#### Лабораторная работа 3

#### ****Общая характеристика****

Данный код представляет собой реализацию графического приложения для обработки изображений с использованием фреймворка **Qt**. Основная функциональность программы заключается в загрузке изображений и их обработке с использованием различных фильтров, включая:

* Преобразование в оттенки серого.
* Увеличение яркости в цветовом пространстве HSV.
* Инверсия цветов в пространстве RGB.
* Локальное бинарное преобразование (методы Бернсена, медианный и адаптивный).

Программа имеет интерфейс с кнопками, которые вызывают соответствующие функции для обработки изображения.

#### ****Основные компоненты****

**Класс** MainWindow

* + Унаследован от QMainWindow и отвечает за создание пользовательского интерфейса и обработку событий.
  + Подключены сигналы кнопок (например, loadButton, procGrey) к слотам для загрузки изображения и применения фильтров.

**Функции обработки изображения**

**Загрузка изображения**:

void MainWindow::loadImage()

Позволяет пользователю выбрать и загрузить изображение через диалоговое окно QFileDialog.

**Обработка фильтров**:

void MainWindow::applyFilter()

Вызывает соответствующий фильтр в зависимости от нажатой кнопки.

**Методы фильтрации**:

* + - applyBernsenThreshold: Метод Бернсена с использованием порогового значения контраста.
    - applyLocalMedianThreshold: Медианный локальный порог.
    - applyAdaptiveThreshold: Адаптивный порог с учетом локального среднего.
    - processInRGB: Инверсия цвета в RGB.
    - processInHSV: Увеличение яркости в пространстве HSV.
    - convertToGray: Преобразование в оттенки серого.

#### ****Подробное описание функций****

**Цветовая обработка**

* + processInRGB Инвертирует цвета изображения, вычитая значения каналов RGB из 255.
  + processInHSV Повышает яркость пикселей в HSV-пространстве, не изменяя тон и насыщенность.

**Преобразование в оттенки серого**

* + convertToGray Переводит изображение в формат Grayscale8, используя стандартный метод яркости.

**Преимущества**:

* Простая и четкая структура кода.
* Модульность: каждая функция отвечает за одну задачу.
* Использование Qt обеспечивает гибкость интерфейса.