

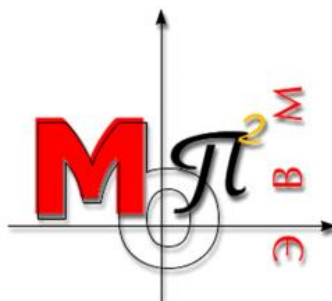
МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное образовательное учреждения высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт компьютерных технологий и информационной безопасности

Кафедра математического обеспечения и применения ЭВМ



ЛАБОРАТОРНАЯ РАБОТА № 2

по дисциплине

«Программирование компьютерной графики»

на тему:

«Форматы графических файлов»

Вариант № 1

Выполнил(и):
Студент(ы) группы
КТ604-(7/8/9)

подпись

Фамилия И. О.

подпись

Фамилия И. О.

подпись

Фамилия И. О.

Проверил:
ассистент кафедры
МОП ЭВМ

подпись

Гуляев Н. А.

Оценка

« ____ » _____ 2024 г.

1 ЦЕЛЬ, ЗАДАЧИ И ПОРЯДОК ВЫПОЛНЕНИЯ

1.1 Цель работы

Дидактическая цель работы состоит в овладении навыками разработки программных модулей, реализующих обработку растровых графических данных, получении умений проектирования специализированных форматов растровых файлов, освоении методик создания программной логики обработки растровой графики.

Практическая цель работы состоит в разработке программного модуля, выполняющего считывание графических данных из файла, преобразовании этих графических данных в формат, доступный устройству вывода, вывод изображения на устройство вывода.

1.2 Практические задачи работы

В ходе выполнения лабораторной работы перед исполнителями ставится ряд задач, включающий разработку специализированного формата растрового файла (контейнера графических данных), составление алгоритмов чтения, записи и обработки графических данных в представленном формате, проектирование и разработку программного модуля, реализующего сохранение, загрузку и вывод изображения на устройство вывода, а также подготовку документации (отчёта).

Практическим результатом выполнения настоящей лабораторной работы должен являться разработанный программный модуль, подготовленные исходные файлы, а также основная документация (отчёт по лабораторной работе).

Достижение практического результата возможно при выполнении практических задач, поставленных перед исполнителями настоящей работы.

Практические задачи соответствуют ряду рабочих задач, который включает задачи разработки компонентов программного модуля (п.3.1), состоящие из задачи разработки функционала чтения и записи графических данных некоторого (определяемого вариантом задания) формата графического файла (п.3.1.1), задачи разработки функционала интерпретации и преобразования пикселей изображения для отображения на устройстве вывода (п.3.1.2), а также задачи разработки целостного приложения, включающего разработанный функционал и осуществляющего отображение изображения и взаимодействие с пользователем (п.3.1.3).

Задачи формирования входных данных (п. 3.2) включают задачу разработки и формализованного описания структуры графического файла, а также задачу создания нескольких файлов данной структуры (п. 3.2.1).

Задачи формирования документации (п. 3.3) включают задачу компоновки разработанной документации по каждой выполненной задаче, задачу формирования документа (отчёта), содержащего текстовое описание проделанных этапов, наглядный и иллюстрационный материал (п. 3.3.1).

1.3 Порядок организации и выполнения работы

Выполнение настоящей лабораторной работы предполагает наличие двух видов порядка организации: коллективное исполнение, предполагающее

наличие бригады (коллектива) исполнителей, и индивидуальное исполнение, предполагающее индивидуальное личное исполнение.

Для коллективного исполнения бригаде (коллективу) исполнителей, включающей от одного до трёх непосредственных исполнителей, предлагаются коллективные варианты заданий, содержащие некоторое количество рабочих задач, конкретизируемых вариантом задания

Обязательное задание коллективных вариантов настоящей лабораторной работы состоит из приведенного в п. 4.1 списка рабочих задач, которые в обязательном порядке должны быть выполнены исполнителями для представления работы к защите. Рабочие задачи коллективных вариантов распределяются между участниками бригады (коллектива) исполнителей по усмотрению самих исполнителей, однако каждый исполнитель должен получить минимум одну рабочую задачу из перечня обязательных (иначе он не считается исполнителем). Одна задача может быть выполнена лишь один раз лишь одним исполнителем. Каждая рабочая задача имеет собственную «стоимость» в баллах и оценивается отдельно – соответствующие баллы выставляются исполнителю этой задачи. Однако некоторые рабочие задачи могут быть взаимосвязаны, что означает и взаимосвязь оценки выполнения данных пунктов (под-заданий). При распределении рабочих задач обязательных заданий коллективного варианта между исполнителями бригады (коллектива) таким образом, что суммарная «стоимость» в баллах всех выбранных конкретным исполнителем заданий не превышает 60% от максимального балла за работу, исполнитель имеет возможность выбрать одно или несколько дополнительных (необязательных) заданий коллективных вариантов.

Дополнительные (необязательные) задания коллективных вариантов включают несколько групп рабочих задач, перечисленных в п. 4.2. Каждый исполнитель из коллектива имеет возможность выбрать для выполнения по одному дополнительному заданию из каждой группы, получая таким образом баллы (но не выше максимального). Одна задача может быть выполнена лишь один раз лишь одним исполнителем бригады (коллектива).

Для индивидуального исполнения исполнителю предлагаются индивидуальные варианты заданий.

Обязательное задание индивидуальных вариантов предполагает выполнение задания коллективного варианта с некоторыми дополнениями по средствам исполнения, либо выполнение творческого задания.

В некоторых случаях к защите может быть представлена частично выполненная лабораторная работа, то есть работа, содержащая не все выполненные обязательные задания коллективного варианта. В таком случае необходимо учитывать, во-первых, невозможность выполнения некоторых дополнительных (необязательных) заданий, во-вторых, необходимость набора конкретным исполнителем суммарно не менее 60% от максимального балла для возможности выставления оценки по данной лабораторной работе.

Все прочие условия, включающие порядок применения в ходе выполнения настоящей работы технических аппаратных средств, программных средств, средств разработки ПО, методов разработки ПО и способов формирования входных данных изложен в разделе «Условия выполнения».

2 УКАЗАНИЯ К ВЫПОЛНЕНИЮ

2.1 Основные требования к функционалу программного модуля

Разработанный в ходе настоящей лабораторной работы программный модуль должен являться целостным приложением, выполняющим чтение, обработку и запись графических данных некоторого формата файла (контейнера), а также отображение загруженного растрового изображения в визуально воспринимаемой форме на устройстве вывода.

Программный модуль должен содержать программную реализацию логики обхода (чтения и записи) некоторого растрового графического файла (контейнера). Данный файл должен быть некоторым образом передан программному модулю, так чтобы программный модуль мог получить доступ к содержимому файла, что может быть реализовано средствами платформы, среды разработки или средствами ОС.

Программный модуль должен содержать программную реализацию логики преобразования пикселей исходного формата (полученных из файла) в формат устройства вывода, что необходимо для выполнения отображения пикселей изображения на устройстве вывода. Для доступа к устройству вывода могут использоваться средства платформы, среды разработки или средств ОС.

Файл, хранящий графические данные, должен иметь чётко заданную структуру, определяемую вариантом задания, что подразумевает полную совместимость программного модуля со всеми файлами данного формата. Файлы могут быть разработаны «вручную» – при помощи специализированных шестнадцатеричных редакторов, либо при помощи разработанного программного модуля.

