## МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ Федеральное государственное бюджетное образовательное учреждение высшего образования «ПЕРМСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»

## КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ В ЭКОЛОГИИ И ПРИРОДОПОЛЬЗОВАНИИ

Под общей редакцией М. А. Даниловой

Допущено методическим советом Пермского государственного национального исследовательского университета в качестве учебного пособия для студентов, обучающихся по направлению подготовки магистров «Экология и природопользование»



Пермь 2018

Коллектив авторов: М. А. Данилова, Ю. С. Васильева, В.П. Красильников Под общей редакцией М. А. Даниловой

Компьютерные технологии в экологии и природопользовании К637 [Электронный ресурс]: учеб. пособие / М. А. Данилова, Ю. С. Васильева, В. П. Красильников; под общ. ред. М. А. Даниловой; Перм. гос. нац. исслед. ун-т. – Пермь, 2018. – 8 Мб; 174 с. – Режим доступа: http://www.psu.ru/files/docs/science/books/ uchebnieposobiya/kompyuternye-tekhnologii-v-ecologii-i-prirodopolzovanii.pdf. – Загл. с экрана.

ISBN 978-5-7944-3223-7

Учебное пособие посвящено современным технологиям обработки и визуализации данных. В результате освоения материала учебного пособия студенты приобретут базовые знания в области компьютерной графики на примере редакторов векторной и растровой графики, таких как CorelDRAW и Adobe Photoshop, а также овладеют навыками работы с открытыми универсальными статистическими средами на примере RStudio.

Содержание учебного пособия охватывает вопросы статистической обработки и визуализации данных. Их знание поможет студентам работать с современными программами для статистической обработки данных и различными графическими редакторами.

Подготовлено для студентов магистратуры, обучающихся по направлению 05.04.06 «Экология и природопользование».

УДК 502:004.9 (075.8) ББК 28.088+32.973

Печатается по решению ученого совета биологического факультета Пермского государственного национального исследовательского университета

Рецензенты: канд. биол. наук, доцент кафедры биологии, экологии и медицинской генетики ПГМУ им. академика Е. А. Вагнера *Е. А. Логинова*; кафедра фармакогнозии с курсом ботаники ПГФА (зав. каф. – д-р фармацевт. наук, проф. *В. Д. Белоногова*)

© ПГНИУ, 2018

ISBN 978-5-7944-3223-7

© Данилова М. А., Васильева Ю. С., Красильников В. П., 2018

## оглавление

ВВЕДЕНИЕ	4
Раздел 1. КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ В БИОЛОГИИ И ЭКОЛОГИИ	6
Глава 1. Компьютерные технологии при работе с графической	
информацией	6
Цветовые модели в компьютерной графике	9
Шрифт и текст в компьютерных технологиях	19
Методы представления графической информации	
Форматы графических файлов	
Глава 2. Программирование для представления и обработки биологиче	ской
и экологической информации	50
Инструменты статистического анализа	50
Язык программирования R	52
Другие языки программирования	54
Раздел 2. ИСПОЛЬЗОВАНИЕ КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ в экологии и природопользовании	50
Блава 3. Графицасина радаиторы 1	
Плава 5. Графические редакторы	50
Редактор векторной графики СогепDRA w	
Редактор растровой графики Adobe Photosnop	
Глава 4. Система статистического анализа и визуализации данных R	141
Начало работы с RStudio. Организация рабочей среды	141
Визуализация данных в R	165
Специальные возможности R в области экологии и природопользования	168
Список литературы	171

#### введение

Использование компьютерных технологий в различных областях науки и при решении образовательных задач стало нормой в последние годы. Научить студента самостоятельно анализировать новую информацию и делать грамотные выводы всегда было одной из задач высшей школы, но именно сейчас эта задача стала наиболее актуальной. Данное учебное пособие знакомит студентов с основами компьютерной графики и программирования при анализе разнообразных данных, связанных с биологией и экологией. Оно призвано помочь сформировать следующие профессиональные компетенции: знание современных компьютерных технологий, применяемых при сборе, хранении, обработке, анализе и передаче способность информации; самостоятельно использовать современные компьютерные технологии научно-исследовательских для решения И производственно-технологических задач профессиональной деятельности; владение методами оценки репрезентативности материала, объема выборок при проведении количественных исследований, а также статистическими методами сравнения полученных данных и определения закономерностей.

Владение компьютерными методами визуализации информации становится в современном мире необходимым навыком, таким же базовым, как письмо и счет. Связать вместе необходимый объем вербальной информации с визуальным рядом для максимально эффективного восприятия этой информации возможно с помощью инфографики. По определению В.В. Лаптева, «Инфографика – это область коммуникативного дизайна, в основе которой лежит графическое представление информации, связей, числовых данных и знаний». Эдвард Тафт, один из основоположников информационного дизайна, дает более краткое определение: «Инфографика – это графический способ подачи информации, данных и знаний». Сама по себе инфографика не нова, к ней можно отнести разнообразные проявления графической подачи информации, от пиктограмм и карт до иллюстрации книг и инженерных чертежей. Однако важность инфографики в условиях лавинообразного накопления информации и одновременного развития компьютерных технологий

позволило по-новому взглянуть на нее. Некоторые современные массивы данных настолько сложны, что стало необходимым разрабатывать новые подходы к их визуализации (Искусство дизайна, 2005).

В научных статьях или на сайтах использование информационных графиков преследует следующие цели:

• упростить детализированную информацию;

• отсеять информацию, прочтение которой в противном случае могло бы занять слишком много времени;

- вызвать интерес читателя;
- разорвать длинные блоки текста, чтобы избежать утомления внимания.

Особое значение следует придавать анализу экологических данных. Экологический мониторинг информационной основой является для природоохранной широкого Полученные деятельности спектра. данные используются для научных исследований, оценки состояния окружающей среды и решений. Глобальная принатия управленческих система экологического мониторинга позволяет получать представление о неоднозначном воздействии антропогенной деятельности на окружающую среду. Для анализа полученных данных, а также для прогнозирования исходя из результатов обработки этих данных выделяются колоссальные вычислительные мощности. Удобная и гибкая система анализа данных является крайне важной для решения задач экологического мониторинга. Использование R или других подобных языков программирования и статистической обработки дает возможность применять любые виды анализа, сложные модели, сопоставлять данные ИЗ разных источников строить И визуализировать результаты удобным образом.

Учебное пособие предназначено для студентов, уже владеющих первоначальными навыками работы с компьютером. В результате изучения данного учебного курса они приобретут базовые умения в области компьютерной графики и освоят основы работы с открытыми универсальными статистическими средами на примере RStudio.

### Раздел 1. КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ В БИОЛОГИИ И ЭКОЛОГИИ

## Глава 1. Компьютерные технологии при работе с графической информацией

В компьютерах первых поколений форма представления результатов решения задач была очень громоздкой и не наглядной – необозримые колонки чисел или огромные таблицы. Очень часто, чтобы облегчить восприятие этой информации, приходилось вручную строить диаграммы, рисовать графики или чертежи. Известно, что в графическом виде информация становится более понятной, лучше воспринимается человеком. Поэтому возникла идея поручить компьютерам графическую обработку информации. Так осуществлять появились графопостроители (или плоттеры), с помощью которых компьютер смог рисовать графики, чертежи, диаграммы. Однако это был только первый шаг в компьютерной графике. Следующим, принципиально новым шагом стало создание графических дисплеев. На графическом дисплее совокупности точек (так называемых «пикселей») различного цвета позволяют создавать статическое И лаже динамическое (движущееся) изображение. Работой графического дисплея управляет графический адаптер, состоящий из двух частей: видеопамяти и дисплейного процессора. Видеопамять служит для хранения видеоинформации – двоичного кода изображения. Дисплейный процессор непрерывно «просматривает» содержимое видеопамяти (несколько десятков раз в секунду) и выводит его на экран (Павлова и др., 2009).

Графический дисплей строит цифровые изображения. *Цифровым* называется изображение, созданное с использованием компьютерной программы с нуля; либо изображение (слайд, фотография), преобразованное в электронную информацию для того, чтобы просматривать, редактировать и управлять им на экране компьютера. Устройства, преобразующие графические изображения в цифровую форму, называются оцифровывающими (сканеры, цифровые фотоаппараты и т.п.).

