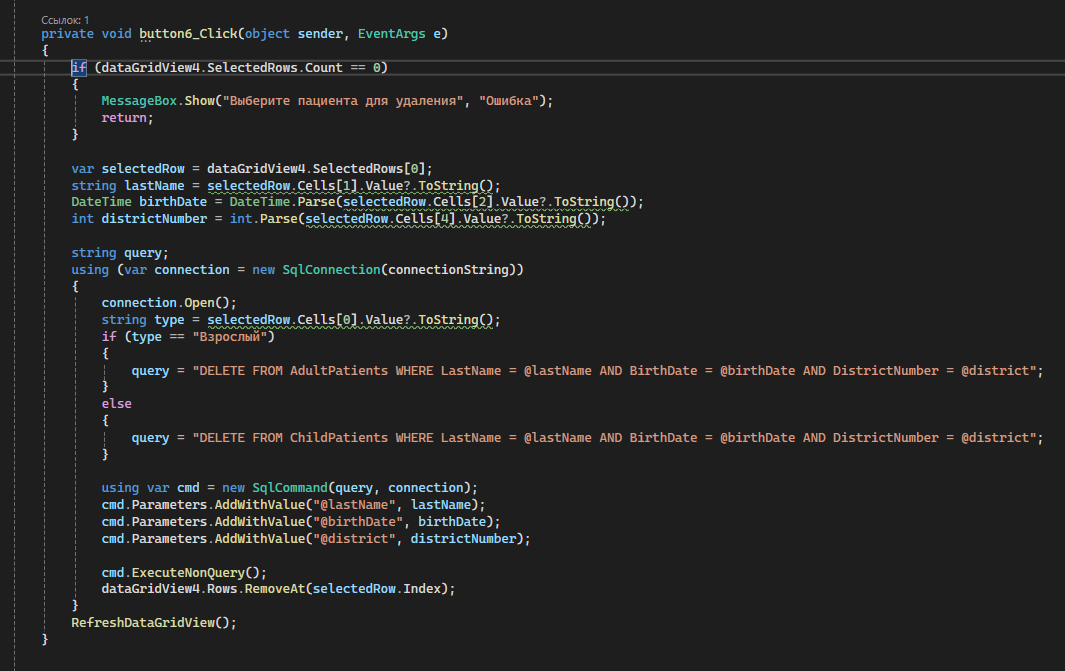
**1.6 Метод удаления**

Рис. 7.

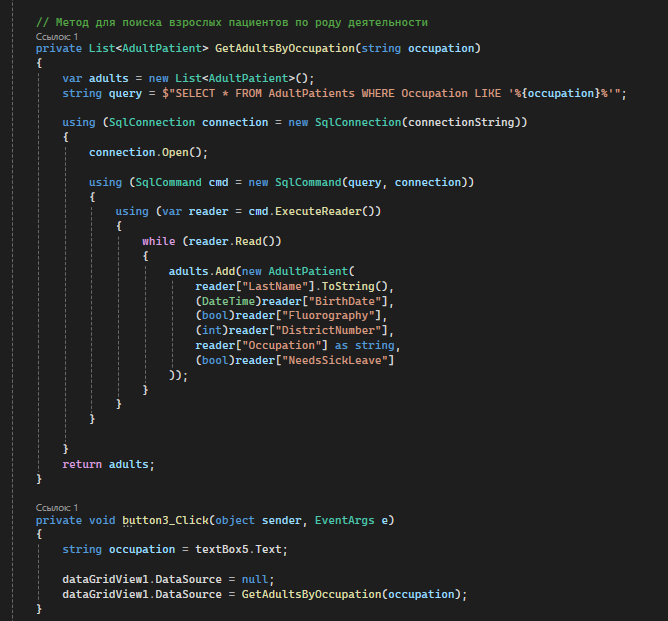
** 1.7 Метод: Поиск взрослых пациентов по роду деятельности**

Рис. 8.

**1.8 Метод: Поиск детей, у которых не пройдена флюорография**

Рис. 9.

**1.9 Метод: Поиск пациентов старше заданного возраста**

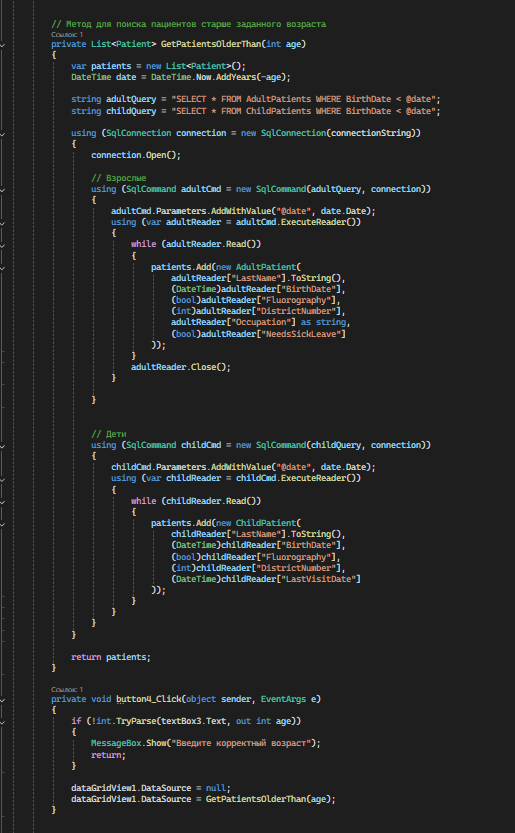
****

Рис. 10.