Приложение в обязательном порядке должно иметь «основное» окно (элемент графического интерфейса), в котором происходит отображение изображения, загруженного из растрового файла. Дополнительно приложение может содержать прочие элементы графического пользовательского интерфейса, несущие соответствующую смысловую нагрузку в рамках конкретной задачи. Непосредственное отображение графического содержимого (пикселей изображения) может быть выполнено при помощи «стороннего» функционала – функционала среды разработки, операционной системы, либо программно-аппаратными средствами платформы. Например, для отображения графического пользовательского интерфейса приложения рекомендуется использовать стандартный графический интерфейс ОС Windows, задействуемый при помощи WinAPI (напрямую или средствами платформы). Для выполнения отрисовки растровой графики также может быть использовано стандартное средство рисования ОС Windows – GDI/GDI+, задействуемое при помощи WinAPI (напрямую или средствами платформы).

Отрисовка пикселей изображения может быть выполнена при помощи произвольных средств – средств среды разработки, операционной системы, либо программно-аппаратными средствами платформы, однако рекомендуется использовать растровые библиотеки, которые позволяют получить доступ к единичным пикселям изображения, выводимого на устройство вывода.

2.2 Средства выполнения работы

При выполнении данной лабораторной работы допускается использование любых IBM-PC-совместимых аппаратных платформ архитектуры x86 (x64). При разработке алгоритмов допускается использование любых (сторонних) средств наглядного представления алгоритма и текстовых редакторов для написания псевдо-кода или алгоритмического языка. Для написания программного кода и ведения непосредственной разработки программного модуля необходимо использовать средства разработки программного обеспечения, совместимых с целевой платформой, а также имеющих возможность реализовать приведенные в разделе 3 задачи (конкретизируемые вариантом задания).

Для формирования тестовых данных (изображений) могут использоваться программные средства растрового редактирования изображений известных форматов («.bmp», «.jpg», «.png» и т.д.):

Microsoft Paint – стандартное программное средство просмотра и редактирования растровых изображений ОС Windows нескольких известных форматов (присутствует во всех версиях ОС);

Paint.NET – бесплатно распространяемое продвинутое программное средство просмотра и редактирования растровых изображений многих известных форматов, доступно по URL: <https://www.getpaint.net>;

Gimp – открытое программное средство просмотра и редактирования растровых изображений ОС Windows, доступно по URL: <https://www.gimp.org>;

Для формирования тестовых данных (изображений) могут использоваться программные редактирование двоичных или шестнадцатеричных данных:

WinHex – условно-бесплатно распространяемое программное средство просмотра и редактирования двоичных и шестнадцатеричных данных в файлах, доступно по URL: <https://winhex.ru>;

XVI32 – бесплатно распространяемое программное средство просмотра и редактирования двоичных и шестнадцатеричных данных в файлах, доступно по URL: <http://www.chmaas.handshake.de/delphi/freeware/xvi32/xvi32.htm>;

Hxd – бесплатно распространяемое программное средство просмотра и редактирования двоичных и шестнадцатеричных данных в файлах, доступно по URL: <https://mh-nexus.de/en/hxd/>.

2.3 Структура файла

Условиями настоящей лабораторной работы определяется общая структура файла (контейнера). Файл (контейнер) всегда имеет фиксированную структуру вне зависимости от формата, разрядности и прочих характеристик пикселя, определяемых вариантом задания. Обязательной частью файла является заголовок – блок данных, содержащий информацию о свойствах изображения (рисунок 1), при этом такие данные не должны быть интерпретированы как графические данные (как часть непосредственного изображения).



Рисунок 1 – Структура файла

Заголовок представляет собой блок фиксированного размера – 7 байт, который содержит 4 целочисленных значения, расположенных последовательно:

1. Ширина изображения в пикселях (word, 2 байта);
2. Высота изображения в пикселях (word, 2 байта);
3. Количество бит, задействованных под один пиксель (byte, 1 байт);
4. Количество значений, находящихся в палитре этого файла (word, 2 байта).

В зависимости от конкретизации варианта задания некоторые поля заголовка могут нести иной смысл, либо не использоваться (однако место под данные поля обязательно выделяется).

Далее в файле должен быть расположен ещё один блок данных нефиксированного размера, хранящий значения палитры. Значение палитры представляет собой независимые от устройства цветовые значения, представленные в формате ARGB. Каждое значение палитры имеет формат ARGB, соответственно размер равен 4 байтам, первый байт хранит данные о непрозрачности, второй – красный канал, третий – зелёный канал, четвёртый – синий канал цвета.

Далее в файле располагаются графические данные, формирующие пиксели изображения, структура и содержимое которых определяется структурой пикселя согласно варианту задания.

2.4 Структура пикселя

Условиями некоторых вариантов настоящей лабораторной работы определяется структура пикселя. Например, пиксель изображения может представлять собой n -битное значение, хранящее некоторое значение, определяющее либо непосредственное значение цвета, либо способ получения этого цветового значения для его вывода на устройство вывода. В зависимости от количества бит на пиксель, один байт может содержать различное количество пикселей (рисунок 2).

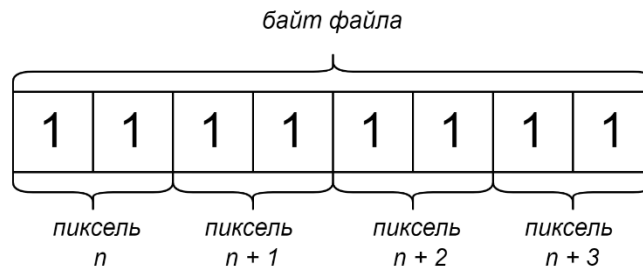


Рисунок 2 – Пример представления два бита на пиксель

Аналогичным образом, при использовании количества бит на пиксель, не кратного одному, двум, четырём и восьми, значение пикселя может быть разделено между двумя последовательно идущими байтами.

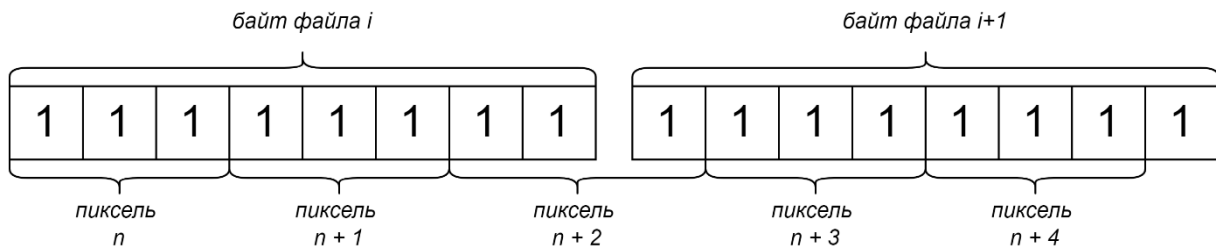


Рисунок 3 – Пример представления два бита на пиксель

Порядок битов, кодирующих пиксель внутри байта, может определяться произвольным образом, например, по принципу: большая координата соответствует большему разряду бита (бит) в байте. Пусть существует изображение размером 4×2 пикселя, содержащее пиксели A, B, C, D, E, F, G и H, кодируемые по 2 бита каждый (рисунок 4).

