

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ

ПРОЕКТ ПО ПРАКТИКУМУ

Зеркальная комната

Поляков Андрей 341/1

9 марта 2022 г.

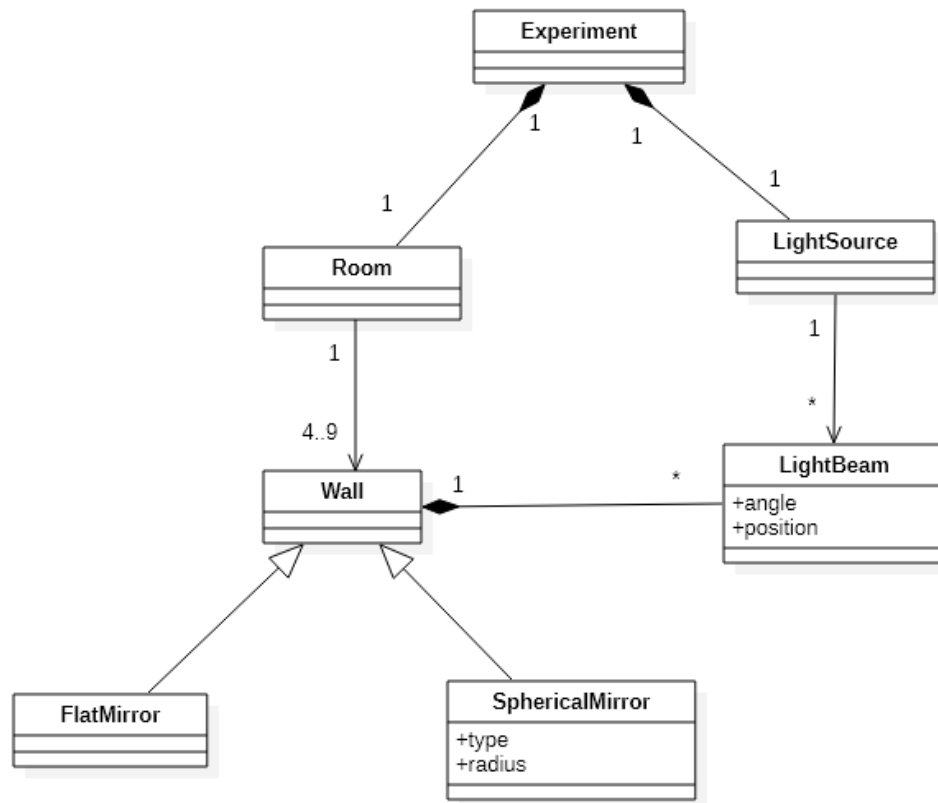
Содержание

1	Постановка задачи	2
2	Диаграмма классов	3
3	Текстовые спецификации интерфейса	4
4	Диаграмма объектов	6
5	Инструментальные средства	7
6	Описание файловой структуры системы	8
7	Пользовательский интерфейс	9
7.1	Окно «Зеркальная комната»	9
7.2	Окно «Параметры зеркала»	10
8	Описание проведенных экспериментов	11

1 Постановка задачи

Зеркальная комната представляет в плане произвольный замкнутый M -угольник ($4 \leq M \leq 9$), каждая стена – плоское или сферическое зеркало. Для проведения экспериментов необходимо определить для каждой стены комнаты вид зеркала (плоское или сферическое), а для каждого сферического зеркала – его тип (вогнутое или выпуклое) и радиус кривизны. Основная функция программной системы – проведение оптического эксперимента, при котором из некоторой точки на одной из стен комнаты, под определенным углом к этой стене (угол может варьироваться от 0 до 180 градусов) выпускается луч света, и затем показывается его путь внутри комнаты с учетом отражений от зеркал. Траектория луча определяется физическими законами отражения от зеркальных поверхностей. Цель моделирования – подбор пользователем системы параметров зеркал и исходного угла выпущенного луча, при которых луч, отражаясь от зеркальных стен, попадает в нужную точку (зону) комнаты. При визуализации оптического эксперимента должен быть показан план комнаты и изображен путь луча в комнате. Пользователь системы должен иметь возможность:

2 Диаграмма классов



3 Текстовые спецификации интерфейса

```
public static class Experiment
{
    public static Random Random = new Random();
    public static Room Room = new Room();
    public static LightSource LightSource = new LightSource();

    // Инициализация комнаты начальными параметрами
    public static void InitializeRoom(Form1 form1, int wallsCount);
}

public class Room
{
    public List<Wall> Walls = new List<Wall>();
}

public abstract class Wall
{
    public Point P1;
    public Point P2;
}

public class FlatMirror : Wall
{
}

public class SphericalMirror : Wall
{
    public int Radius;
}

public class LightSource
{
    public List<LightBeam> Beams = new List<LightBeam>();
}

public class LightBeam
{
    public int Position;
```

```

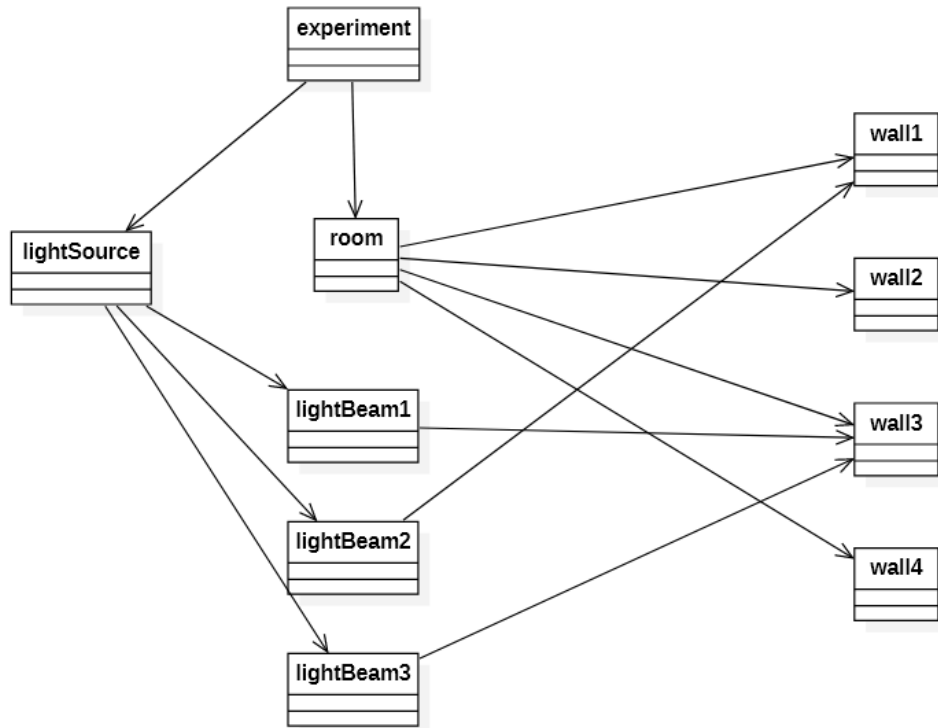
        public int Angle;
        public Wall Wall;
    }

    public partial class Form1 : Form
    {
        public System.Timers.Timer timer = new System.Timers.Timer(30);
        private bool down = false;
        private int pointIndex = -1;
        private Point old;
        private int segmentIndex = -1;
        private Point source;

        // Расчет лучей по таймеру
        private void Timer_Elapsed(object sender, System.Timers.ElapsedEventArgs e);
        // Кнопка запуска моделирования
        private void buttonStart_Click(object sender, EventArgs e);
        // Отрисовка моделируемой комнаты
        private void drawPanel_Paint(object sender, PaintEventArgs e);
        // Проекция точки на прямую
        private Point project(Point line1, Point line2, Point toProject);
        // Расстояние между точками
        private double distance(Point p1, Point p2);
        // Проверка на вложенность точки в прямоугольник
        private bool inside(Point p, Point p1, Point p2);
        // Обработка начала нажатия
        private void drawPanel_MouseDown(object sender, MouseEventArgs e);
        // Обработка перемещающегося нажатия
        private void drawPanel_MouseMove(object sender, MouseEventArgs e);
        // Обработка окончания нажатия
        private void drawPanel_MouseUp(object sender, MouseEventArgs e);
    }

