Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему

ASCII-графика. Визуализация ASCII-изображений

Студент гр. 053502

Василевский С.В.

Руководитель ассистент кафедры информатики

Удовин И.А.

Минск 2021

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

УТВЕРЖДАЮ

Заведующий кафедрой Информатики

––––––––––––––––––––––––

(подпись)

Волорова Н.А. 2021 г.

ЗАДАНИЕ

по курсовому проекту

Студенту *Василевскому Сергею Витальевичу*  –––––––– –––––\_\_\_–

1. Тема работы *ASCII-графика. Визуализация ASCII-изображений*

2. Срок сдачи студентом законченной работы 3*01.06.2021 г*.–––   \_\_\_\_ \_\_\_\_\_

3. Исходные данные к работе *Операционная система Windows. Язык программирования Java.*

4. Содержание расчётно-пояснительной записки (перечень вопросов, которые подлежат разработке)

*Введение. 1. Анализ предметной области. 2. Разработка программного средства. 3. Демонстрация работы. Заключение. приложение.*

5. Консультант по курсовой работе *Удовин И.А.*

6. Дата выдачи задания *01.02.2021 г.*

7. Календарный график работы над проектом на весь период проектирования (с обозначением сроков выполнения и процентом от общего объёма работы):

*раздел 1, Введение к 28.02.2021г. – 10 % готовности работы;*

*раздел 2 к 15.03.2021г. – 30 % готовности работы;*

*раздел 3 к 15.04.2021г. – 60 % готовности работы;*

*раздел 4 к 10.05.2021г. – 80 % готовности работы;*

*Заключение к 20.05.2021г.– 90 % готовности работы; оформление пояснительной записки и графического материала к 24.05.2021г. – 100 % готовности работы.*

*Защита курсового проекта 00.06.2021 г.*

РУКОВОДИТЕЛЬ *Удовин И.А.*

(подпись)

Задание принял к исполнению *Василевский С.В 01.02.2021 г.* (дата и подпись студента)

Содержание

[ВВЕДЕНИЕ](#_Toc9767693) 5

[1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ](#_Toc9767694) 7

[1.1 Анализ аналогов программного средства](#_Toc9767695) 7

[1.1.1 Онлайн сервис преобразования изображений https://asciiart.club/](#_Toc9767696) 7

[1.1.2 Приложение Ascii-Generator 2.](#_Toc9767697) 7

[1.2 Стили ASCII-изображений](#_Toc9767700) 8

[1.3 Методы генерации ASCII-арта](#_Toc9767703) 9

[2 Постановка задачи](#_Toc9767708) 10

[3 РАЗРАБОТКА ПРОГРАММНОГО СРЕДСТВА](#_Toc9767709) 10

[3.1 Описание среды](#_Toc9767710) 10

[3.2 Алгоритм преобразования изображения](#_Toc9767714) 11

[3.3 Требования к программе 1](#_Toc9767715)2

[4 ДЕМОНСТРАЦИЯ РАБОТЫ 1](#_Toc9767716)2

[ЗАКЛЮЧЕНИЕ 1](#_Toc9767717)4

# Введение

Люди будут создавать искусство для любой новой технологии. Примером тому служит ASCII Art – искусство, созданное для общения людей с помощью рисунков созданных из символов клавиатуры компьютера в Интернете.

ASCII-графика является неотъемлемой частью Интернета, также как и смайлики, аватарки или интернет мемы. Мы говорим о фотографиях, сделанных из текста: букв, цифр и специальных символов из таблицы Юникода.

ASCII Art очень распространённое искусство в сегодняшнем современном мире интернет общения. Это и форумы, и социальные сети, и электронная почта.

 Однако искусство рисовать символами возникло задолго до появления компьютеров в Древнем Египте. Письменность древних египтян была основана на пиктографическом письме, а пиктограммы – это и есть своеобразные текстовые рисунки.

 В пиктографическом письме каждый рисунок или последовательность рисунков несут смысловую нагрузку. Из различных пиктографических систем зародилось два вида фонетической письменности: иероглифическое письмо Востока и буквенная письменность Запада. С помощью иероглифического письма можно образно и компактно выразить целую фразу одним графическим символом, а буквенная письменность динамична и универсальна.

 За несколько лет до появления печатных машинок, в 1865 году,  Люис Кэрролл в знаменитой книге «Приключения Алисы в стране чудес» разместил текст третьей главы в виде мышиного хвоста. Так изменилась форма графической поэзии.