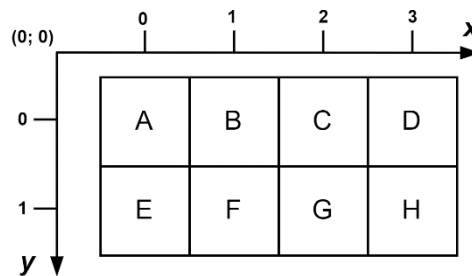


Рисунок 4 – Пример изображения

Поместив пиксели данного изображения построчно в последовательно расположенные байты в файле в указанном порядке (большой разряд содержит большую координату), может быть получено следующее упакованное представление (рисунок 5).

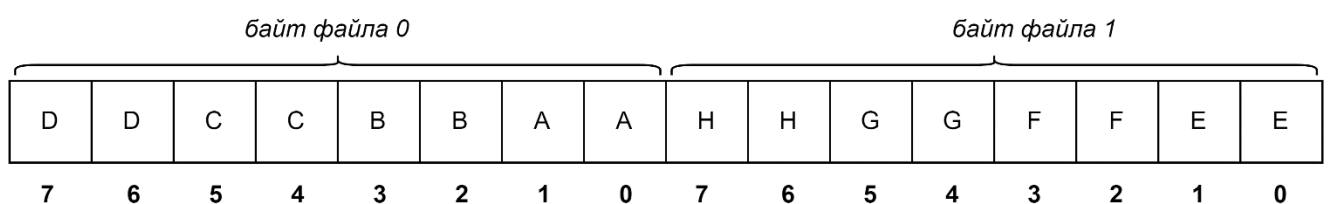


Рисунок 5 – Упакованное представление

Так, пиксель с А координатами (0; 0) ожидаемо будет расположен в нулевом байте в нулевом и первом бите. Пиксель D с координатами (3; 0) будет расположен в нулевом байте в шестом и седьмом бите, а пиксель E с координатами (0; 1) будет расположен в нулевом и первом битах второго байта файла. Аналогично может быть использована формула свёртки двухмерного массива в одномерный и обратная формула – развёртки одномерного массива в двухмерный с заданной шириной. Также важно иметь в виду другой значимый нюанс – порядок битов внутри самого пикселя, в зависимости от которого значение пикселя может быть закодировано от старшего бита к младшему или наоборот.

2.5 Подготовка исходных данных

В качестве исходных данных для выполнения настоящей лабораторной работы необходимо принять созданные при помощи средств двоичного или шестнадцатеричного редактирования (WinHex, XVI32 и т.д.), либо при помощи фрагмента разработанной программной логики нескольких изображений произвольного (желательно небольшого) разрешения, соответствующих указанному (вариантом задания) формату файла и содержащих соответствующие варианту задания данные. Также может быть разработан вспомогательный программный модуль, генерирующий некоторое изображение и сохраняющий файл в соответствующем формате.

3 ПРАКТИЧЕСКИЕ ЗАДАЧИ

3.1 Задачи разработки программного модуля

3.1.1 Программная реализация логики чтения и записи файла (контейнера)

Программный модуль должен содержать программную реализацию логики обхода (чтения и записи) растрового графического файла (контейнера), определённой вариантом задания структуры. Программная реализация должна быть основана на разработанных алгоритмах обхода графического файла этой структуры. Техническое воплощение может представлять собой фрагмент (фрагменты) программной логики, реализованной посредством конструкций языка и среды разработки (классы, модули, функции).

Функционал этого фрагмента программной логики должен предполагать проход содержимого файла (контейнера) до искомого блока данных, произведение считывания или записи этого блока.

Входными данными этого фрагмента программной логики (фрагмента программного модуля) должны быть файлы, хранящиеся в файловой системе ОС, имеющие определённую внутреннюю структуру (согласно варианту задания). Файлы содержат элементы структуры, блоки данных, которые несут полезное содержимое – данные, описывающие характеристики хранимого изображения (ширина, высота, разрядность и другие характеристики), непосредственные пиксели изображения (в определяемом варианте задания представлении), а также другое содержимое (определяется дополнительным и индивидуальным заданиями).

Выходными данными этого фрагмента программной логики (фрагмента программного модуля) должны быть искомые блоки данных – данные, описывающие характеристики хранимого изображения (ширина, высота, разрядность и другие характеристики), пиксели (в некотором представлении), а также другое содержимое (согласно дополнительным и индивидуальным заданиям).

Разработанные алгоритмы должны быть представлены в наглядном виде при помощи блок-схем согласно ГОСТ 19.701-90, либо при помощи псевдо-кода.

Разработанный программный код (фрагмент) должен быть приведён в документации (разделе «Приложение» отчёта);

3.1.2 Программная реализация логики преобразования пикселей изображения

Программный модуль должен содержать программную реализацию логики получения блока данных, содержащих пиксель (пиксели) изображения в определённом варианте задания формате, и преобразования в формат пикселей устройства вывода – реализацию прямого и обратного преобразования пикселей между форматом пикселя устройства вывода и форматом пикселя, определяемого вариантом задания. Программная реализация должна быть основана на разработанных алгоритмах на базе соответствующего математического аппарата (алгебра, булева алгебра). Вычисления (преобразования) могут содержать любые численные и булевы вычислительные действия, поддерживаемые средой и языком разработки. Техническое

воплощение может представлять собой фрагмент (фрагменты) программной логики и конструкций языка и среды разработки (классы, модули, функции).

Функционал этого фрагмента программной логики должен предполагать преобразование пикселя (пикселей), содержащихся в полученном (считанном) из файла (контейнера) блоке данных, из определённого вариантом задания формата представления пикселей в формат пикселей устройства вывода.

Входными данными являются пиксели, содержащиеся в считанном из файла (контейнера) блоке (блоках) данных, хранящиеся в некотором (определённом варианте задания) представлении пикселей.

Выходными данными этого фрагмента программной логики (фрагмента программного модуля) должны быть пиксели (пиксель) формата устройства вывода, которые могут быть отображены на устройстве вывода.

Разработанные алгоритмы должны быть представлены в наглядном виде при помощи блок-схем согласно ГОСТ 19.701-90, либо при помощи псевдо-кода.

Разработанный программный код (фрагмент) должен быть приведён в документации (разделе «Приложение» отчёта).

3.1.3 Разработка приложения с пользовательским интерфейсом

Необходимо произвести разработку приложения, включающего разработанный в п.3.1.1 и п.3.1.2 функционал, а также дополнительный функционал, заключающийся в отображении загружаемых из файлов изображений и обеспечении пользователю возможности взаимодействия с приложением.

Функционал этого фрагмента программной логики должен предполагать получение и отображение на устройстве вывода пикселей, считанных из файла (п. 3.1.1) и преобразованных в формат устройства вывода (п. 3.1.2) с учётом указанных пользователем параметров отображения.

Отображение изображения в рамках этого фрагмента программной логики должно быть произведено только при помощи специализированного элемента графического интерфейса (оконной формы, либо другого элемента интерфейса). Пиксели изображения должны быть отрисованы напрямую, например, при помощи заполнения раstra соответствующего элемента интерфейса, либо при помощи отрисовки «замещающих» элементов, например, закрашенных прямоугольников.

Параметры отображения должны предполагать следующие изменения пользователем характеристик отображаемого изображения:

- Масштаб изображения – пользователь должен иметь возможность указать масштаб изображения в коэффициенте или в процентах (рекомендуется 100-500%);
- Смещение изображения – пользователь должен иметь возможность указать смещение изображения по вертикали и горизонтали;
- Попиксельное отображение – пользователь должен иметь возможность включать и отключать отображение сетки раstra (при достаточном уровне масштаба), в котором расположены пиксели.