```

4 Диаграмма объектов



5 Инструментальные средства

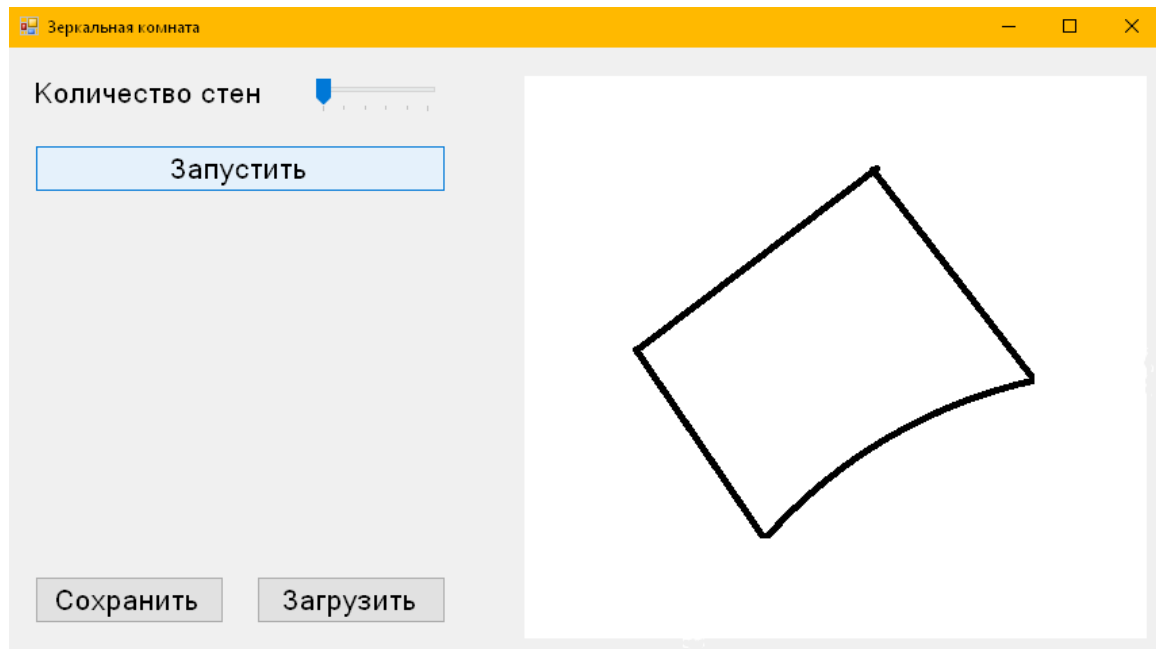
1. Язык разработки - C
2. Среда разработки - Microsoft Visual Studio .NET Framework 4.7.2
3. Используемые библиотеки:
 - System.Windows.Forms