 И уже в начале XXвека появляется поэзия в виде фигурных стихотворений французского поэта Гийома Аполлинера – каллиграммы. Аполлинер составлял некоторые свои стихи в виде рисунков из слов, что помогало раскрыть тему стихотворения и глубину эмоций поэта, развивали воображение и ассоциативное мышление. Каллиграммы объединили два вида письменности и послужили некоторым толчком к зарождению ASCII Art искусства.

 С появлением печатных машинок производители в 90-е годы XIX века для  привлечения новых покупателей стали устраивать конкурсы, на которых состязались машинистки и секретари. Там они проводили соревнования на скорость печати и на лучший рисунок из символов, созданный при помощи печатных символов. Так появились первые художники ASCII art. Самая известная художница того времени – это Флора Стейси. Она создала рисунок бабочки целиком созданный из печатных символов.

 В середине XX века появляется компьютер, и увлечение рисунками из символов набирает популярность. Рисование буквами становится очень популярным занятием в среде программистов. Рисунки символами широко используются при переписке и общении в сети. Появляются первые отголоски ASCII art – смайлики, улыбающиеся рожицы из скобок и точек. В ASCII art появляются различные стили и направления. Например: Line Style – рисование с помощью линий;  Solid Style – рисование буквами и другими символами.

Это был визуальный язык общения в первых языках программирования до появления WWW сетей. Это было время, когда для доступа в Интернет использовались модемы для коммутируемых телефонных линий. Искусство ASCII Art тогда казалось эстетически красивым и правильным для сред которые были на начальном этапе своего развития.

Аббревиатура ASCII расшифровывается как Американский стандартный код для обмена информацией (American Standard Code for Information Interchange), полный набор которого составляет 128 символов. Однако при рисовании символами используется порядка ста символов, а остальные – это служебные коды, применяемые при передаче данных и форматировании текста.

# АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

# Анализ аналогов программного средства

### Онлайн сервис преобразования изображений https://asciiart.club/

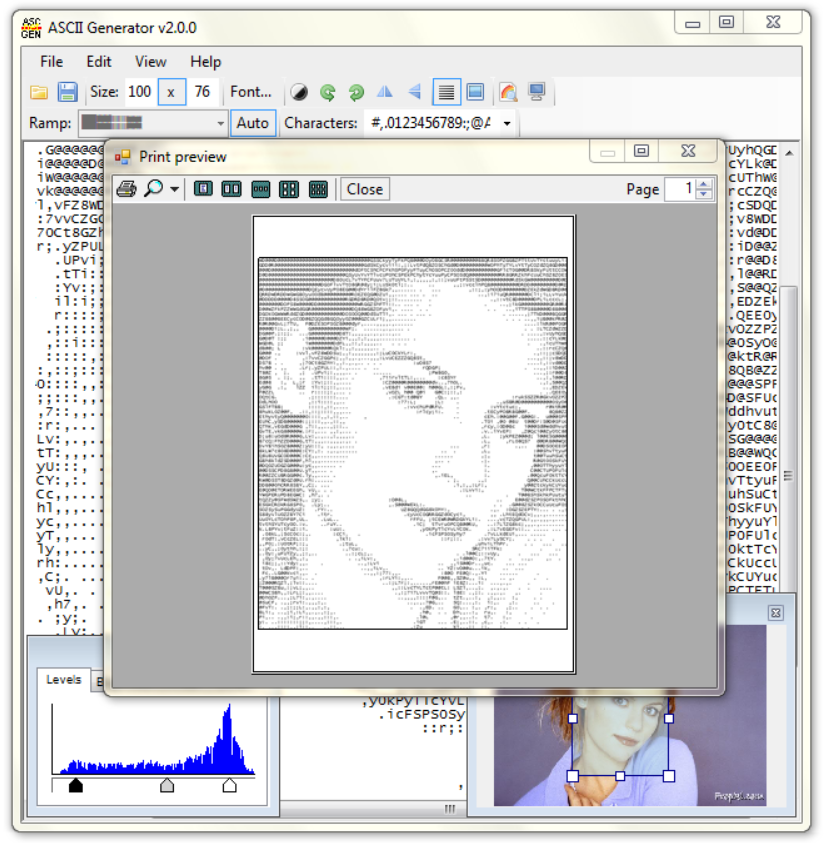
Данный сервис является удобным инструментом для преобразований изображений в ASCII-art объекты. Отличительной особенностью данного сайта является получение изображения сразу в чёрно-белом, цветном и адаптированным под HTML страницу форматах. Присутствует библиотека преобразованных ASCII-art изображений.



