Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Программирование на языках высокого уровня

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему

ПРОГРАММА «ФОТОРЕДАКТОР»

БГУИР КП 1-40 02 01 203 ПЗ

Студент: гр. 250502 Болашенко В.С.

Руководитель: Богдан Е. В.

МИНСК 2023

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc149522755)

[1 Обзор литературы 4](#_Toc149522756)

[2 Функциональное проектирование 5](#_Toc149522757)

# ВВЕДЕНИЕ

С++ — компилируемый язык программирования общего назначения. Он поддерживает такие парадигмы программирования, как процедурное программирование, объектно-ориентированное программирование (ООП), обобщенное программирование.

Язык программирования широко используется для разработки программного обеспечения: создание разнообразных прикладных программ, разработка операционных систем, драйверов устройств, а также видеоигр и многое другое.

Также язык программирование С++ позволяет подключать фреймворки, которые расширяют возможности языка: позволяют создавать оконные приложения, игры, обрабатывать фото- и видеоматериалов и др. Примерами крупных фреймворков являются Qt (для разработки ПО) и OpenCV (обработка изображений, компьютерное зрение).

ООП — подход к программированию как к моделированию информационных объектов, решающий на новом уровне основную задачу структурного программирования: структурирование информации с точки зрения управляемости, что существенно улучшает управляемость самим процессом моделирования, что, в свою очередь, особенно важно при реализации крупных проектов.

Основные принципы структурирования в случае ООП связаны с различными аспектами базового понимания предметной задачи, которое требуется для оптимального управления соответствующей моделью:

* абстракция для выделения в моделируемом предмете важного для решения конкретной задачи по предмету, в конечном счёте — контекстное понимание предмета, формализуемое в виде класса;
* инкапсуляция для быстрой и безопасной организации собственно иерархической управляемости: чтобы было достаточно простой команды «что делать», без одновременного уточнения как именно делать, так как это уже другой уровень управления;
* наследование для быстрой и безопасной организации родственных понятий: чтобы было достаточно на каждом иерархическом шаге учитывать только изменения, не дублируя всё остальное, учтённое на предыдущих шагах;
* полиморфизм для определения точки, в которой единое управление лучше распараллелить или наоборот — собрать воедино.

# Обзор литературы

* 1. **Обзор методов и алгоритмов решения поставленной задачи**

# Функциональное проектирование

В данном разделе описываются входные и выходные данные программы, диаграмма классов, а также приводится описание используемых классов и их методов.

**2.1 Структура входных и выходных данных**

В качестве входных данных программа принимает изображение типов .png и .jpg над которым будет проводится обработка.

В качестве выходных данных формируется обработанное изображение с выбранным пользователем форматом.

**2.2 Разработка диаграммы классов**

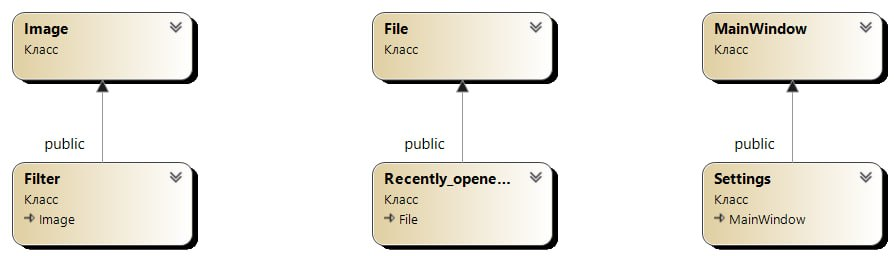


Рисунок — диаграмма классов (WIP)

**2.3 Описание классов**

**WIP**