Входными данными являются пиксели в формате устройства вывода, а также параметры отображения, которые пользователь может передать в

программный модуль (например, при помощи элементов пользовательского интерфейса).

Выходными данными этого фрагмента программной логики (фрагмента программного модуля) должны быть отображаемые на устройстве вывода изображения.

Руководство пользователя, поясняющее порядок работы пользователя с программным модулем, должно быть приведено в виде текста, при необходимости дополненного иллюстрациями (интерфейса приложения).

Разработанный программный код (фрагмент) должен быть приведён в документации (разделе «Приложение» отчёта).

3.2 Задачи формирования обрабатываемых данных

3.2.1 Создание специализированного формата графического файла

Программный модуль должен выполнять чтение и запись данных в некоторый строго определённый формат графического файла (контейнера), для чего необходимо произвести создание соответствующего формата файла. Необходимо выполнить формализованное описание структуры графического файла – всех «элементов» этого файла – «блоков» данных, согласно варианту задания. Далее необходимо выполнить создание нескольких экземпляров файлов, содержащих различные изображения с различными параметрами (ширина, высота, палитра и т.д.), варьируемые согласно варианту задания.

Формальное описание структуры файла должно быть разработано и реализовано в наглядном представлении – в виде графической схемы структуры файла (диаграмма, отображающая структурные элементы файла) с необходимыми пояснениями (текст, содержащий комментарии к элементам схемы).

Создание файлов описанной структуры может быть выполнено «ручным» способом – при помощи шестнадцатеричного (двоичного) редактора, либо при помощи некоторого отдельного фрагмента программной логики. Созданные файлы должны быть обязательно сохранены в файловую систему для возможности дальнейшего открытия и использования программным модулем. Для демонстрации достаточно двух-трёх файлов, содержащих различные изображения, различные наборы палитр (при наличии), различные разрешения и т.д.

3.3 Задачи формирования документации

3.3.1 Создание (компонование) отчёта

Формируемая документация (отчёт) должна включать:

Описание дидактической и практической цели работы – пункт, содержащий тезисное описание целей, которые необходимо достичь в ходе выполнения данной работы;

Постановку задания и вариант основного задания – пункт, содержащий краткое описание основного (обязательного) задания, перечня рабочих задач основного (обязательного) задания с распределением по исполнителям, а также перечень задач дополнительного задания с распределением по исполнителям. Для каждого исполнителя приводится список задач согласно форме Ф2;

Описание и наглядное представление структуры растрового файла – пункт, содержащий описание и наглядное представление (в виде схемы) структуры растрового файла, определяемой вариантом задания. Если выполнены дополнительные задания, предписывающие внесение изменений в структуру файла, блоков или пикселя, то такие изменения также приводятся в этом пункте (с указанием исполнителя);

Описание методов преобразования пикселей – пункт, включающий описание и наглядное представление (в виде схемы) структуры пикселя, а также математизированное описание преобразований пикселей между форматом, определяемым вариантом задания, и форматом устройства вывода. Формальное описание необходимо приводить в виде формул, а также выражений булевой или векторной алгебры. Если выполнены дополнительные задания, предписывающие дополнительные преобразования пикселей, то их описание также приводятся в этом пункте (с указанием исполнителя);

Описание разработанных алгоритмов – пункт, включающий представленное в наглядном виде (в виде блок-схем или псевдо-кода) описание всех разработанных алгоритмов:

- Алгоритм чтения и записи файла (контейнера), разработанный в п.3.1.1;
- Алгоритм преобразование пикселей, разработанный в п.3.1.2;
- Алгоритмы, разработанные в п.3.1.1, п.3.1.2;
- Алгоритмы, разработанные при выполнении дополнительных заданий;

Описание используемых средств – пункт, содержащий описание средств разработки, средств вывода графики, задействованных технологий в виде текста, терминов, при необходимости – изображений и диаграмм. Необходимо привести описание основных используемых средств разработки, целевой платформы, встроенных средств платформы. При использовании в программном модуле функционала сторонних библиотек – необходимо приводить краткое описание данных библиотек и используемого функционала – конкретных библиотечных методов, при этом необходимо указывать сигнатуры методов (функций) с кратким пояснением. Если выполнены дополнительные задания, предписывающие использование других (дополнительных) технологий, то их описание также приводится в данном пункте;

Описание функционала разработанного программного модуля – пункт, содержащий описание функционала разработанного программного модуля – структурные элементы программного модуля – сигнатуры функций, типы данных, классы (при использовании ООП) с кратким пояснением роли и назначения (без листинга кода, листинг приводится в разделе «приложение»);

Описание полученных результатов работы – пункт, содержащий несколько примеров исходного изображения и содержимого (в двоичном виде) файла, в котором находится это изображение;

Приложение (листинг) – раздел, содержащий листинг исходного кода;

Приложение (графический материал) – раздел, содержащий полученные результаты (изображения, таблицы, графики и т.д.);

4 ВАРИАНТЫ ЗАДАНИЯ

4.1 Обязательные рабочие задачи коллективных вариантов задания

4.1.1 Рабочая задача КВ.О.1. Разработка программной логики чтения и записи графического файла

Постановка данной задачи соответствует постановке практической задачи п.3.1.1 в рамках разработки программного модуля. Содержание настоящей рабочей задачи включает разработку алгоритмов обхода (для чтения и записи) содержимого растрового графического файла (контейнера) формата, разработанного в п.3.2.1, а также выполнение программной реализации соответствующего фрагмента (элемента) программного модуля.

В ходе выполнения необходимо:

- Разработать и выполнить формализованное описание при помощи блок-схем или псевдо-кода алгоритма (алгоритмов) обхода содержимого графического файла разработанного в п.3.2.1 формата для чтения и записи данных;
- Произвести программную реализацию разработанного алгоритма обхода содержимого графического файла разработанного в п.3.2.1 формата для чтения и записи данных;
- Произвести отладку и проверку полученного фрагмента программного модуля на тестовых данных (файлах), созданных в п.3.2.1;

Результат должен содержать:

- Блок-схему согласно ГОСТ 19.701-90 или псевдо-код, содержащий формализованное описание алгоритма, при необходимости – словесные пояснения в виде комментариев;
- Фрагмент программного модуля (программного кода), содержащего реализацию разработанного алгоритма;

Оценивается:

- Соответствие разработанного алгоритма поставленной задаче – алгоритм реализует корректный обход содержимого файла, программная реализация не содержит ошибок, что подтверждается проверкой на тестовых данных (файлах, созданных в п.3.2.1) – **1 балл**;
- Возможность пояснения исполнителем элементов алгоритма, программной реализации, аспектов функционирования соответствующего фрагмента программного модуля, возможность корректировки и внесения изменений в алгоритм или программный модуль – **1 балл**;

Балл за выполнение рабочей задачи КВ.О.1: 2 балла.

4.1.2 Рабочая задача КВ.О.2. Разработка программной логики преобразования пикселей изображения

Постановка данной задачи соответствует постановке практической задачи п.3.1.2 в рамках разработки программного модуля. Содержание настоящей рабочей задачи включает разработку алгоритмов прямого и обратного преобразования пикселей, организованных особым (согласно конкретному варианту задания) способом, полученных из графического файла формата, разработанного в п.3.2.1 при помощи программной логики обхода и считывания, разработанной в п.3.1.1, в формат пикселей устройства вывода, а также

выполнение программной реализации соответствующего фрагмента (элемента) программного модуля.