### **Приложение Ascii-Generator 2.**

Ascii Generator 2 (Ascgen2)-это бесплатное приложение для Windows для преобразования изображений в высококачественный художественный текст ASCII, который точно представляет исходное изображение. На официальном сайте присутствует подробная инструкция и информация о продукте (<https://ascgendotnet.jmsoftware.co.uk/>.).

Особенностью данного приложения является поддержание вывода для отображения в шрифтах переменной ширины, с настройкой вывода в режиме реального времени. Хотя поддержание официально не было приостановлено, обновления после 2015 года не выходят.



### **1.2 Стили ASCII-изображений**

В ASCII-арте можно использовать разные методы для получения различных художественных эффектов. Электронные схемы и схемы были реализованы на пишущей машинке или телетайпе и служили предлогом для ASCII.

Надпись «в стиле пишущей машинки», состоящая из отдельных буквенных знаков:



Штриховой рисунок, для создания фигур:



Сплошное искусство, для создания объектов с заливкой:



Затенение с использованием символов разной интенсивности для создания градиентов или контрастов:



### **Методы генерации ASCII-арта**

В то время как некоторые предпочитают использовать простые для создания ASCII-артов специализированные программы, такие как JavE , которые часто имитируют функции и инструменты редакторов растровых изображений. Для графических изображений Block ASCII и ANSI художник почти всегда использует специальный текстовый редактор, потому что для генерации требуемых символов на стандартной клавиатуре необходимо знать Alt-код для каждого символа. Например, Alt +178даст ▓, Alt +177даст ▒, а Alt +8даст ◘.

В специальных текстовых редакторах есть наборы специальных символов, назначенных существующим клавишам на клавиатуре. Популярные редакторы на основе DOS , такие как TheDraw и ACiDDraw, имели несколько наборов различных специальных символов, сопоставленных с функциональными клавишами , чтобы упростить использование этих символов. художник, который может переключаться между отдельными наборами символов с помощью основных сочетаний клавиш. PabloDraw - один из немногих специальных художественных редакторов ASCII / ANSI, которые были разработаны для Windows.

### **Постановка задачи**

Цель: применение теоретических и практических знаний для создания программы, которая преобразует и визуализирует изображение в набор ASCII символов.

В данном проекте главными задачами станут:

* Создание удобного пользовательского интерфейса для взаимодействия со всеми функциями проекта
* Возможность выбора и создания набора символов, используемого для вывода итогового изображения.
* Возможность вывода преобразованного изображения в режиме реального времени.
* Визуализация результата алгоритма.
* Сохранение получившегося ASCII-объекта в формате \*.txt.

### **РАЗРАБОТКА ПРОГРАММНОГО СРЕДСТВА**

### **Описание среды**

В качестве среды была выбрана платформа JavaFX.

JavaFX представляет инструментарий для создания кроссплатформенных графических приложений на платформе Java.

JavaFX позволяет создавать приложения с богатой насыщенной графикой благодаря использованию аппаратного ускорения графики и возможностей GPU.

С помощью JavaFX можно создавать программы для различных операционных систем: Windows, MacOS, Linux и для самых различных устройств: десктопы, смартфоны, планшеты, встроенные устройства, ТВ. Приложение на JavaFX будет работать везде, где установлена исполняемая среда Java (JRE).

JavaFX предоставляет большие возможности по сравнению с рядом других подобных платформ, в частности, по сравнению со Swing. Это и большой набор элементов управления, и возможности по работе с мультимедиа, двухмерной и трехмерной графикой, декларативный способ описания интерфейса с помощью языка разметки FXML, возможность стилизации интерфейса с помощью CSS, интеграция со Swing и многое другое.

На данный момент JavaFX представляет предпочтительный способ для создания графических приложений с помощью языка Java, который пришел на смену AWT и Swing. Также стоит отметить, что для работы с JavaFX вместо Java теоретически можно использовать любой язык программирования, который поддерживается JVM.

### **Алгоритм преобразования изображений**

Приложение должно автоматически преобразовывать изображение в текстовые символы, что в свою очередь является частным случаем векторного квантования. По этому логичным шагом будет сэмплирование изображения до оттенков серого с точностью менее 8-битной, а затем назначать символ для каждого значения.

Для решения поставленной задачи были использованы классы BufferedImage и Color.