6 Описание файловой структуры системы

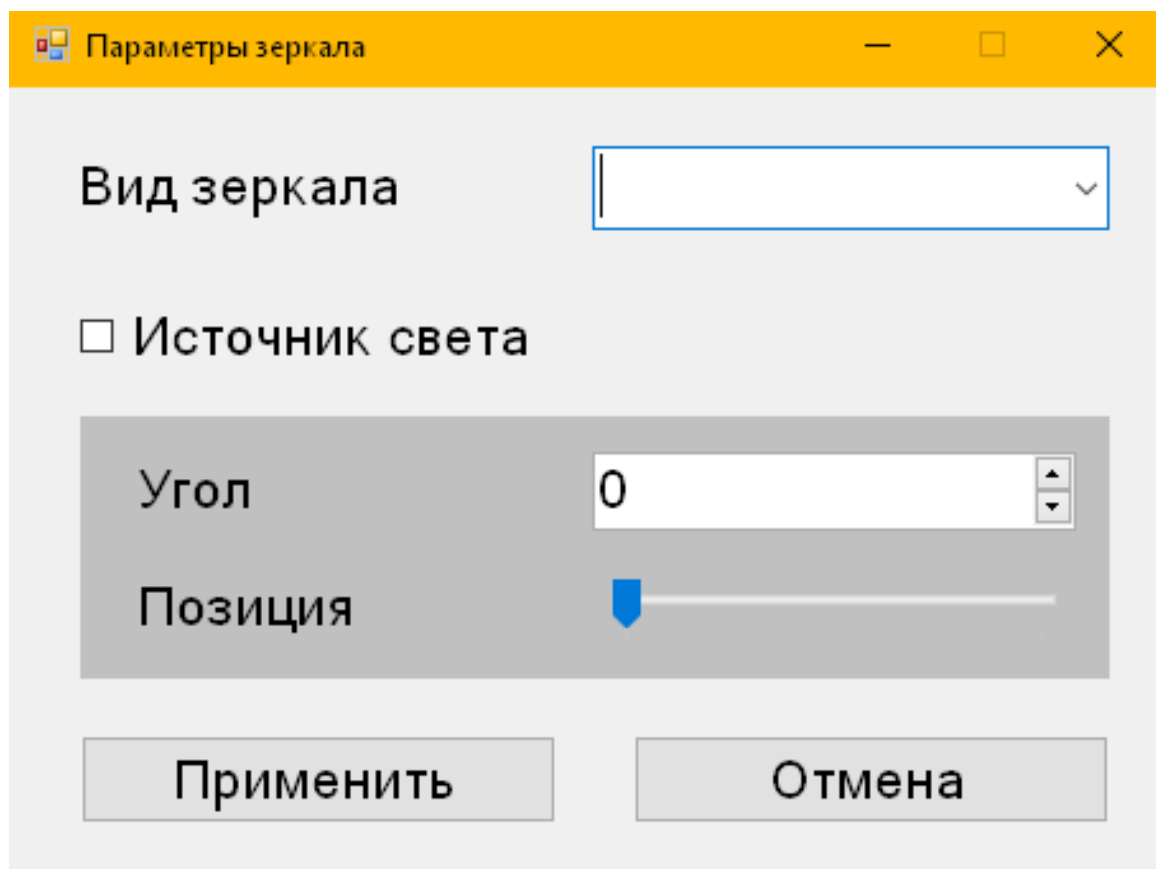
1. Program.cs - главный файл, запуск программы
2. Experiment.cs - основной класс эксперимента
3. Room.cs - класс моделируемой комнаты
4. Wall.cs - абстрактный класс стены
5. FlatMirror.cs - класс зеркальной стены, наследующийся от Wall
6. SphericalMirror.cs - класс сферической стены, наследующийся от Wall
7. LightSource.cs - класс со списком лучей
8. LightBeam.cs - класс отдельного луча света от стены до стены
9. Form1.cs - форма основного окна моделирования
10. Form2.cs - форма редактирования параметров стены

7 Пользовательский интерфейс

7.1 Окно «Зеркальная комната»



7.2 Окно «Параметры зеркала»



The image shows a software window titled "Параметры зеркала" (Mirror Parameters). The window has a yellow title bar with standard Windows controls (minimize, maximize, close). The main area is light gray and contains the following elements:

- A label "Вид зеркала" (Mirror View) followed by a dropdown menu that is currently empty.
- A checkbox labeled "Источник света" (Light Source) which is currently unchecked.
- A gray-shaded rectangular area containing:
 - A label "Угол" (Angle) followed by a numeric input field showing the value "0".
 - A label "Позиция" (Position) followed by a horizontal slider control with a blue triangular handle positioned at the left end.
- At the bottom, there are two buttons: "Применить" (Apply) on the left and "Отмена" (Cancel) on the right.

8 Описание проведенных экспериментов