**Практика 10-11. «Работа с БД общее задание».**

**1.1 Постановка задачи**

Разработать приложение для обеспечения безопасности данных с формой входа/регистрации, смены пароля, режимом пользователя и администратора, реализовав возможность заходить под разными аккаунтами с хранением информации о пользователях в базе данных.

**1.2 Таблицы**

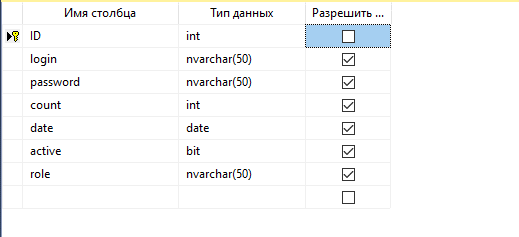
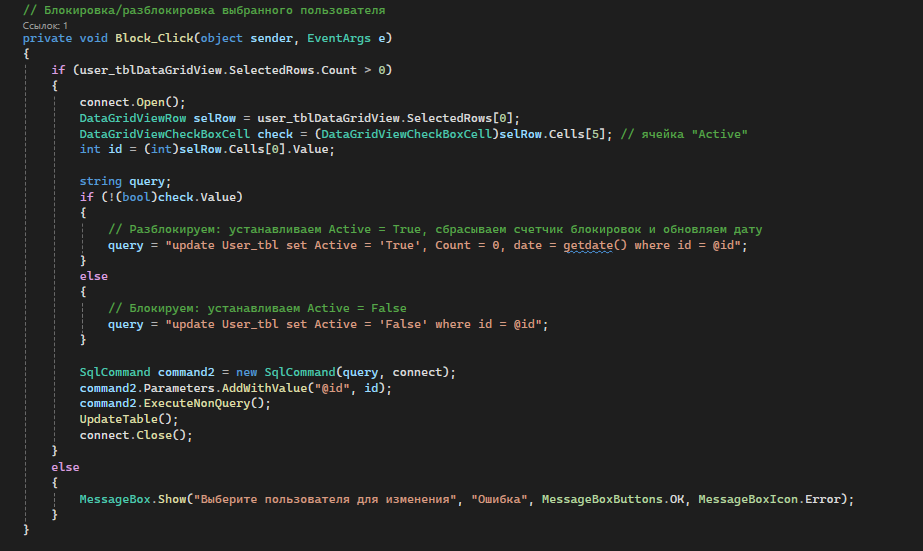
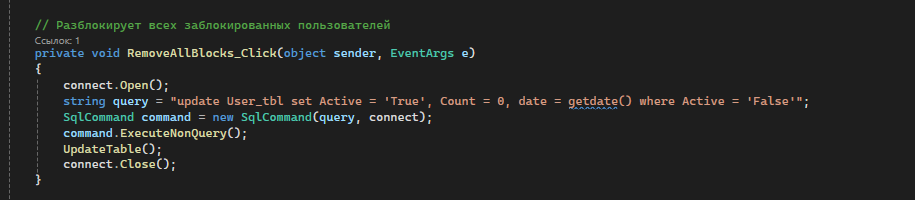


Рис. 1. Таблица User\_tbl в режиме «Проект».

**1.3 Код методов**

****Рис. 2.

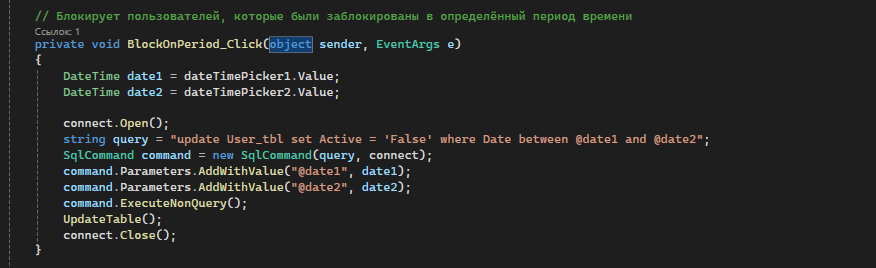
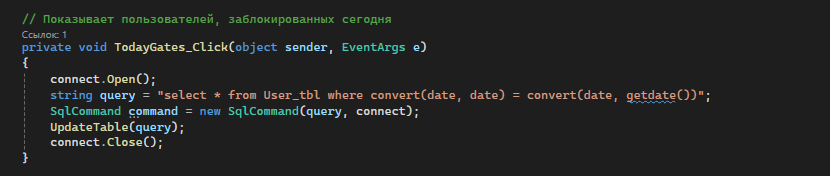
****Рис. 3.

Рис. 4.

Рис. 5.****