Класс BufferedImage позволяет взаимодействовать с загруженным изображением. В нашем случае с помощью метода “\*.getRGB(x,y)”, мы получаем объект класса Color, который содержит всю необходимую информацию о цвете пикселя:



Имея объект класса Color мы можем получить значение оттенка серого данного пикселя, путём смешивания цветовых каналов в пропорциях 80/20:



В итоге, циклически проходя по всему изображению, мы получим его чёрно-белый аналог, который загрузим в объект класса BufferegImage для дальнейшего взаимодействия.

Для выявления самого яркого и самого тусклого из заданного множества символов, было принято решение рисовать каждый символ на объекте класса Canvas и аналогично проходить по каждому пикселю символа. Ключевым моментом является то, что в теперь мы берём только один из цветов, путём побитового сдвига(т. к. изображение символа будет чёрно-белым).



Далее нам остаётся только отсортировать словарь символом по параметру

totalBrightness(int) и установить соответствие между значение пикселя и его представлением в заданном списке символов.

### **3.3 Требования к программе**

К проекту предъявлены следующие требования

1. Интерфейс

Интерфейс программного средства должен содержать рабочую область, на которой будет располагаться итоговое изображение.

1. Функциональность

Пользователю представляется выбор символов для построения, инверсия цвета, масштабирование и сохранение изображения в виде текстового документа.

1. Наглядность

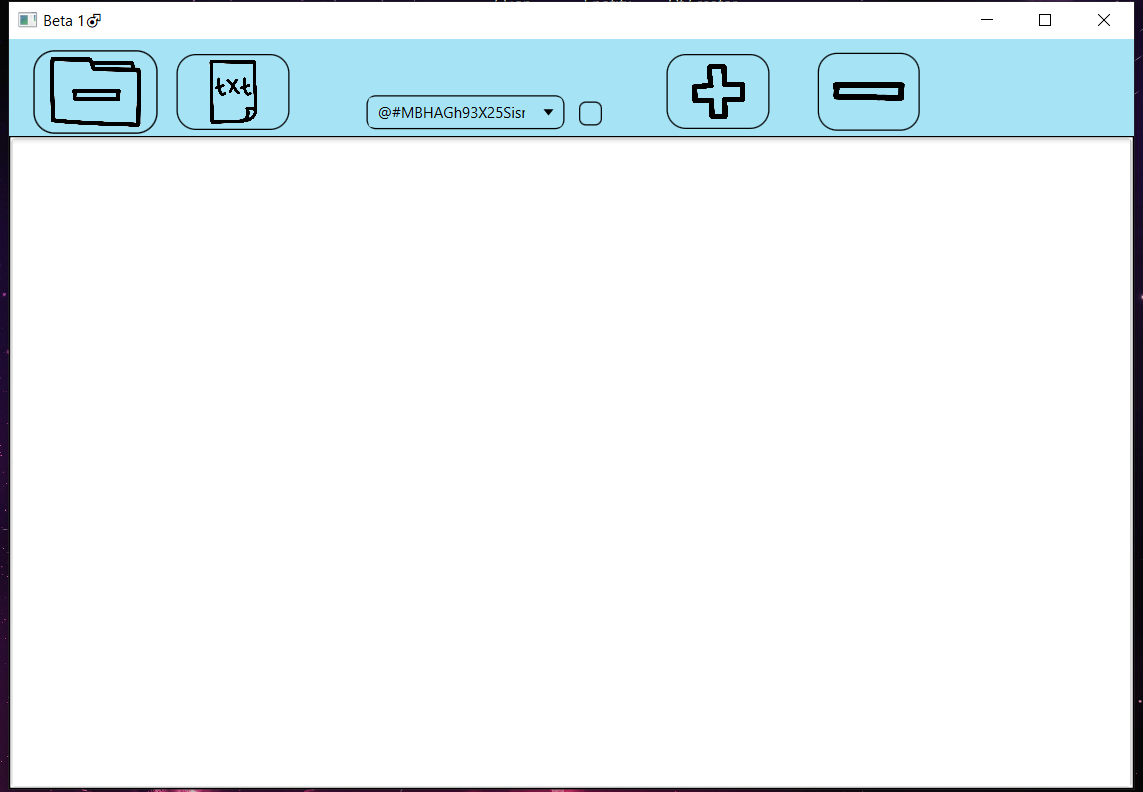
должен быть удобным и понятным для пользователя