Компьютерная графика – использование вычислительной техники для создания цифровых графических изображений, их отображения различными средствами и манипулирования ими. Слово «графика» означает изображение линиями, штрихами, точками, т. е. каким бы сложным ни казалось изображение, созданное в компьютере, по своей сути любое из них относится к графике. Несмотря на то что для работы с компьютерной графикой существует множество классов программного обеспечения, различают всего три вида компьютерной графики. Это растровая графика, векторная графика и фрактальная графика. Они различаются принципами формирования изображения при отображении на экране монитора или при печати на бумаге. Растровую графику применяют при разработке электронных (мультимедийных) и полиграфических изданий. Иллюстрации, выполненные средствами растровой графики, редко создают вручную с помощью компьютерных программ. Чаще для этой цели используют отсканированные бумаге, фотографии. иллюстрации, подготовленные художником на ИЛИ Соответственно, большинство графических редакторов, предназначенных для работы с растровыми иллюстрациями, ориентированы не столько на создание изображений, сколько на их обработку. Программные средства для работы с векторной графикой предназначены в первую очередь для создания иллюстраций и в меньшей степени для их обработки. Такие средства широко используют в дизайнерских бюро, рекламных агентствах. редакциях И издательствах. Оформительские работы, основанные на применении шрифтов и простейших геометрических элементов, решаются средствами векторной графики намного проще. Программные средства для работы с фрактальной графикой предназначены для автоматической генерации изображений путем математических расчетов. Создание фрактальной художественной композиции состоит не в рисовании или оформлении, а в программировании. Фрактальную графику часто используют в развлекательных программах (Васильев, Морозов, 2005).

#### Основные понятия компьютерной графики

В компьютерной графике следует четко различать следующие характеристики: разрешение экрана, разрешение печатающего устройства и

разрешение изображения. Все эти понятия относятся к разным объектам. Друг с другом эти виды разрешения никак не связаны пока не потребуется узнать, какой физический размер будет иметь картинка на экране монитора, отпечаток на бумаге или файл на жестком диске. Разрешение экрана – это свойство компьютерной системы (зависит от монитора и видеокарты) и операционной системы (зависит от настроек ОС). Разрешение экрана измеряется в пикселях и определяет размер изображения, которое может поместиться на экране целиком. Разрешение принтера - это свойство принтера, выражающее количество отдельных точек, которые могут быть напечатаны на участке единичной длины. Оно измеряется в единицах dpi (точки на дюйм) и определяет размер изображения при заданном качестве или, наоборот, качество изображения при заданном размере. Разрешение изображения – это свойство самого изображения. Оно тоже измеряется в точках на дюйм и задается при создании изображения в графическом редакторе или с помощью сканера. Значение разрешения изображения хранится в файле изображения и неразрывно связано с другим свойством изображения – его физическим размером. Физический размер изображения может измеряться как в пикселях, так и в единицах длины (миллиметрах, сантиметрах, дюймах). Он задается при создании изображения и хранится вместе с файлом. Если изображение готовят для демонстрации на экране, то его ширину и высоту задают в пикселях, чтобы знать, какую часть экрана оно занимает (Гумерова, 2013).

Современное применение компьютерной графики очень разнообразно. Рассмотрим основные области применения компьютерной графики в биологии и природопользовании. К данному направлению чаще всего относят понятие научной графики. *Научная графика* – визуализация (наглядное изображение) объектов научных исследований, графическая обработка результатов расчётов, проведение вычислительных экспериментов с наглядным представлением их результатов. Можно выделить несколько основных задач, решаемых в данной области применения компьютерной графики:

1. Цифровая фотография, фотодокументирование в биологических и экологических исследованиях. Масштабирование и трансформирование изображений. Оптимизация фотоизображений средствами редакторов графики.

2. Компьютерная планиметрия и морфометрия, определение размеров и площади биологических объектов.

3. Документальное оформление результатов исследований. Векторизация изображений. Иллюстрирование текстовых документов вставкой и внедрением графических объектов.

4. Визуализация данных в научных исследованиях и образовании. Программные средства создания диаграмм и графиков, научная графика и сплайны.

5. Создание и демонстрация компьютерных презентаций, сопровождающих научные доклады и отражающих материалы квалификационных работ. Постерное представление результатов исследований.

#### Цветовые модели в компьютерной графике

Если принять всю информацию, которую получает человек из окружающего мира, за 100%, то доля информации, получаемой через органы зрения (по разным источникам), будет равняться от 80 до 90%. Таким образом, значимость зрительной информации для человека огромна, однако почему мы видим мир именно так, за счёт чего это достигается? Мы видим предметы окружающего мира в зависимости от того, какую способность, поглощать или отражать световые потоки, они имеют. Если вы читали роман Г. Уэллса «Человек-невидимка», то помните, что герой романа создал вещество, которое в полной мере было способно отражать световые потоки, что и позволяло становиться невидимым. Однако реальность довольно далека от фантастики, и большинство предметов окружающего мира не имеет 100%-ной отражательной способности, именно поэтому мы способны их видеть. Для полного процессов цветоощущения необходимо понимания иметь представление о некоторых основополагающих понятиях этого явления. Ключом ко

всему является свет. Ещё из школьного курса физики вы знаете, что *свет* – электромагнитное излучение, воспринимаемое человеческим глазом. Свет может быть излучаемым, а может быть отраженным.

*Излучаемый свет* – это свет, испускаемый активным источником (Солнце, звезды, лампочка, монитор).

*Отраженный свет* – свет, возникающий при отражении поверхностью предметов световых волн, падающих на нее от источника света.

Свет, попадающий на глаз непосредственно от источника, не претерпевает никаких изменений и доходит до глаза в своём изначальном состоянии, однако если в глаз попадает свет, отраженный каким-либо объектом, то тогда с ним происходят определенные изменения. Вспомните школьные опыты по оптике: поток света, проходя через призму, разбивается на составляющие части спектра видимого излучения, представленные красным, оранжевым, желтым, зеленым, голубым, синим и фиолетовым цветом. Так мы подходим к основной для нас характеристике света – цвету.

*Цвет* – это субъективная характеристика света, воспринимаемая человеком через зрительные органы. Следует понимать, что цветоощущение крайне субъективное явление и зависит от ряда физических, физиологических и психологических факторов. Таким образом, глупо спорить о том, какой это цвет, смысл в данном случае может иметь лишь измерение состава излучения (рис. 1).



Рис. 1. Излучение, поглощение и отражение света

Монитор компьютера, подобно Солнцу или лампочке, является источником света. Бумага (или какая-либо иная поверхность, на которой печатают изображение) отражает свет. Исходя из этого на человеческий глаз могут падать световые волны как в процессе излучения, так и в процессе отражения. Именно поэтому существует две модели цветового описания – *аддитивная* и *субтрактивная*.

#### Аддитивная цветовая модель (RGB)

Данная цветовая модель получила своё название за счет добавления (англ. *addition*) цветов к черному цвету. Данная технология широко используется при передаче цвета в электронных устройствах, будь то экран смартфона или монитор компьютера. Если мы посмотрим на монитор под увеличительным стеклом, то увидим, что он весь состоит из большого количества светодиодов (электронных пушек), которые подразделяются на три типа в зависимости от излучаемого цвета: красный (*red*) – R; зеленый (*green*) – G; синий (*blue*) – В. Таким образом, аддитивную модель цветопередачи ещё называют модель RGB либо модель K3C (на русский манер), а цвета красный, зеленый и синий – *базовыми* цветами.

Так как светодиоды имеют маленькие размеры, при должном разрешении дисплеев человеческий глаз не способен различать их по отдельности, поскольку процедура смешивания цветов в один. Различные комбинации цветового взаимодействия светодиодов дают различные вариации цветовой палитры (рис. 2):

- Красный (R) + синий (B) = маджента (M).
- Красный (R) + зеленый (G) = желтый (Y).
- Зеленый (G) + синий (B) = голубой (C).
- Красный (R) + зеленый (G) + синий (B) = белый (W).
- Отсутствие цветов дает черный цвет (К).