В ходе выполнения необходимо:

- Выполнить разработку и формализованное описание при помощи блок-схем или псевдо-кода алгоритма преобразования пикселей изображения (формата, содержащегося в файле) в формат устройства вывода;

- Выполнить программную реализацию разработанного алгоритма преобразования пикселей изображения исходного формата (содержащегося в файле) в формат устройства вывода;

- Произвести отладку и проверку полученного фрагмента программного модуля на тестовых данных (файлах), созданных в п.3.2.1;

Результат должен содержать:

- Блок-схему согласно ГОСТ 19.701-90 или псевдо-код, содержащий формализованное описание алгоритма, при необходимости – словесные пояснения в виде комментариев;

- Фрагмент программного модуля (программного кода), содержащего реализацию разработанного алгоритма;

Оценивается:

- Соответствие разработанного алгоритма и программной реализации поставленной задаче – алгоритм реализует корректное преобразование цветовых значений и позиций пикселей, программная реализация не содержит ошибок, что подтверждается проверкой на тестовых данных (файлах), созданных в п.3.2.1 – **1 балл**;

- Возможность пояснения элементов алгоритма, включая математический аппарат, элементов программной реализации, аспектов функционирования соответствующего фрагмента программного модуля, возможность корректировки и внесения изменений в алгоритм или программный модуль – **1 балл**;

Балл за выполнение рабочей задачи КВ.О.2: 2 балла.

4.1.3 Рабочая задача КВ.О.3. Разработка приложения и пользовательского интерфейса

Постановка данной задачи соответствует постановке практической задачи п.3.1.3 в рамках разработки программного модуля. Содержание настоящей рабочей задачи включает разработку приложения, реализующего доступ ко всему функционалу (фрагментам программного модуля), разработанному в п.3.1.2 и п.3.1.3, включая реализацию сохранения, загрузки и вывода изображения на устройство вывода, а также отображения изображения пользователю с указанными параметрами отображения.

В ходе выполнения необходимо:

- Выполнить компоновку разработанного в п.3.1.2 и п.3.1.3 функционала при помощи средств языка программирования (в виде библиотек, модулей или подпрограмм), предоставив программный интерфейс;

- Выполнить разработку графического интерфейса приложения – экранных форм (окон), предоставляющих пользователю возможность управления работой приложения, изменения параметров отображения, а визуального отображения, загруженного из файла;

- Выполнить отладку разработанной программной логики и проверку корректности работы пользовательского интерфейса;

Результат должен содержать:

- Описание порядка работы пользователя с приложением (руководство пользователя);

- Фрагмент (фрагменты) программного кода, содержащего реализацию разработанной программной логики;

Оценивается:

- Соответствие функционала разработанного приложения поставленной задаче – приложение производит загрузку из файла (созданного в п.3.2.1) и выполняет корректное отображение изображения с указанными пользователем настройками (параметрами отображения) – **1 балл**;

- Возможность пояснения элементов программной реализации, программных интерфейсов, функционирования пользовательского интерфейса – **1 балл**;

Балл за выполнение рабочей задачи КВ.О.3: 2 балла.

4.1.4 Рабочая задача КВ.О.4. Разработка формата графического файла

Постановка данной задачи соответствует постановке практической задачи п.3.2.1 в рамках формирования обрабатываемых данных. Содержание настоящей рабочей задачи включает разработку особого (согласно конкретному варианту задания) формата графического файла, определяющего способ хранения содержащихся графических данных (пиксели определённого формата), формирующих некоторое изображение, выводимое программным модулем на устройство вывода.

В ходе выполнения необходимо:

- Выполнить формализованное описание структуры графического файла – всех «элементов» файла и «блоков» данных в наглядном представлении (графическая схема структуры файла) с необходимыми пояснениями (текст, содержащий комментарии к элементам схемы);

- Выполнить создание файлов описанной структуры, содержащих некоторые изображения (необходимые далее для отображения при помощи разработанного программного модуля) вручную при помощи шестнадцатеричного (двоичного) редактора (см. средства выполнения работы), либо автоматизированным способом (при помощи специализированного программного модуля, фрагмента программного кода и т.д.);

Результат должен содержать:

- Наличие графической схемы (диаграммы), иллюстрирующей структуру графического файла согласно варианту задания, а также текстовое описание с пояснениями к графической схеме;

- Наличие созданных файлов описанной структуры, содержащих изображения, которые могут быть отображены при помощи разработанного программного модуля;

Оценивается:

- Соответствие разработанной структуры файла требованиям условий и варианта задания, корректность и полнота наглядного описания структуры файла при помощи схемы – **1 балл**;

- Факт наличия и соответствия созданных файлов, содержащих изображения, описанной структуре, возможность отображения при помощи программного модуля, возможность разбора и пояснения содержимого файлов – **1 балл**;

Балл за выполнение рабочей задачи КВ.О.4: 2 балла.

4.1.5 Рабочая задача КВ.О.5. Формирование (отчёта) документации

Постановка данной задачи соответствует постановке практической задачи п.3.3.1 в рамках формирования документации. Содержание настоящей рабочей задачи включает разработку отчёта о выполненной лабораторной работе.

В ходе выполнения необходимо:

- Выполнить компоновку в единый документ всех фрагментов документации по всем выполненным в рамках настоящей работы задачам, распределить материал по главам (разделам) и пунктам (подразделам), распределить материал в приложениях;

- Произвести оформление документа согласно ГОСТ 7.32-2001;

Результат должен содержать:

- Подготовленный и оформленный отчёт по лабораторной работе;

Оценивается:

- Полнота изложения задания, требований и условий выполнения работы, качество отражения результатов выполненных рабочих задач в отчёте, наличие необходимых элементов наглядного представления – блок-схем или псевдокода, а также графического материала – схем, диаграмм, графиков, скрин-шотов – **1 балл**;

Балл за выполнение рабочей задачи КВ.О.5: 1 балл.

4.2 Содержание задания коллективных вариантов

4.2.1 Задание коллективного варианта №1

Формат растрового файла: файл содержит пиксели растрового изображения, цвет пикселей содержится в дискретной палитре;

Формат пикселя: пиксель содержит «координаты» строки и столбца палитры, в которой хранится цвет этого пикселя, при отрисовке пикселя используется цвет из соответствующих «координат» палитры. Значения строки и столбца в двоичном представлении пикселя кодируются минимально необходимым количеством битов, биты расположены в байтах файла упакованным образом (в одном байте могут находиться биты текущего и следующего пикселя);

Формат палитры: палитра представляет собой матрицу размером 5 × 3 элементов. Столбцы палитры содержат некоторые цвета, а строки – оттенки этих цветов. Каждый пиксель ссылается на один из элементов палитры, расположенном в соответствующих «координатах» (строки и столбца);

Формат цвета палитры: цвет элементов палитры кодируется при помощи формата ARGB (32-битный цвет, по 8 бит на компоненту цвета);

4.2.2 Задание коллективного варианта №2

Формат растрового файла: файл содержит пиксели растрового изображения, цвет пикселей содержится в непрерывной палитре – квадрате, в вершинах которого расположены некоторые цвета;