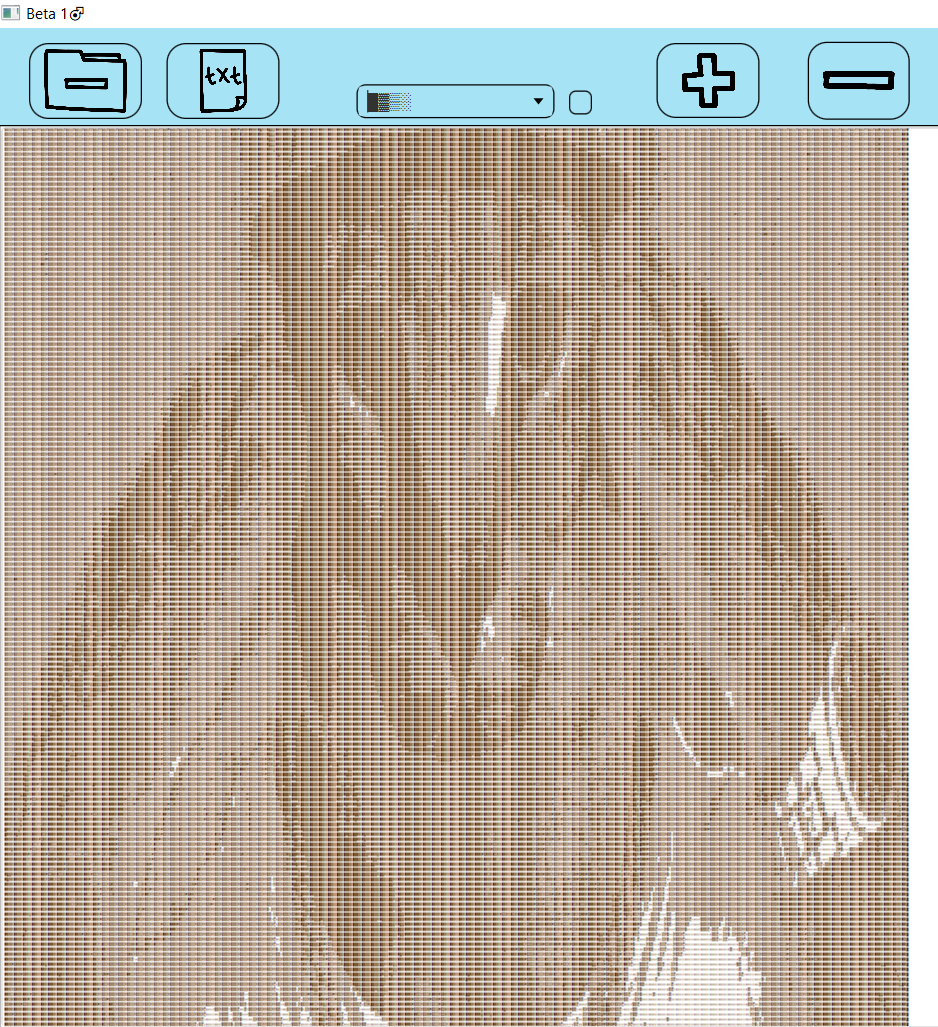
1. Среда разработки

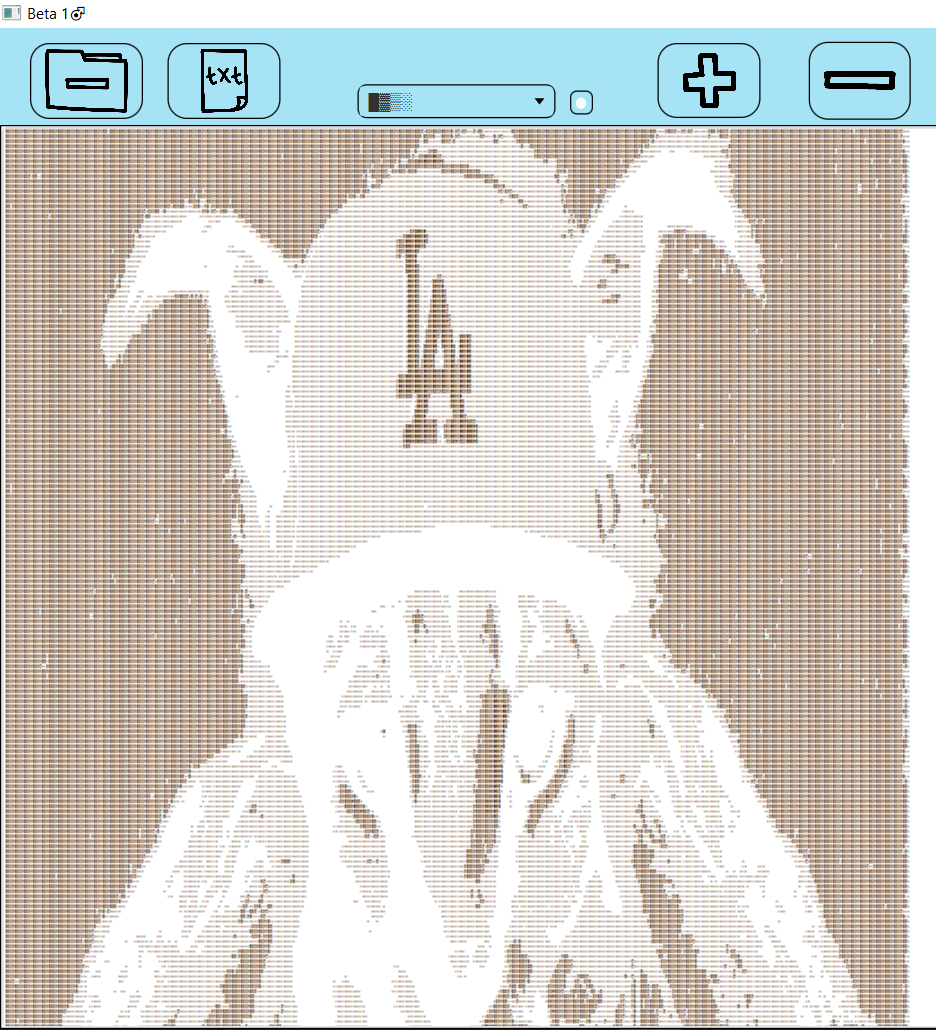
Средой разработки данного программного средства выбран InteliJ IDEA.

### **4. ДЕМОНСТРАЦИЯ РАБОТЫ**

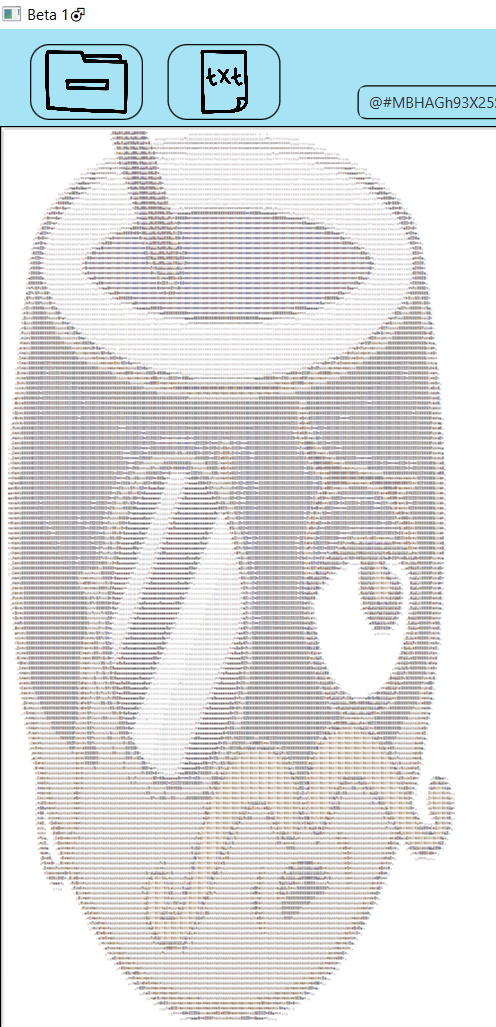
Интерфейс:



Пример преобразования: С инверсией:



Пример 2:



**ЗАКЛЮЧЕНИЕ**

В результате работы над алгоритмом было разработана программа, которая позволяет удобно и быстро преобразовать изображение в ASCII-объект.

Плюсами данного программного средства можно считать удобный и понятный интерфейс, скорость преобразования и возможность выбора списка символов. Стоит отметить архитектуру проекта, которая позволяет быстро добавлять к проекту необходимые функции.

В качестве языка разработки был выбран Java и фреймворк InteliJ IDEA, так как библиотека алгоритмов, на которой основан данный проект была реализована именно на этом языке программирования.

Разработанное программное средство представляет собой как законченный проект с полностью реализованным функционалом, так и основу для развития более совершенного программного средства, которое сможет включать в себе более обширный функционал.