Рис. 2. Цветовая модель RGB

Различные оттенки получают посредством изменения интенсивности свечения тех или иных светодиодов. Цветовая модель RGB адекватна цветовому восприятию человеческого глаза, рецепторы которого также настроены на красный, зеленый и синий цвета.

#### Субтрактивная цветовая модель (СМҮК)

Способ передачи цветовой палитры отраженного света описывает субтрактивная цветовая модель. Так как бумага и другие материалы, используемые для печати, отражают свет, было принято считать количество света, отраженного поверхностью, а не поглощенного. Если из белого цвета вычесть три базовых RGB, тройку дополнительных цветов СМҮ. Таким то ΜЫ получим образом. субтрактивная цветовая модель получила своё название от вычитания (англ. subtract) базовых цветов из белого цвета.

В модели СМҮК (рис. 3) используются четыре цвета, первые три в аббревиатуре названы по первой букве английских слов, описывающих соответствующий цвет: голубой (*cyan*) – C; пурпурный или маджента (*magenta*) – M; желтый (*yellow*) – Y; а четвертая К обозначает черный. Согласно одной из версий, обозначение черного цвета буквой К принято считать попыткой избежать

путаницы черного цвета (*black*) и синего (*blue*) из модели RGB по первой букве и обозначать черный цвет по последней (*blacK*).



Рис. 3. Цветовая модель СМҮК

Как уже говорилось, модель СМҮК базируется на вычитании цветов из белого света. Это происходит благодаря тому, что белая бумага изначально отражает все цвета белого света, однако окрашенная бумага отражает лишь часть цветов белого света, остальную же часть она поглощает, что и обеспечивает цветовое восприятие при печати. Таким образом, различные сочетания цветов модели СМҮК (голубого, мадженты и желтого) на белой бумаге способны давать следующие варианты цветовой палитры:

- Голубой (С) + маджента (М) + желтый (Y) = черный (К).
- Голубой (С) + маджента (М) = синий (В).
- Желтый (Y) + маджента (M) = красный (R).
- Желтый (Y) + голубой (C)= зеленый (G).

• Белый цвет мы получаем при условии полного отсутствия всех трех основных цветов данной модели.

Теоретически черный цвет образуется при равном соединении базовых цветов на белом фоне, однако из-за погрешностей печати либо качества типографской краски зачастую получается темно-серый, либо темно-бурый цвет в различных градациях, поэтому при печати дополнительно добавляется черная краска.

# Взаимодействие аддитивной и субтрактивной цветовых моделей, передача цвета при печати

Цвет окружающих предметов определяется нашими глазами благодаря особым структурам – колбочкам и палочкам. По чувствительности к свету различных длин волн колбочки разделяют на три типа:

S-типа – фиолетово-синий спектр;

М-типа – зелено-желтый спектр;

L-типа – желто-красный спектр.

Модель RGB является более адекватной человеческому глазу, нежели модель CMYK, однако она также не способна в полной мере передать все цвета видимого нами спектра. Таким образом, наибольшее количество цветов мы можем увидеть в натуральном цветовом охвате, несколько меньше – в RGB и ещё меньше – в CMYK. Именно поэтому фотографии получаются более блеклые и тусклые по сравнению с настоящими объектами или их изображениями на экране (рис. 4).



Рис. 4. Сравнение спектров видимого света, RGB и CMYK

Важно помнить о проблеме соответствия спектров и учитывать это при работе в графических редакторах, таких как Adobe Photoshop или CorelDRAW. Данные редакторы способны предупредить пользователя о несоответствии цвета и цветовой модели, однако всегда нужно учитывать, с какой цветовой моделью вы работаете и какие цветовые спектры ей доступны. Для сравнения на рис. 5 представлены цветовые охваты моделей RGB и CMYK.



Рис. 5. Цветовые охваты моделей RGB и CMYK

Мы уже обсуждали, как происходит формирование цвета в каждой из моделей по отдельности, для представления процедуры взаимодействия этих моделей существует схема цветового круга (рис. 6). Однако следует иметь в виду, что в данной схеме рассмотрена ситуация использования модели СМҮК без черного цвета.



Рис. 6. Цветовой круг взаимосвязи моделей RGB и CMYK

Как вы помните, создание цвета в модели RBG происходит по принципу излучения, а в модели CMYK – по принципу вычитания, что значительно усложняет процедуру колоризации. Так, синий цвет в модели RGB требует полной работы синих диодов, а синий цвет модели CMYK требует определенного соотношения голубых и маджентовых красителей. В табл. 1 представлены описания некоторых цветов в моделях RGB и CMYK (диапазон изменений цвета – от 0 до 255).

Цвет	Модель RGB	Модель СМҮК
Синий	0R, 0G, 255B	100C, 79M, 0Y, 0K
Изумрудный	0R, 174G, 94B	76C, 2M, 82Y, 1K
Аквамарин	127R, 255G, 212B	33C, 0M, 25Y, 0K
Амарантовый	229R, 43G, 80B	1C, 91M, 58Y, 0K
Сомон	239R, 175G, 140B	4C, 40M, 42Y, 1K
Фуксия	247R, 84G, 225B	32C, 61M, 0Y, 0K

Таблица 1. Описание цветов в моделях RGB и CMYK

Важно помнить, что цветоделение в модели СМҮК идет не посредством сплошного цветового закрашивания областей, а формируется из отдельно окрашенных точек – *растров* (рис.7).



Рис. 7. Точечные растры

Растры формируют сетки определенного цвета, которые располагаются под разными углами относительно других сеток, что позволяет избегать перекрывания и

наложения цветов. Малые размеры растровых сеток обеспечивают эффект слияния и формируют цвет для нашего глаза.

#### Другие цветовые модели

*HSB*. В цветовой модели HSB (англ. *Hue, Saturation, Brightness* – тон, насыщенность, яркость) или HSV (англ. *Hue, Saturation, Value* – тон, насыщенность, значение) любой цвет получается из спектрального цвета добавлением определённого процента белой и чёрной краски, по сути, добавлением серой краски. Существует три основных понятия цветовой модели HSB:

Н - Оттенок (*hue*) – значение, определяющее положение цвета в спектре. Например, зеленый расположен между желтым и синим.

S - Насыщенность (*saturation*) –параметр управления цветом, чистота оттенка цвета в диапазоне от серого до чистого цвета.

В - Яркость (*brightness*) –яркость цвета по шкале от черного до белого на мониторе пользователя. Измеряется в процентах: от 0 до 100%. Нулевая яркость – это чёрный цвет.

Существует несколько вариантов визуализации данной цветовой модели, но наиболее широко применяется кольцевая модель (рис. 8):



Рис. 8. Цветовое кольцо модели HSV (HSB)

Радужное кольцо используется для подбора оттенка, а внутренний треугольник обеспечивает выбор насыщенности и значения цвета. Вертикальная ось

треугольника регулирует насыщенность, а горизонтальная позволяет изменять значение цвета. Таким образом, для выбора цвета нужно первоначально указать оттенок, а потом выбрать необходимый цвет из треугольника. Хотя модель HSB и декларирована как аппаратно-независимая, но все-таки в её основе лежит модель RGB. Так или иначе для отображения на мониторе HSB конвертируется в модель RGB, а для печати – в CMYK, что значительно влияет на качество итоговой продукции.

*LAB*. Неформальная аббревиатура этой модели (LAB) не определяет цветовое пространство однозначно. Чаще всего, говоря о пространстве LAB, подразумевается CIELAB (CIE 1976 L\*A\*B). Модель состоит из трёх каналов, взаимосвязь которых показана на рис. 9:

L (Lightness) – устанавливает координаты света (100) и тени (0);

а – спектр от зеленого (-128) до пурпурного (127);

b – спектр от голубого (-128) до желтого (127).