Формат пикселя: пиксель содержит декартовы координаты x и y точки внутри квадрата, определяющего «палитру», при этом $x \in [0; 1]$, $y \in [0; 1]$. Точка $P(x; y)$, расположенная на квадрате, содержащем цвета, определяет цвет пикселя как результат билинейной интерполяции (между вершинами квадрата). Значения координаты x и координаты y представлены двухбайтовыми вещественными числами, которые расположены последовательно (сначала x , потом y);

Формат палитры: палитра всегда содержит четыре цветовых значения, каждое из которых расположено в четырёхугольнике с вершинами $A[0; 0]$, $B[0; 1]$, $C[1; 1]$, $D[1; 0]$ (каждая вершина соответствует некоторому цвету). Любая точка внутри этого прямоугольника соответствует «промежуточному» цвету, получаемому при помощи «смешивания» цветов, находящихся в вершинах при помощи билинейной интерполяции;

Формат цвета палитры: цвет элементов палитры кодируется при помощи формата ARGB (32-битный цвет, по 8 бит на компоненту цвета);

4.2.3 Задание коллективного варианта №3

Формат растрового файла: файл содержит пиксели растрового изображения, цвет пикселей содержится непосредственно в самих пикселях в канальном представлении, при этом существует несколько «видов» пикселей, каждый «вид» пикселя предполагает различное количество бит для кодирования каналов цвета;

Формат пикселя: пиксель содержит флаг, указывающий «вид» этого пикселя, а также цветовые каналы, содержащие значение компонент цвета (красный, синий, зелёный). Существует четыре вида пикселей: «полный», «эквивалентный», «красные тона», «сине-зелёные тона»;

- Тип пикселя «полный»: первый бит отводится под хранение значения флага, три следующих бита – под красный канал, под зелёный и синий каналы – по два соответственно;

- Тип пикселя «эквивалентный»: первый бит отводится под хранение значения флага, по два бита – под красный канал, зелёный и синий каналы;

- Тип пикселя «красные тона»: первый бит отводится под хранение значения флага, три следующих бита – под красный, по одному биту – под зелёный и синий каналы соответственно;

- Тип пикселя «сине-зелёные тона»: первый бит отводится под хранение значения флага, один следующий бит – под красный, по два бита – под зелёный и синий каналы соответственно;

В двоичном представлении биты расположены в байтах файла упакованным образом (в одном байте могут находиться биты текущего и следующего пикселя);

Формат палитры: палитра не используется, размер блока палитры равен нулю;

4.2.4 Задание коллективного варианта №4

Формат растрового файла: файл содержит пиксели растрового изображения, цвет пикселей содержится в дискретной палитре – в секторах круга, где каждому сектору присвоен некоторый цвет (между секторами цвет не интерполируется);

Формат пикселя: пиксель содержит декартовы координаты x и y точки внутри круга, определяющего «палитру», при этом $x^2 + y^2 \leq R$. Точка $P(x; y)$, расположенная в круге, обязательно попадает в какой-либо сектор этого круга, содержащий какой-либо цвет. Цвет пикселя определяется цветом сектора, в котором лежит точка, хранящаяся в данном пикселе. Значения координаты x и координаты y представлены двухбайтовыми вещественными числами, которые расположены последовательно (сначала x , потом y);

Формат палитры: палитра может содержать произвольное количество цветовых значений (что отражается в заголовке), при этом каждый элемент палитры должен содержать дополнительную информацию (угол, длина дуги) о секторе, к которому относится (рекомендуется располагать секторы по порядку);

4.3 Дополнительные рабочие задачи коллективных вариантов

4.3.1 Рабочие задачи группы «А» – разработка программной логики модификации графических данных в файле

Рабочая задача КВ.Д.А1: разработать фрагмент программной логики, реализующей модификацию находящихся в файле пикселей изображения таким образом, что производится изменение уровня контрастности (по одной из формул для контрастности) всего изображения. Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего изображение сохраняется в новый файл. Произвести документирование (указать использованные формулы, этапы вычисления), произвести устное пояснение произведённых вычислений.

Балл за выполнение рабочей задачи КВ.Д.А1: 1 балл.

Балл за выполнение рабочей задачи КВ.Д.А1 для варианта №2: 2 балла.

Рабочая задача КВ.Д.А2: разработать фрагмент программной логики, реализующей модификацию находящихся в файле пикселей изображения таким образом, что производится замена оттенков (некоторый интервал значений, либо заданный перечень цветов) цвета пикселей на некоторый другой (один) цвет, установленный в программе или указанный пользователем. Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего изображение сохраняется в новый файл. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.А2: 1 балл.

Рабочая задача КВ.Д.А3: разработать фрагмент программной логики, реализующей модификацию находящихся в файле пикселей изображения таким образом, что производится копирование некоторой части одного изображения, хранящегося в одном файле, в другое, хранящееся в другом файле. Исходный размер копируемого фрагмента может быть указан в программе, либо задан пользователем. Оба файла имеют формат, в котором производится хранение

согласно варианту задания. Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего изображение сохраняется в новый файл. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.А3: 1 балл.

Рабочая задача КВ.Д.А4: разработать фрагмент программной логики, реализующей модификацию находящихся в файле пикселей изображения таким образом, что производится кратное масштабирование изображения (2, 4, 6 раз). Изменение пикселей (при необходимости – заголовка файла) производится в исходном формате (в формате, в котором производится хранение в файле), после чего изображение сохраняется в новый файл. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.А4: 2 балла.

Балл за выполнение рабочей задачи КВ.Д.А4 для варианта №2: 1 балл.

Рабочая задача КВ.Д.А5: разработать фрагмент программной логики, реализующей модификацию находящихся в файле пикселей изображения таким образом, что ко всем пикселям изображения производится применение медианного фильтра апертурой размером 5×5 . Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего изображение сохраняется в новый файл. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.А5: 2 балла.

Рабочая задача КВ.Д.А6: реализовать функционал «наложения» двух загруженных из файлов (формата, в котором производится хранение согласно варианту задания) изображений по одному из методов, используемых в растровых редакторах: «Add», «Multiply», «OR», «XOR», «LinearBurn», «ColorBurn», «LinearDodge» или «ColorDodge». Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего изображение сохраняется в новый файл. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.А6: 2 балла.

Балл за выполнение рабочей задачи КВ.Д.А6 для варианта №2: 3 балла.

4.3.2 Рабочие задачи группы «Б» – разработка программной логики создания и дополнения изображения

Рабочая задача КВ.Д.Б1: разработать фрагмент программной логики, реализующей изменение цветового значения пикселя загруженного из файла изображения в указанных пользователем программного модуля координатах. Реализовать элементы графического пользовательского интерфейса (окна, диалоги, элементы управления), позволяющие пользователю выполнить выбор и установку значения пикселя. Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего

существующий файл перезаписывается. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Б1: 1 балл.

Рабочая задача КВ.Д.Б2: разработать фрагмент программной логики, реализующей возможность «рисования» на загруженном из файла изображении заданным цветом при помощи манипулятора «мышь». Реализовать элементы графического пользовательского интерфейса (окна, диалоги, элементы управления), позволяющие пользователю выполнить выбор цветового значения («цвета» кисти). Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего существующий файл перезаписывается. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Б2: 1 балл.

Рабочая задача КВ.Д.Б3: разработать фрагмент программной логики, реализующей возможность «заливки» на загруженном из файла изображении заданным цветом некоторой области, пиксели которой имеют одинаковый цвет. Допускается использовать любой произвольный алгоритм заливки. Реализовать элементы графического пользовательского интерфейса (окна, диалоги, элементы управления), позволяющие пользователю выполнить выбор цветового значения («цвета» закрашки) и затравочной точки. Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего существующий файл перезаписывается. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Б3: 1 балл.

