

## Лабораторная работа №10. ФИНАЛЬНАЯ.

Ну что, сыночки, добрались до последней лабы?)

Тут теории не будет, но будет большая финальная задача, которая включает в себя применение всех накопленных знаний.

Было приятно с вами работать и душить вас на защите)))

### Задания по вариантам.

#### 1. Симулятор управления светофором:

Реализовать приложение для симуляции работы перекрестка с управлением светофором. Использовать паттерн *MVC*. Модель должна содержать состояние каждого сигнала (красный, желтый, зеленый) и длительность каждого этапа. *View* отображает текущие состояния светофоров на перекрестке. *Controller* обрабатывает кнопки "Начать/остановить" симуляцию.

#### 2. Редактор геометрических фигур:

Создать приложение для рисования геометрических фигур. Использовать паттерн *Команда* для реализации операций рисования, изменения цвета, изменения размера и удаления фигур. Фигуры должны быть представлены классами (круг, квадрат, треугольник).

#### 3. Игра "Змейка":

Реализовать классическую игру "Змейка" с использованием паттерна *Одиночка* для управления игровыми настройками (скорость, размер игрового поля). Интерфейс должен отображать текущий счет, длину змейки и кнопки управления игрой (пауза, старт).

#### 4. Симуляция солнечной системы:

Использовать паттерн *Наблюдатель* для реализации системы, где планеты вращаются вокруг Солнца. Когда Солнце изменяет свои свойства (цвет, интенсивность излучения), планеты обновляют свои цвета.

#### 5. Управление персонажем в 2D-игре:

Создать приложение для управления движением персонажа по экрану. Использовать паттерн *Стратегия* для изменения поведения персонажа (движение по прямой, движение с ускорением, прыжки). Реализовать графический интерфейс для выбора стратегии.

#### 6. Гармонические колебания:

Симулировать движение точки, совершающей гармонические колебания. Использовать паттерн *MVC*. Модель хранит параметры колебаний

(амплитуда, частота). View отображает траекторию движения. Controller обрабатывает пользовательский ввод для изменения параметров.

#### **7. Часы с будильником:**

Создать аналоговые или цифровые часы с возможностью установки будильника. Использовать паттерн *Наблюдатель*, где будильник наблюдает за часами и срабатывает при достижении заданного времени.

#### **8. Редактор диаграмм:**

Разработать приложение для рисования UML-диаграмм. Использовать паттерн *Приспособленец* для оптимизации хранения множества одинаковых элементов (линий, прямоугольников).

#### **9. Симулятор погоды:**

Создать приложение для отображения данных о погоде. Использовать паттерн *Наблюдатель* для обновления интерфейса при изменении данных (температура, давление, влажность).

#### **10. Система умного дома:**

Реализовать систему управления умным домом. Использовать паттерн *Команда* для управления устройствами (включение света, изменение температуры). Реализовать графический интерфейс для взаимодействия с системой.

#### **11. Графический редактор:**

Создать графический редактор с возможностью рисования различных фигур (линии, круги, прямоугольники). Использовать паттерн *Фабричный метод* для создания фигур. Реализовать меню для выбора типа фигуры и настройки ее свойств.

#### **12. Моделирование поезда на станции:**

Симулировать процесс прибытия и отправления поездов на станцию. Использовать паттерн *MVC*. Модель хранит состояние поездов (прибытие, отправление, остановка). View отображает состояние станции. Controller управляет симуляцией.

#### **13. Игра "Арканоид":**

Реализовать игру "Арканоид". Использовать паттерн *Стратегия* для изменения поведения платформы (ускорение, расширение). Реализовать графический интерфейс для отображения состояния игры.

#### **14. Симулятор очереди в банке:**

Создать приложение для симуляции работы банка. Использовать паттерн *Одиночка* для управления настройками банка (количество окон, среднее время обслуживания). Реализовать графический интерфейс для отображения очереди и текущего состояния окон.

### **15. Редактор текста с функцией отмены:**

Реализовать текстовый редактор с функцией отмены/повтора действий (ввод текста, удаление текста). Использовать паттерн *Команда* для реализации функциональности отмены.