Рис. 9. Цветовое пространство модели LAB

LAB представляет собой трёхмерное пространство, где отрицательные значения а и b содержат холодные цвета, положительные – тёплые. Ось L определяет светлость, где 100 – участок, соответствующий диффузному белому. Изменение параметров L не влияет на насыщенность цвета – он сохраняет естественные свойства так же, как в реальности цвет не становится грязнее оттого, что на него падает тень. Черно-белая гамма получается при нулевых значениях координат a и b. LAB – это аппаратно-независимая цветовая модель, т.е. она не зависит от способа передачи нам цвета. Она содержит в себе цвета как RGB, так и CMYK, что позволяет ей с минимальными потерями конвертировать изображение из одной цветовой модели в другую. Еще одним достоинством является то, что она, в отличие от цветовой модели HSB, соответствует особенностям восприятия цвета глазом человека. Эта модель часто используется для улучшения качества изображения и конвертирования изображений из одного цветового пространства в другое.

#### ШРИФТ И ТЕКСТ В КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЯХ

#### Понятие шрифта. Структура букв шрифта

Огромный поток информации, которую современный человек воспринимает ежедневно, ежечасно, передается во многом с помощью шрифта. Шрифт, независимо от техники его исполнения, представляет собой упорядоченную графическую форму определенной системы письма. В полиграфии он является важнейших оформления любой печатной ОДНИМ ИЗ средств продукции (Филиппович, 2012). Шрифт – это графический рисунок начертаний букв и знаков, составляющих единую стилистическую и композиционную систему, набор символов определенного размера и рисунка. В узком, типографском смысле, шрифтом называется комплект типографских литер, предназначенных для набора текста. Термин «шрифт» раньше означал более узкое понятие – набор символов определенной гарнитуры, размера и начертания. Все элементы в хорошем шрифте гармонично сочетаются, и, таким образом, кроме прямого назначения – передачи информации – шрифты, использованные в публикации, являются элементом лизайна.

Все символы алфавита шрифта состоят из элементов, единство форм которых обеспечивает единство рисунка всей шрифтовой гарнитуры. Фактически каждый типовой элемент повторяется в различных буквах алфавита. Знаки кириллического и латинского алфавитов строятся на основе трех основных геометрических форм

или их комбинаций: прямоугольной или квадратной, округлой и треугольной (Филиппович, 2012). В каллиграфии и типографике сложились определенные анатомию букв латинского и термины, характеризующие кириллического алфавитов. Каждая буква состоит из штрихов: основных (более толстых) и соединительных. Современные печатные шрифты были созданы на основе рукописных, которые писались ширококонечным пером, отсюда переменная толщина штриха. К верхним выносным элементам буквы относятся штрихи строчных букв, которые выступают за среднюю линию. Нижние выносные элементы – это части букв, которые опускаются ниже линии шрифта. Размеры этих элементов сильно различаются у разных шрифтов, выносные элементы не выходят за пределы кегельной площадки, на которой они размещаются. Особенным элементом структуры буквы считается засечка – слегка расширяющийся росчерк на концах штрихов, благодаря которому существует определенная классификация шрифтов. Засечки – это не только декоративные элементы. Они играют важную роль в восприятии шрифта, поскольку помогают глазу отделить один знак от другого и выявить отдельные буквы. Они также упорядочивают горизонтальную текстуру шрифта, создавая своеобразную дорожку, которая уверенно ведет глаз вдоль строки (Филиппович, 2012).

#### Основные атрибуты текста

Стиль текста – это совокупность всех параметров оформления текста, присущих данному его отрезку. Атрибуты стиля текста включают следующие приемы оформления: гарнитура шрифта, начертание, кегль, интерлиньяж, межбуквенный просвет, междусловный пробел, выключка, отступ первой строки, втяжка (отступы справа и слева), межабзацные отбивки и др.

Гарнитура шрифта – набор из одного или нескольких шрифтов в одном или нескольких размерах и начертаниях, имеющих стилевое единство рисунка и состоящих из определённого набора типографских знаков. Гарнитура обычно содержит алфавитно-цифровые и пунктуационные знаки и специальные символы. Существуют также гарнитуры, целиком состоящие из неалфавитных символов, – например, содержащие математические или картографические знаки. Термин

«гарнитура» часто смешивают с термином «шрифт», различие между терминами состоит в том, что шрифт определяет свойства конкретного члена семейства шрифтов, например, полужирное или курсивное начертание, в то время как гарнитура определяет согласованный стиль семейства шрифтов. Каждая гарнитура имеет своё наименование. Гарнитуры можно разделить на две основные категории: *с засечками* (антиква и брусковые шрифты) и *без засечек* (гротески).

Начертание – вариант рисунка наборного шрифта одной гарнитуры, т.е. комплект строчных и прописных знаков, цифр, знаков препинания. Начертания шрифта характеризуются едиными стилевыми особенностями рисунка, определенными пропорциями (шириной), насыщенностью, наклоном знаков курсивное, наклонное), декоративной обработкой контура (прямое. знаков (контурное, оконтуренное, оттененное и др.). Насыщенность шрифта определяется изменением толщины основных и соединительных штрихов одноименных знаков в различных начертаниях и может меняться от светлой до сверхжирной – светлое, полужирное, жирное и др. Ширина шрифта зависит от пропорций букв: отличают шрифт с уменьшенной шириной букв от шрифта с увеличенной шириной букв, различают начертания узкое, нормальное, широкое и др. Наклонный шрифт образуется простым искажением (наклоном) символов исходного шрифта. В курсивном же начертании буквы имеют сходство с прямыми лишь по стилю, но рисунок их совсем другой. Наклонное начертание может быть создано программой, изначально содержаться В файле шрифта. Наиболее курсивное должно распространенные начертания – Normal (обычный), Bold (полужирный), Italic (курсив или наклонный), Bold Italic (полужирный курсив или наклонный), *Condensed* (узкий) и *Extended* (широкий).

И, пожалуй, нельзя не упомянуть о такой характеристике шрифта, как комплектность. *Комплектность* определяется полнотой знаков, необходимых для набора текста: строчных, прописных, цифр, знаков препинания и т.д.

#### Метрические атрибуты текста

Любой символ располагается в прямоугольнике – на кегельной площадке. Высота этого прямоугольника – это кегль, т.е. величина площадки, на которой

размещается знак (буква). Размер кегля определяется в пунктах. В основу измерений шрифтов положена система Дидо, распространенная в Европе и принятая в России, и так называемая англоамериканская система, или система Пика. В любом случае основной единицей измерения является типографский пункт, равный в системе Дидо 0,376 мм, а в системе Пика – 0,352 мм. Еще во времена металлического набора у наборщиков сложилось профессиональное наименование кеглей различного размера: кегль 6 пунктов – нонпарель, кегль 8 пунктов – петит, кегль 9 пунктов – боргес и т.д.

Очень важной характеристикой шрифта является величина апроша (полуапроша), она определяет расстояние между буквами. Слишком плотные узкие апроши приводят к визуальному слипанию знаков, а слишком широкие – к тому, что каждая буква воспринимается отдельно (Филиппович, 2012). Процесс установки межбуквенных расстояний (апрошей) для шрифта в зависимости от кегля называется *трекингом*. Трекинг задается для нескольких символов и характеризует величину межсимвольного пробела в группе символов. Пробелы меняются одинаково для всех выделенных символов. Как бы хорошо не были расставлены апроши в шрифте, в реальном наборе могут встретиться такие сочетания знаков, которые все равно будут образовывать визуальные дыры или сгущения и тем самым нарушать ритм чтения. Особенно это заметно в сочетаниях прописных знаков при крупном кегле: ГА, ТА, АТА, ЬТ и т. п. Визуальное выравнивание межбуквенных просветов в подобных сочетаниях называется кернингом, при этом происходит изменение ширины пробела для конкретных пар литер. Обычно кернинг делается в сторону уменьшения пробела таким образом, что выступающие части одной литеры пары заходят в пространство второй литеры (рис. 10), в результате размещение символов становится зрительно более равномерным. Кернинг и трекинг, будучи атрибутами символов, характеризуют не сами символы, а расстояние между ними, т.е. межсимвольные пробелы. Они необходимы для улучшения зрительного восприятия текста (рис. 10).



