

Эскизный проект программного обеспечения px2txt (p2t)

1. Постановка задачи.

1.1. Цель проекта.

Разработать программный комплекс, предназначенный для преобразования изображения в формате JPEG/PNG в ASCII-символы с последующей печатью на чёрно-белом принтере.

Разработать прикладное программное обеспечение для преобразования изображения в формате JPEG/PNG в ASCII-символы с возможностью вывода результата на чёрно-белый принтер. Программа должна позволять пользователю загрузить изображение, выбрать параметры обработки (разрешение, шрифт, уровень контрастности), выполнить предварительный просмотр изображения после перевода в ASCII и печать полученного текстового изображения.

1.2. Общие сведения о программном комплексе.

Название проекта	px2txt (p2t)
Классификация программы (по ГОСТ 19.101-2024)	Прикладная программа для обработки изображений
Область применения	Стилизация изображений, создание ASCII-арта из фотографий, печать текстовых версий изображений на чёрно-белом принтере.
Целевая платформа	Операционная система Microsoft Windows 10/11
Используемые технологии	Язык программирования C++ стандарта C++14
	Библиотека FreeImage (по лицензии GPLv2)
	Библиотека Windows API
	Библиотека GDI/Direct2D

1.3. Ожидаемый результат.

Создание программы, которая преобразует изображение в ASCII-символы, позволяет предварительно посмотреть результат, настроить параметры и выполнить печать ASCII-арта на чёрно белом принтере.

2. Перечень компонентов и связей между ними.

2.1. Общее описание структуры программы.

Программа состоит из 6 основных модулей, каждый из которых выполняет определённые функции и взаимодействует с другими компонентами системы. Все модули объединены в единое приложение через интерфейс пользователя и обмен данными.

2.2. Основные программные модули:

2.2.1. Модуль загрузки изображения

Модуль загрузки изображения получает на входе информацию о названии и пути к файлу в формате JPEG или PNG, после чего загружает изображение (информацию о нём: разрешение, значения цвета пикселей) в память и передаёт на выход.

2.2.2. Модуль предварительной обработки изображения

Получает на входе загруженное в память изображение (информацию о нём: разрешение, значения цвета пикселей), выполняет его предобработку: перевод изображения в чёрно-белый формат (градации серого), нормализацию значений пикселей в диапазоне $[0;1]$ для последующего сопоставления. На выходе передаёт информацию о пикселях входного изображения в диапазоне $[0;1]$. В модуле предусматривается возможность изменения размера входного изображения (его масштабирование к меньшему или большему размеру) при определённом разрешении.

2.2.3. Модуль сопоставления символов

Получает на входе информацию о пикселях входного изображения в диапазоне $[0;1]$. Производит сопоставление значений пикселей с символами ASCII, формируя символьное представление входных числовых данных. Возвращает ASCII-символьное представление в виде текстового файла, выполняет его сохранение.

2.2.4. Модуль предпросмотра

Получает на входе текстовый файл с ASCII-символами, параметры печати файла, формирует предварительный вид печатного документа и форматирует текстовый файл в соответствии с параметрами печати. Возвращает отформатированный текстовый файл.

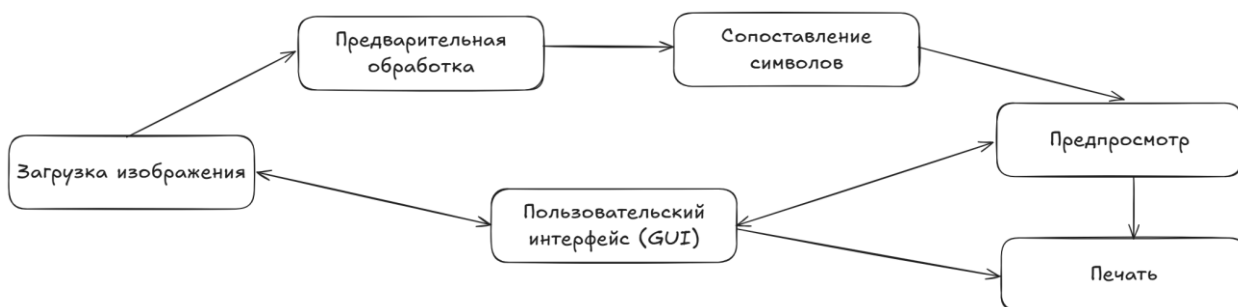
2.2.5. Модуль печати

Получает на входе текстовый файл с ASCII-символами, пользовательские параметры печати: формат бумаги, масштаб, количество страниц, число копий, параметры страницы печати (ориентация, переплёт, поля). Выполняет печать текстового файла на чёрно-белом принтере с заданными пользователем параметрами печати.

2.2.6. Пользовательский интерфейс (GUI)

Получает логи и данные от других модулей, отправляет им необходимые данные и управляющие инструкции. Обладает возможностью ввода или выбора входного файла изображения, имеет окно предпросмотра печатаемого файла, настройку параметров печати и вызов операции начала или отмены печати.

2.3. Связи между компонентами



Через **Пользовательский интерфейс (GUI)** вводится или выбирается с помощью диалогового окна файл изображения. Информация о пути файла передаётся в **Модуль**

загрузки изображения, где происходит его загрузка и передача в **Модуль предварительной обработки**.

Модуль предварительной обработки выполняет обработку изображения и передаёт **Модулю сопоставления символов**, который формирует и сохраняет текстовый файл с ASCII-символами.

Сформированный **Модулем сопоставления** текстовый файл передаётся **Модулю предпросмотра**, в котором формируется предварительный вид печатного документа и передаётся в **Пользовательский интерфейс** для отображения пользователю на дисплее, файл с учётом форматирования передаётся в модуль печати.

Пользовательский интерфейс (GUI) передаёт в **Модуль печати** параметры, описанные в 2.2.5 и 2.2.6, и непосредственно инструкцию для начала печати или её отмены.

Параметры печати, описанные в 2.2.5 и 2.2.6, также передаются **Пользовательским интерфейсом** в **Модуль предпросмотра**, меняя предварительный вид печатного документа и текстовый файл с учётом нового форматирования.

2.4. Классификация компонентов (по ГОСТ 19.101-2024)

Модуль загрузки изображения	Прикладной модуль (обработка данных)
Модуль обработки изображения	Прикладной модуль (преобразование данных)
Модуль выбора символов	Прикладной модуль (алгоритмическая логика)
Модуль предпросмотра	Прикладной модуль (визуализация)
Модуль печати	Системный модуль (интеграция с ОС)
Пользовательский интерфейс (GUI)	Прикладной модуль (взаимодействие с пользователем)

3. Оценка затрат на разработку.

Оценка ведётся по трудозатратам и используемым ресурсам. В качестве исходных данных принят объём кода около 2500–3000 строк и срок разработки до 5 месяцев.

Этап разработки	Время (чел.-дней)
Реализация модуля загрузки изображения	10
Реализация модуля обработки	20
Реализация модуля сопоставления	30
Реализация модуля предпросмотра	7
Реализация модуля печати	12
Разработка пользовательского интерфейса	20
Тестирование и отладка	30
Документирование	7

Итого: 136 чел.-дней по предварительной оценке.

4. План реализации.

Ниже приведен предварительный план реализации программного обеспечения с приблизительным распределением времени на поставленные задачи.

Этап разработки	Срок выполнения
1. Реализация модулей	1-3-й месяцы
2. Интеграция	4-й месяц
3. Тестирование	4-5-й месяцы
4. Документирование	4-5-й месяцы (параллельно с этапом 3)