Рабочая задача КВ.Д.Б4: разработать фрагмент программной логики, реализующей возможность отрисовки на загруженном из файла изображении заданным цветом заливки и заданным цветом контура в заданных координатах с заданным размером некоторой геометрической фигуры (прямоугольник, треугольник, круг и т.д.). Реализовать элементы графического пользовательского интерфейса (окна, диалоги, элементы управления), позволяющие пользователю выполнить выбор параметров рисуемой фигуры. Изменение пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего существующий файл перезаписывается. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Б4: 1 балл.

Рабочая задача КВ.Д.Б5: разработать специализированный формат файла, хранящий информацию о фигурах, отрисовываемых в доп. задании КВ.Д.Б3 (положение, размер, цвет). Файл данного формата должен хранить параметры каждой фигуры, при загрузке данного файла программный модуль должен выполнять отрисовку соответствующих фигур в открытом файле. Реализовать программную логику чтения данного файла и формирования некоторого изображения в графическом файле на основе описанных в этом

файле фигур. Выполнить формирование нескольких таких файлов, продемонстрировать при работе программного модуля. Произвести документирование (указать использованные средства и способы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Б5: 2 балла.

Рабочая задача КВ.Д.Б6: разработать фрагмент программной логики, реализующий генерацию файлов (изображений в формате, определяемым вариантом заданий), содержащих некоторый узор, например, «плитки», «круги», «градиент» и т.д. Выполнить формирование нескольких таких файлов, продемонстрировать при работе программного модуля. Произвести документирование (указать использованные средства и способы формирования фрактальных изображений), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Б6: 2 балла

Рабочая задача КВ.Д.Б7: разработать фрагмент программной логики, реализующий генерацию файлов (изображений в формате, определяемым вариантом заданий), содержащих изображения геометрических фракталов. Использовать один из фракталов: «Треугольник Серпинского», «Ковёр Серпинского», «Остров Коха». Геометрические фракталы должны быть цветными. Выполнить формирование нескольких таких файлов, продемонстрировать при работе программного модуля. Произвести документирование (указать использованные средства и способы формирования фрактальных изображений), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Б7: 2 балла

4.3.3 Рабочие задачи группы «В» – преобразование и дополнение формата графического файла

Рабочая задача КВ.Д.В1: разработать и реализовать ASCII-представление (текстовое представление) исходного формата графического файла (формата, определяемого текущим вариантом коллективного задания). Реализовать некоторую систему тэгов (разработать оригинальную или использовать существующую – XML или JSON), обеспечивающих соответствие типа «ключ-значение». Реализовать функционал «преобразования» графического файла (заголовков, палитры, пикселей) исходного формата в файл разработанного формата. Обработка пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего формируется новый файл разработанного формата, в котором производится запись необходимых элементов (тэгов) и непосредственных пикселей изображения, преобразованных к корректному формату целевого файла.

Для преобразований не допускается использование «стандартных» средств среды разработки, платформы или ОС (например, сериализации) – необходимо выполнить разработку алгоритма записи файла нового формата самостоятельно.

Произвести документирование (привести структуру нового формата файла, кратко пояснить основные элементы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.В1: 2 балла.

Рабочая задача КВ.Д.В2: дополнить исходный формат графического файла (формат, определяемый текущим вариантом коллективного задания) дополнительными данными, определяющими физическую форму (габариты) фрагментов (каких-либо объектов) на изображении. Для этого некоторые фрагменты изображения (хранящегося в файле) должны быть выделены, после чего должны быть описаны их ограничивающие прямоугольники. Координаты ограничивающих прямоугольников должны быть добавлены в файл «модифицированного» формата (например, в качестве отдельного блока данных). Обработка содержимого производится в исходном формате (в формате, в котором производится хранение в файле), после чего формируется новый файл «модифицированного» формата, в котором производится запись необходимых данных, описывающих габариты фрагментов изображения.

Произвести документирование (привести структуру нового формата файла, кратко пояснить основные элементы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.В2: 2 балла.

Рабочая задача КВ.Д.В3: модифицировать исходный формат графического файла (формат, определяемый текущим вариантом коллективного задания) таким образом, чтобы вместо одного изображения (определяемое пикселями) было возможно хранить несколько изображений, представляющих собой при последовательном показе анимацию. Для этого должны быть внесены некоторые изменения в структуру формата файла – модифицирован заголовок, либо добавлены «маркеры», отмечающие кадры анимации. Также необходимо модифицировать функционал программного модуля таким образом, чтобы вместо статичного изображения происходило отображение очередного кадра анимации, загруженной из файла.

Произвести документирование (привести структуру нового формата файла, кратко пояснить основные элементы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.В3: 2 балла.

Рабочая задача КВ.Д.В4: реализовать функционал «преобразования» графического файла (заголовков, палитры, пикселей) исходного формата (формат, предлагаемый текущим вариантом коллективного задания) в другой (целевой) формат (формат, предлагаемый другим вариантом коллективного задания). Обработка пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего формируется новый файл (целевого формата), в который производится запись преобразованного содержимого.

Произвести документирование (привести структуру нового формата файла, кратко пояснить основные элементы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.В4: 3 балла.

Рабочая задача КВ.Д.В5: реализовать функционал «преобразования» графического файла (заголовков, палитры, пикселей) исходного формата (формат, предлагаемый текущим вариантом коллективного задания) в известный

формат растровых файлов «Microsoft BMP». Обработка пикселей производится в исходном формате (в формате, в котором производится хранение в файле), после чего формируется новый файл формата «BMP-24» (24-х разрядный точечный рисунок «BMP»), в котором производится запись необходимых элементов данного формата и непосредственных пикселей изображения, преобразованных к корректному формату целевого файла.

Для преобразований не допускается использование «стандартных» средств среды разработки, платформы или ОС – необходимо выполнить разработку алгоритма и программную реализацию побайтовой записи файла формата «BMP» самостоятельно.

Произвести документирование (привести структуру формата «BMP», кратко пояснить основные элементы), произвести устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.В5: 3 балла.

4.3.4 Рабочие задачи группы «Г» – реализация дополнительного функционала программного модуля

Рабочая задача КВ.Д.Г1: разработать элемент (элементы) графического пользовательского интерфейса (диалоговое окно, оконную форму, или специализированный компонент), позволяющий выполнять просмотр и изменение цветовых значений палитры (формата, определяемого текущим вариантом коллективного задания), загруженной из файла. После завершения изменения палитры в программном модуле, файл перезаписывается (по крайней мере, его палитра обновляется), загружается и отображается заново (уже с обновлённой палитрой). Произвести документирование – письменное и устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Г1: 1 балл.

Рабочая задача КВ.Д.Г2: разработать элемент (элементы) графического пользовательского интерфейса (диалоговое окно, оконную форму, или специализированный компонент), позволяющий выполнять просмотр и изменение значений в заголовке загруженного файла (формата, определяемого текущим вариантом коллективного задания). Изменение значений ширины и высоты приведёт к изменению размеров изображения (изменению количества пикселей), изменение значения количества элементов палитры (не для всех вариантов) приведёт к изменению количества элементов палитры, изменение глубины цвета (бит на пиксель) не требуется. Произвести документирование – письменное и устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Г2: 1 балл.