Рис.10. Пример кернинга (по Филиппович, 2012)

Программы верстки снабжены возможностью автоматического кернинга, они имеют собственные таблицы пар кернинга. В таких программах можно организовать автоматическую корректировку пробелов в парах кернинга в тексте любой длины. После обработки текста автоматическим кернингом в заголовках и другом крупном тексте дополнительно проводят ручной кернинг, т.е. символы сближают вручную. Чем крупнее текст, тем большее значение для его внешнего вида имеет кернинг. Кернинг и трекинг измеряются в специальных относительных единицах – тысячных долях круглой шпации (ems/1000).

Все буквы текста располагаются на одной линии – линии шрифта или базовой линии. Базовая линия – это линия, проходящая по нижнему краю основного элемента символа, который как бы лежит на ней. Расстояние между базовыми линиями соседних строк (складывается из кегля шрифта и расстояния между строками) называют интерлиньянж (рис. 11). Расстояние между строками (интерлиньяж) задается для всего абзаца. Общая высота строки равна кеглю самого крупного шрифта, использованного в ней, плюс небольшое расстояние. Стандартная величина интерлиньяжа определяется как 110-120% кегля, но так бывает не всегда. Выравнивание текста или выключка – способ расположения неполной набранной строки (например, заголовка) относительно вертикальных границ полосы набора. Строки могут быть выровнены по левому краю, при этом строка занимает не всю отведенную для неё ширину, оставляя свободное пространство справа (выключка влево); по правому краю при неровном левом (выключка вправо) или по обоим краям (полная выключка). Совпадение середины каждой строки с центральной осью полосы или колонки набора при неровных боковых краях называется выключка по центру. Набор, выключенный по одному краю и неровный с другого, называется флаговым. Разумеется, выравнивание не имеет смысла для отдельных символов, это атрибут абзаца. Для текста любого типа выравнивание может быть задано для каждого абзаца отдельно.



Рис. 11. Интерлиньяж (по Филиппович, 2012)

Атрибуты текста: а – литера; б – кегельная площадка; в – кегль; г – базовая линия шрифта; д – высота прописного знака; е – высота строчного знака; ж – межбуквенный просвет; з – полуапрош; и – интерлиньяж. Структура букв шрифта: 1 – основной штрих; 2 – соединительный штрих; 3 – засечка; 4 – верхний выносной элемент; 5 – нижний выносной элемент; 6 – 13 – структурно-декоративные элементы литер

#### Виды компьютерных шрифтов

*Растровые шрифты*. Изображение символов шрифта (как и любое другое) на экране дисплея является растровым изображением, т.е. состоит из окрашенных в разные цвета точек, или пикселей (*pixels*). В случае текста таких цветов только два – цвет символа и цвет фона. Если условно обозначить точку, окрашенную в цвет символа, единицей, а в цвет фона – нулем, любой изображаемый на экране символ можно представить в виде прямоугольной таблички из нулей и единиц – битовой карты (*bitmap*). Віtmap-шрифт является наиболее удобным для отображения на экране с точки зрения скорости прорисовки и затрат ресурсов компьютера на обработку. Однако размеры символов на экране могут изменяться – следовательно, для одной гарнитуры необходимы bitmap-шрифты нескольких размеров.

Векторные шрифты. В векторных шрифтах каждый символ представлен в виде совокупности геометрических примитивов – обычно отрезков прямых и дуг окружности, заданных своими координатами относительно «точки привязки» (origin point) символа. Масштабирование векторного шрифта производится простым умножением всех координат на соответствующий множитель. Однако для прорисовки шрифтов с качеством, пригодным для полиграфии, в векторных шрифтах понадобилось бы слишком большое количество элементов, образующих контуры букв с переменной толщиной, – эти контуры пришлось бы «набирать» из множества тонких линий. В настоящее время векторные шрифты используются только в некоторых программах, связанных с подготовкой чертежей.

Контурные шрифты. Более эффективным решением проблемы является использование так называемых «outline» (контурных) шрифтов. При необходимости отобразить на том или ином устройстве символ какого-то конкретного размера битовая карта для данного символа и данного кегля строится путем «заполнения» контуров буквы точками, размер которых соответствует разрешению устройства вывода, т.е. производится растрирование нужного символа на соответствующее разрешение. Для запоминания кривых, очерчивающих границы символов, используют разбиение кривой (или ломаной) линии на участки и замену получившихся фрагментов кривых полиномами второй или третьей степени (Филиппович, 2012).

#### Методы представления графической информации

#### Векторная графика

Векторная графика – вид компьютерной графики, в которой изображение представляется в виде совокупности отдельных объектов, описанных математически. Хотя на первый взгляд это может показаться сложнее, чем использование растровых массивов, но для некоторых видов изображений использование математических описаний является более простым способом. Ключевым моментом векторной графики является то, что она использует комбинацию компьютерных команд и математических формул для объекта. Это позволяет компьютерным устройствам вычислять и помещать в нужном месте реальные точки при рисовании этих объектов (Балыкина и др., 2008).

Принципы векторной графики основаны на математическом аппарате и имеют целью построение линейных контуров, составленных из элементарных кривых. Векторная графика описывает изображения с использованием прямых и изогнутых линий, называемых векторами, а также параметров, описывающих цвета и расположение. В векторной графике простые векторные объекты, из которых строятся изображения, называются графическими примитивами (прямая линия, дуга, эллипс, окружность, прямоугольник) и используются при создании более сложных объектов. Основным логическим элементом векторной графики является геометрический объект. В качестве объекта принимаются примитивы, составные фигуры или фигуры, построенные из примитивов, цветовые заливки, в том числе градиенты. Для создания объектов примитивов используются простые описания. Прямая дуги, окружности, эллипсы И области однотонного линия. ИЛИ изменяющегося света – это двухмерные рисунки, используемые для создания детализированных изображений. В трехмерной компьютерной графике для создания сложных рисунков могут использоваться такие элементы, как сферы, кубы.

*Линия* – это элементарный объект векторной графики. Любой сложный объект можно разложить на линии, прямые или кривые. Поэтому часто векторную графику называют объектно-ориентированной. Все, что есть в векторной иллюстрации, состоит из линий. Например, объект четырехугольник можно рассматривать как четыре связанные линии. Как и любой объект, линия обладает свойствами: формой (прямая, кривая), толщиной, цветом, начертанием или стилем (сплошная, пунктирная). Замкнутые линии имеют свойство заполнения – цветом, текстурой, узором и т.п. Охватываемое ими пространство может быть заполнено другими объектами (текстуры, карты) или выбранным цветом. Каждая незамкнутая линия имеет 2 вершины, называемые узлами. С помощью узлов можно соединять линии между собой. Узлы также имеют свойства, параметры которых влияют на форму

конца линии и характер сопряжения с другими объектами. Линия описывается математически как единый объект, и потому объем данных для отображения объекта средствами векторной графики существенно меньше, чем в растровой графике. При редактировании элементов векторной графики изменяются параметры прямых и изогнутых линий, описывающих форму этих элементов. Можно переносить элементы, менять их размер, форму и цвет, но это не отразится на качестве их визуального представления. Масштабирование в любую сторону осуществяется без потери качества, так как увеличение или уменьшение объектов производится увеличением или уменьшением соответствующих коэффициентов в математических формулах.

Векторная графика не зависит от разрешения, т.е. может быть показана в разнообразных выходных устройствах с различным разрешением без потери качества. Векторное представление заключается в описании элементов изображения математическими кривыми с указанием их цветов и заполняемости. Для компьютера подобные описания представляются в виде команд, каждая из которых определяет некоторую функцию и ее параметры. Пример: рисовать линию от точки А до точки В. Информация о цвете объекта сохраняется как часть его описания, т. е. в виде векторной команды. Различные векторные форматы обладают различными цветовыми возможностями. Простейшие форматы, которые могут не содержать вообще никакой информации о цвете, используют цвет по умолчанию тех устройств, на которые они выводятся, другие форматы способны сохранять данные о полном 32-битном цвете. Какую бы цветовую модель ни применял векторный формат, на размер файла он не влияет. В обычных векторных объектах значение цвета относится ко всему объекту в целом. Цвет объекта хранится в виде части его векторного описания (Балыкина и др., 2008).

Векторные команды сообщают устройству вывода о том, что необходимо нарисовать объект, используя максимально возможное число элементов (видеопикселей, или точек). Чем больше элементов используется устройством вывода для создания объекта, тем лучше этот объект выглядит. Команды, описывающие векторные объекты, большинству пользователей, возможно, никогда

не придется увидеть. Определять, как описывать объекты, будет компьютерная программа, которая используется для подготовки векторных объектов. Для создания векторных рисунков необходимо использовать один из многочисленных иллюстрационных пакетов. Важным объектом векторной графики является сплайн. Сплайн – это кривая, посредством которой описывается та или иная геометрическая фигура. На сплайнах построены современные шрифты *TrueType* и *PostScript* (Балыкина и др., 2008). Файлы векторной графики могут содержать растровые изображения в качестве одного из типов. Большинство векторных редакторов позволяют только разместить растровое изображение в векторной иллюстрации, изменить его размер, выполнить перемещение, поворот, обрезку, но не дают возможности работать с отдельными пикселями, так как объектом (примитивом) здесь является весь растровый фрагмент в целом. Векторное изображение можно строить вручную путем создания и объединения простейших контуров либо получать путем трассировки (векторизации) растровых изображений. Перед использованием векторного изображения очень часто выполняется операция перевода векторного изображения в растровое. Такая операция называется растрированием изображения.

Векторный формат более компактный, но он совершенно не пригоден для хранения фотографических изображений. В этом формате задавать их математически было бы очень громоздко. Ведь каждый мельчайший блик в этом случае будет представляться не совокупностью одноцветных точек, а сложнейшей математической формулой или совокупностью графических примитивов, каждый из которых является формулой. Это приводит к утяжелению файла. А вот рисунки и чертежи гораздо удобнее и практичнее делать именно в векторном виде.

Основными достоинствами векторной графики являются:

– Изменение масштаба без потери качества и практически без увеличения размеров исходного файла. Масштабирование, поворот, искривление могут быть сведены к паре-тройке элементарных преобразований над векторами. В тех областях графики, где важное значение имеет сохранение ясных и четких контуров,

например, в шрифтовых композициях, в создании логотипов и прочее, векторные программы незаменимы.

– Небольшой размер файла по сравнению с растровыми изображениями; это связано с тем, что сохраняется не само изображение, а только некоторые основные данные, используя которые, программа всякий раз воссоздает изображение заново. Кроме того, описание цветовых характеристик почти не увеличивает размер файла.

Огромная точность (до сотой доли микрона) и высокое качество печати.

Отсутствие проблем с экспортом векторного изображения в растровое.

 Возможность редактирования каждого элемента изображения в отдельности.

#### Основными недостатками являются:

Сложность экспорта из растрового в векторный формат. Векторизация приводит к наследованию последним невозможности корректного масштабирования в большую сторону. От увеличения линейных размеров количество деталей или оттенков на единицу площади больше не становится. Это ограничение накладывается разрешением вводных устройств (сканеров, цифровых) фотокамер и др.). Векторные рисунки состоят из различных команд, посылаемых от компьютера к устройствам вывода (принтеру). Принтеры содержат свои собственные микропроцессоры, которые интерпретируют эти команды и пытаются их перевести в точки на листе бумаги. Иногда из-за проблем связи между двумя процессорами принтер не может распечатать отдельные детали рисунков. В зависимости от типов принтера случаются проблемы, и у вас может оказаться чистый лист бумаги, частично напечатанный рисунок или сообщение об ошибке.

– Невозможность применения обширной библиотеки эффектов, используемых при работе с растровыми изображениями.

Природа избегает прямых линий. К сожалению, они являются основными компонентами векторных рисунков. До недавнего времени это означало, что уделом векторной графики были изображения, которые никогда не старались выглядеть естественно, например, двухмерные чертежи и круговые диаграммы, созданные

специальными программами САПР, двух- и трехмерные технические иллюстрации, стилизованные рисунки и значки, состоящие из прямых линий и областей, закрашенных однотонным цветом (Балыкина и др., 2008).

#### Математические основы векторной графики

В основе векторной графики лежат математические представления о свойствах геометрических фигур. Простейшим объектом векторной графики, как уже было сказано, является линия. Поэтому в основе векторной графики лежит, прежде всего, математическое представление линии. Рассмотрим подробнее способы представления различных объектов в векторной графике.

*Точка*. Это объект на плоскости, который задается двумя числами (x, y), определяющими ее положение относительно начала координат.

Прямая линия. Известно, что для задания прямой линии достаточно двух параметров. Обычно график прямой линии описывается уравнением y=kx+b. Зная параметры k и b, всегда можно нарисовать бесконечную прямую линию в известной системе координат.

*Отрезок прямой*. Он отличается тем, что требует для описания еще двух параметров – например, координат х<sub>1</sub> и х<sub>2</sub> начала и конца отрезка.

Кривая второго порядка. К кривым второго порядка относятся параболы, гиперболы, эллипсы, окружности и другие линии, уравнения которых не содержат степеней выше второй. Отличаются кривые второго порядка тем, что не имеют точек перегиба. Самая общая формула кривой второго порядка может выглядеть, например, так:

## $x^{2}+a_{1}y^{2}+a_{2}xy+a_{3}x+a_{4}y+a_{5}=0.$

Таким образом, для бесконечной описания кривой второго порядка достаточно пяти параметров. Если требуется построить отрезок кривой, понадобятся еще два параметра. Прямые линии являются всего лишь частным случаем кривых второго порядка.

*Кривая третьего порядка.* Отличительная особенность этих более сложных кривых состоит в том, что они могут иметь точку перегиба. Например, график функции у=х<sup>3</sup> имеет точку перегиба в начале координат. Именно эта особенность позволяет сделать кривые третьего порядка основой отображения природных объектов в векторной графике. Например, линии изгиба человеческого тела весьма близки к кривым третьего порядка. В общем случае уравнение кривой третьего порядка можно записать так:

 $x^{3}+a_{1}y^{3}+a_{2}x^{2}y+a_{3}xy^{2}+a_{4}x^{2}+a_{5}y^{2}+a_{6}xy+a_{7}x+a_{8}y+a_{9}=0.$ 

Таким образом, кривая третьего порядка описывается девятью параметрами. Описание ее отрезка потребует на два параметра больше. Все прямые и кривые второго порядка (например, окружности или эллипсы) являются частными случаями кривых третьего порядка.

*Кривые Безье*. Рисовать кривую третьего порядка по заданным коэффициентам ее уравнения – трудоемкая задача. Для упрощения этой процедуры в векторных редакторах применяют не любые кривые третьего порядка, а их особый вид, называемый кривыми Безье.

Отрезки кривых Безье – это частный случай отрезков кривых третьего порядка. Они описываются не одиннадцатью параметрами, как произвольные отрезки кривых третьего порядка, а лишь восемью, и потому работать с ними удобнее. Метод построения кривой Безье основан на использовании пары касательных, проведенных к линии в точках ее окончания (рис. 12). Для построения кривой требуется 4 контрольные точки. Но кривая физически проходит только через две из них – опорных. Одна из этих точек называется начальной, а другая – конечной. Две точки остаются в стороне, они получили название управляющих. Для того чтобы их не «потерять», в программах векторной графики управляющие точки соединяются с опорными точками линией (Васильев, Морозов, 2005).



Рис. 12. Кривая Безье (по Васильев, Морозов, 2005).

В программах векторной графики возможно интерактивное перемещение опорных и управляющих точек. Если перемещаются начальная или конечная точки, кривая станет соответствующим образом изменяться (вытягиваться или то сжиматься). Перемещение управляющих точек изменяет кривизну соответствующей части кривой Безье. Таким образом, с помощью перемещения этих четырех точек получают неограниченное количество форм кривой Безье, которая может быть лишь одним отдельным сегментом сложного векторного контура. В каждом сегменте можно добавлять опорные точки, которые тоже позволяют изменять форму кривой. На форму линии влияет угол наклона касательной и длина ее отрезка. Таким образом, касательные играют роль виртуальных «рычагов», с помощью которых управляют кривой.