Рабочая задача КВ.Д.Г3: совместно с выполнением «КВ.Д.А1», «КВ.Д.А2», «КВ.Д.А3», «КВ.Д.А4», «КВ.Д.А5», «КВ.Д.А6» выполнить разработку элемента (элементов) графического пользовательского интерфейса (диалоговое окно, оконную форму, или специализированный компонент), позволяющий пользователю выполнять настройку параметров преобразования пикселей (в зависимости от вида преобразования). Произвести документирование – письменное и устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Г3: 1 балл.

Рабочая задача КВ.Д.Г4: разработать элемент (элементы) графического пользовательского интерфейса (диалоговое окно, оконную форму, или специализированный компонент), позволяющий пользователю выполнять выделение некоторой области изображения и сохранение данной области в новый файл (этого же формата). При сохранении должна производиться коррекция заголовка сохраняемой области. Произвести документирование – письменное и устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Г4: 1 балл.

Рабочая задача КВ.Д.Г5: совместно с выполнением «КВ.Д.Б6» и «КВ.Д.Б7» разработать элемент (элементы) графического пользовательского интерфейса (диалоговое окно, оконную форму, или специализированный компонент), позволяющий пользователю выполнять настройку параметров текстуры (узора): размер элементов текстуры, расстояние между элементами, а также настройку параметров фрактала: размер, глубину рекурсии, цвет контура и закраски. Произвести документирование – письменное и устное пояснение аспектов реализации.

Балл за выполнение рабочей задачи КВ.Д.Г5: 2 балла.

4.3.5 Рабочие задачи группы «Д» – применение программно-аппаратных средств обработки данных

Рабочая задача КВ.Д.Д1: при выполнении операций над цветовыми значениями или координатами заменить все элементарные арифметическо-логические действия, выполняемые средствами языка программирования или среды выполнения в программном коде, ассемблерными вставками, реализующими элементарные арифметическо-логические действия: сдвиги, целочисленное сложение, целочисленное умножение и деление. Произвести документирование – письменное и устное пояснение используемых ассемблерных команд.

Балл за выполнение рабочей задачи КВ.Д.Д1: 1 балл.

Рабочая задача КВ.Д.Д2: при выполнении операций над цветовыми значениями или координатами заменить все элементарные одинарные арифметическо-логические действия, выполняемые средствами языка программирования или среды выполнения в программном коде, векторными (параллельными) действиями, реализуемыми встроенными или сторонними библиотеками, которые используют SIMD-инструкции x86 (MMX, SSE, AVX) для обработки векторных значений (например, суммы или разности цвета в векторном представлении и др.).

Балл за выполнение рабочей задачи КВ.Д.Д2: 1 балл.

Рабочая задача КВ.Д.Д3: при выполнении операций над цветовыми значениями или координатами заменить все элементарные одинарные арифметическо-логические действия, выполняемые средствами языка программирования или среды выполнения в программном коде, векторными (параллельными) действиями, реализуемыми в ассемблерных вставках, в которых напрямую задействуются SIMD-инструкции x86 (MMX, SSE, AVX) для обработки векторных значений (например, суммы или разности цвета в векторном представлении и др.).

Балл за выполнение рабочей задачи КВ.Д.ДЗ: 2 балла.

4.4 Задания индивидуальных вариантов

4.4.1 Рабочая задача ИВ.О.1. Выполнение задания коллективного варианта средствами вывода графики Windows

Выполнить обязательные рабочие задачи коллективного варианта, при этом реализовать вывод всей графики в рамках разрабатываемого приложения только посредством использования вызовов функций Windows API (GDI/GDI+). Допускается использование заголовочных файлов Windows.h или Windows.pas, либо их аналогов, сохраняющих типы данных и сигнатуры функций WinAPI, экспорт функций библиотек WinAPI. Использование каких-либо других встроенных либо внешних подключаемых модулей для вывода графики не допускается. Создание экранных форм (окон) может произведено любым способом (использование WinAPI для создания форм не является обязательным);

Балл за выполнение индивидуального задания: 9 баллов + 3 бонус.

4.4.2 Рабочая задача ИВ.О.2. Выполнение задания коллективного варианта в среде DOS

Выполнить обязательные рабочие задачи коллективного варианта, при этом использовать среду разработки Turbo C/C++, Turbo Pascal, Borland C/C++, либо Borland Pascal и ОС DOS. Реализовать вывод всей графики в рамках разрабатываемого приложения посредством использования вызовов стандартных библиотек среды разработки, операционной системы DOS или вызовов BIOS;

Балл за выполнение индивидуального задания: 9 баллов + 3 бонус.

4.4.3 Рабочая задача ИВ.О.3. Выполнение задания коллективного варианта при помощи TASM

Выполнить обязательные рабочие задачи коллективного варианта, при этом программный модуль необходимо разработать посредством СРПО TASM для среды DOS 6.xx, а вывод графики реализовать при помощи прерываний int10h в графическом режиме VGA/EGA;

Балл за выполнение индивидуального задания: 9 баллов + 6 бонус.

4.4.4 Рабочая задача ИВ.О.4. Разработка растрового редактора

Выполнить разработку программного модуля, содержащего основной функционал растрового редактора со следующим обязательным функционалом:

- Отрисовка одного пикселя в заданных координатах (пользователем в указанных при помощи мыши координатах) изображения;
- Непрерывная отрисовка группы пикселей в изменяющихся координатах посредством движения мыши (аналог инструмента «кисть» Paint с «толщиной»);
- Отрисовка контура прямоугольника и закрашенного прямоугольника заданного цвета в заданных координатах (при помощи мыши);
- Отрисовка загруженного из файла изображения в заданных координатах (указанные при помощи мыши) на текущем;
- Копирование и вставка указанной (при помощи мыши) части изображения в другую часть этого же изображения;

- Сохранение и загрузка изображения в один из форматов графического файла, предлагаемого одним из коллективных вариантов задания (по выбору исполнителя);

Вся обработка пикселей (отрисовка) должна производиться над пикселями в исходном формате (в формате, в котором производится хранение в файле согласно варианту).

Балл за выполнение индивидуального задания: 9 баллов.

4.4.5 Рабочая задача ИВ.О.5. Разработка растрового редактора для среды DOS

Выполнить разработку программного модуля, содержащего основной функционал растрового редактора со следующим обязательным функционалом:

- Отрисовка одного пикселя в заданных координатах (пользователем в указанных при помощи мыши координатах) изображения;
- Непрерывная отрисовка группы пикселей в изменяющихся координатах посредством движения мыши (аналог инструмента «кисть» Paint с «толщиной»);
- Отрисовка контура прямоугольника и закрашенного прямоугольника заданного цвета в заданных координатах (при помощи мыши);
- Отрисовка загруженного из файла изображения в заданных координатах (указанные при помощи мыши) на текущем;
- Копирование и вставка указанной (при помощи мыши) части изображения в другую часть этого же изображения;
- Сохранение и загрузка изображения в один из форматов графического файла, предлагаемого одним из коллективных вариантов задания (по выбору исполнителя);

Вся обработка пикселей (отрисовка) должна производиться над пикселями в исходном формате (в формате, в котором производится хранение в файле согласно варианту). Программный модуль должен быть скомпилирован под среду DOS x86.

Балл за выполнение индивидуального задания: 9 баллов + 3 бонус.