#### Растровая графика

Минимальной единицей растровой графики является мелкий элемент – пиксель. Растровые изображения напоминают лист клетчатой бумаги, на которой каждая клетка закрашена каким-либо цветом, образуя в своей совокупности рисунок. Как можно заметить, принцип растровой графики чрезвычайно прост и понятен. Люди использовали отдаленное понятие растровой графики ещё задолго до создания портативного компьютера, ярким примером такого использования могут послужить мозаичные витражи либо схемы вышивки.



Рис. 13. Растровое изображение

В компьютерной графике термин пиксель обозначает достаточно большое количество понятий, таких как:

- Наименьший элемент на мониторе компьютера.
- Отдельный элемент растрового изображения.
- Точка изображения, напечатанного на принтере.

Таким образом, для избегания путаницы нам необходимо внести ясность и определить некий перечень терминов, которые мы и будем использовать в нашей дальнейшей работе.

• Видеопиксель – наименьший элемент изображения на мониторе, либо экране.

• Пиксель (px) – отдельный элемент растрового изображения.

• Точка – наименьший элемент, создаваемый принтером при печати изображения.

Важно понимать, что для воссоздания одного пикселя могут использоваться несколько видеопикселей либо точек. У растрового изображения существует несколько характеристик. Самыми важными следует считать: разрешение, размер и цветовая модель. В некоторых случаях среди пользователей происходит путаница понятий размера и разрешения. Дабы избежать возможных проблем, следует четко понимать принципиальную разницу этих определений.

*Разрешение* – это количество пикселей на дюйм – для описания изображения на экране либо точек на дюйм – в случае с печатным изображением.

*Размер* – общее количество пикселей в изображении, зачастую измеряется в мегапикселях (Мп). Данный параметр есть не что иное, как произведение количества пикселей по высоте на количество пикселей по ширине. Таким образом, если мы имеем изображение величиной 800 х 600, то его размер будет 800 х 600 = 480 000 пикселей, или 0,48 Мп.

Цветовая модель – характеристика изображения, описывающая его представления на основе цветовых каналов. Одни из наиболее часто применяемых цветовых моделей это: RGB, CMYK, HSB, LAB. Более подробно они были рассмотрены в этой главе ранее.

#### Преимущества растровой графики

• Растровая графика позволяет создать практически любой рисунок независимо от сложности, в отличие, например, от векторной, где невозможно точно передать эффект перехода от одного цвета к другому без потерь в размере файла.

• Распространённость – растровая графика в данный момент широко используется. Её можно встретить повсеместно, начиная от маленьких значков и заканчивая внушительными баннерами.

• Растровое представление изображения естественно для большинства устройств ввода-вывода графической информации, таких как мониторы (за исключением векторных устройств вывода), матричные и струйные принтеры, цифровые фотоаппараты, сканеры, а также сотовые телефоны.

#### Недостатки растровой графики

• Большой размер файла. Фактически для каждого пиксела приходится хранить информацию о его координатах и цвете.

• Ухудшение качества изображения при его увеличении, т.к. увеличивается размер точки (эффект пикселизации).

#### Особенности редакторов растровой и векторной графики

Для построения, коррекции, сохранения и получения «бумажных» копий рисунков и других изображений используется специальная программа – графический редактор. *Графические редакторы* – это инструменты компьютерного графика, с помощью которых он создает и редактирует изображения. В настоящее время существует много различных графических редакторов. Поэтому важно знать, какой редактор наилучшим образом подходит для решения конкретной задачи. Улучшение качества изображений, а также монтаж фотографий выполняются в редакторах растровой графики. Для создания иллюстраций обычно используются редакторы векторной графики, которые называют также программами рисования.

Любой графический редактор содержит набор инструментов для работы с Для создания изображений изображениями. используются определённые «инструменты» – линейка («отрезок»), прямоугольник, круг, эллипс и т.д. Такие изображать инструменты, позволяющие простые фигуры, называются «графическими примитивами». Можно без труда нарисовать окружность, квадрат или многоугольник произвольной формы. Для этого нужно выбрать в таблице инструментов графический примитив и установить курсор в соответствующую точку рабочего поля. Для прямоугольника в эту точку будет помещен левый верхний угол, для круга и эллипса – центр. Рисовать изображение можно как с помощью базовых инструментов (примитивов), так и в «ручном» режиме. При помощи графического редактора возможно строить изображения путем компоновки их из других, ранее созданных изображений, объединяя их с текстом и изменяя цвета. Поэтому в графическом редакторе реализованы функции, позволяющие:

- «вырезать», «склеивать» и «стирать» произвольные части изображения;
- «перемещать», «копировать», «удалять», «зеркально отражать» и «вращать отдельные» части изображений;
- применять для рисования произвольные «краски» и «кисти»;
- увеличивать фрагмент изображения для проработки мелких деталей;
- добавлять к рисункам текст;

- масштабировать (изменять размер) изображение, выполнять его перемещение и поворот;
- сохранять рисунки на внешних носителях, осуществлять их поиск и воспроизведение.

#### Среда графического редактора

Среда большинства графических редакторов состоит из трёх основных частей. С левой стороны экрана располагается набор пиктограмм (условных рисунков) с изображением инструментов, которыми можно пользоваться В процессе редактирования изображений – это инструментальная часть среды графического редактора. В нижней части экрана – другая часть среды – палитра, из которой можно выбрать краски требуемого цвета. Оставшаяся часть экрана представляет собой пустой «холст» (рабочее поле). Над рабочим полем находится третья часть меню, позволяющее изменять режимы работы. В левом нижнем углу экрана выводится калибровочная шкала, которая позволяет устанавливать ширину рабочего инструмента (кисти, резинки и т. д.).

#### Режимы работы графического редактора

Режимы графического редактора определяют возможные действия, а также команды, которые можно отдавать редактору в данном режиме.

*Режим работы с рисунком (рисование)*. В этом режиме на рабочем поле находится изображение инструмента. С помощью инструмента можно наносить рисунок, редактировать его, манипулировать его фрагментами.

*Режим выбора и настройки инструмента.* Курсор – указатель, находится в поле экрана с изображениями инструментов (меню инструментов). Кроме того, с помощью меню можно настроить инструмент на определенный тип и ширину линии, орнамент закраски.

*Режим выбора рабочих цветов*. Курсор находится в поле экрана с изображением цветовой палитры. В этом режиме можно установить цвет фона, цвет рисунка. Некоторые графические редакторы дают возможность пользователю изменять палитру.
*Режим работы с внешними устройствами*. В этом режиме можно выполнять команды записи рисунка на диск, считывания рисунка с диска, вывода рисунка на печать. Графические редакторы на профессиональных ПК могут работать со сканером, используя его для ввода изображения с репродукций.

Работая с графическим редактором, пользователь применяет не только клавиатуру, но и манипулятор-мышь. Создавая изображения на экране компьютера, можно не только рисовать их самому, но и использовать другие изображения, например, фотографии, рисунки из книг и т.д. Для ввода такой дополнительной графической информации в компьютере используется специальное устройство – сканер (Павлова и др., 2009).

Несмотря на то что редакторы растровой и векторной графики могут использовать одинаковые инструменты, способы представления создаваемых ими изображений различны. Прежде чем выполнить операцию над фрагментом изображения, его необходимо «выделить». В редакторах векторной графики выделяют объекты (векторные примитивы), а в редакторах растровой графики – области (наборы пикселей). Чтобы выделить объект, достаточно щелкнуть на нем мышью. Выделение же области – более сложная задача, так как в этом случае необходимо точно указать, какая группа пикселей составляет область. Вот почему в редакторах растровой графики встречаются разнообразные инструменты выделения. Некоторые из них используются для выделения областей простой формы (прямоугольников или эллипсов), другие – для выделения областей со сложной криволинейной границы. Так как основное понятие растровой графики – пиксель, большинство инструментов и команд редакторов растровой графики изменяют яркость и цветовые оттенки отдельных пикселей. Это дает возможность резкость изображений, осветлять или затемнять отдельные его улучшать фрагменты, а также удалять небольшие дефекты (царапины и т. д.). Основное понятие векторной графики – объект. Поэтому редакторы векторной графики содержат команды упорядочения, взаимного выравнивания, пересечения объектов, исключения одних объектов из других. Таким образом, можно создавать новые объекты сложной формы из более простых. Как правило, в редакторах растровой и

векторной графики имеются средства для получения эффекта объема. Однако трехмерные образы реальных объектов следует создавать в программах трехмерного моделирования (Залогова, 2005).

#### Векторные графические редакторы

Сфера применения векторной графики очень широка, она играет огромную роль в компьютерной полиграфии. Векторными методами формируются шрифты True Tupe. Векторные редакторы применяют для создания графических изображений высокой четкости и точности: чертежей, схем, диаграмм, фигурных заголовков, фирменных логотипов и стилей. С их помощью создают также штриховые рисунки.

Основные редакторы векторной графики: CorelDraw, Adobe Illustrator, FreeHand. Все эти редакторы работают с одними и теми же объектами векторной графики, основаны на одних и тех же принципах, имеют схожие инструменты, и, соответственно, приемы создания векторных изображений в этих редакторах удивительно похожи. Для работы с изображением каждый векторный редактор имеет панель инструментов и другие элементы управления. Инструменты панели управления служат для простейших операций с контурами. Векторные редакторы позволяют создавать новые конструкции символов с помощью инструментов для работы с контурами. В векторных редакторах есть средства применения эффектов к простым объектам (видтинювання, выдавливания, искажения, прозрачность и т.д.). Прочие элементы управления сосредоточены в строке меню и специальных диалоговых окнах. В векторном редакторе Adobe Illustrator, например, эти диалоговые окна называют палитрами. Векторное изображение можно строить вручную путем создания и объединения простейших контуров либо получать путем трассировки (векторизации) растровых изображений. Следует отметить наличие средств обработки текста, к которым можно применить все приемы редактирования объектов, управлять размерами текста и возвращать в любую сторону. Текстовые объекты в векторных редакторах рассматриваются как объекты особой породы. В арсенале векторных программ есть множество инструментов для выполнения

различных задач как в традиционных операционных средах, так и в работе с Интернетом.

Редактор *CorelDRAW*. На сегодняшний день это наиболее распространенный редактор векторной графики. Помимо обработки векторной графики, в этом пакете существует обработчик растровой графики (Corel PHOTO-PAINT), трассировщик изображений (CorelTRACE), редактор шрифтов, подготовки текстур и создания штрихкодов, а также огромные коллекции с изображениями (CorelGallery). Сайт программы: www.corel.com.

Редактор Adobe Illustrator. Ветеран векторной графики – пакет Illustrator, когда-то представлял собой выдающееся достижение в этой области и послужил моделью, которая легла в основу многих программ. Adobe Illustrator по своей Adobe Photoshop: эквивалентен растровому редактору мощности имеет аналогичный интерфейс, позволяет подключать различные фильтры и эффекты, многие графические форматы. Сайт понимает программы: www.adobe.com/products/illustrator.

Редактор *Macromedia FreeHand*. Был разработан фирмой Aldus, позднее права перекупила Macromedia. Пакет обеспечивает высокое удобство работы с кистями, эффекты выдавливания и трехмерное вращение векторных объектов, может работать в многостраничном режиме. Сайт программы: www.macromedia.com/software/freehand.

Редактор *XFig.* Этот графический редактор предназначен для работы в операционной системе Linux и встроен в графическую оболочку KDE. Стандартный набор геометрических фигур включает в себя: окружность и эллипс с возможностями их создания по двум различным начальным условиям – радиусу или диаметру, замкнутые и незамкнутые линии и ломаные, правильные и неправильные многоугольники, дуги и различные прямоугольники.

Профессиональные дизайнеры используют для создания векторной графики инструменты, проверенные годами, например, Adobe Illustrator и CorelDraw. Но существует множество бесплатных и онлайновых инструментов. С их помощью

можно перевести растровое изображение в векторное, сделать набросок несложного дизайна, сгенерировать узор гильоша и так далее (Бондаренко, 2013).

«*Рисунки Google»* – самый простой векторный редактор, который всегда под рукой. Самый известный, простой и доступный онлайновый векторный редактор – это «Рисунки» в сервисе «Google Диск». При помощи веб-приложения «Рисунки» можно создавать различные фигуры, стрелки, выноски, графические элементы формул, добавлять прямые линии, кривые, ломаные, стрелки и каракули. Также доступно добавление текстовых элементов, импорт растровых изображений. Векторные рисунки импортировать можно, но редактировать — нет (доступен только просмотр). Сайт программы: https://www.google.com/intl/ru/drive/

*Autotracer* – бесплатный конвертер растровых картинок в векторные. С его помощью можно преобразовать файлы PNG, BMP, JPEG и GIF в векторный формат и сохранить в формат SVG, EPS, PDF, DXF. Сервис полностью бесплатен, но есть одно ограничение: размер исходного файла не должен превышать один мегабайт. Перед преобразованием можно задать некоторые настройки: определиться с числом цветов, которые будут оставлены на растровом изображении перед векторизацией, а также избавиться от зашумленности, указать, в каких случаях линии будут приняты за кривые. Сайт программы: www.autotracer.org/

Vector Paint – векторный редактор в Chrome, который может работать и без браузера. Запускать такие приложения можно прямо с рабочего стола – открытый браузер, равно как и доступ к Интернету, для их использования не нужны. Для работы с изображениями и создания новых Vector Paint содержит все основные инструменты и режимы веркторного редактора. По умолчанию работа ведется на одном слое, но при необходимости можно добавить возможность работы с несколькими слоями. Эта возможность реализована в виде дополнения. В Vector Paint предусмотрены и другие дополнения, которые могут быть с легкостью добавлены в приложение. Это расширенные возможности для работы с текстом, добавление маркеров, использование изображений для создания повторяющихся рисунков на формах. Готовые проекты сохраняются в формате SVG, а также могут быть экспортированы в PNG и JPEG. Сайт программы: http://vectorpaint.yaks.co.nz/

Inkscape – векторный редактор имеющий массу достоинств. Во-первых, он кроссплатформенный (может запускаться в среде Windows, Linux, Mac). Во-вторых, поддерживает большое число популярных векторных форматов, в числе которых SVG, SVGZ, EMF, DXF, EPS, PostScript, WMF и др. В редакторе есть и стандартные для любого векторного редактора средства для создания графики, он дает выполнять различные действия с контурами, в распоряжении возможность пользователя большая библиотека фильтров и специальный редактор, при помощи которого можно создавать собственные эффекты. В программе реализована поддержка графических планшетов, благодаря чему в ней можно вручную рисовать, управляя наклоном и силой нажатия пера. Inkscape – полноценная альтернатива коммерческим пакетам. Сайт программы: https://inkscape.org/ru/. Существуют и различные on-line векторные редакторы разной степени сложности и содержания: SVG-Edit (https://svg-edit.ru.), Chittram (www.chittram.com), ZebraTRACE бесплатное создание гильошей (http://maxim-s-barabash.github.io/ZebraTrace/).

#### ФОРМАТЫ ГРАФИЧЕСКИХ ФАЙЛОВ

#### Стандартные и собственные форматы графических редакторов

Проблема сохранения изображений для последующей их обработки чрезвычайно важна. С ней сталкиваются пользователи любых графических систем. форматы обеспечивают различный уровень качества Разные сохраняемых растровых графических изображений в файлах разного размера. Это обуславливает необходимость корректного подбора используемых форматов файлов. Формат графического файла – способ представления графических данных на внешнем носителе, т.е. определяет способ хранения информации в файле (растровый или векторный), а также форму хранения информации (используемый алгоритм сжатия). Некоторые форматы графических файлов являются стандартными (универсальными), так как могут быть обработаны большинством графических редакторов. Некоторые программы обработки изображений используют собственные форматы (оригинальные), которые распознаются только самой

создающей программой. Преимущество оригинальных форматов файлов состоит в том, что они позволяют сохранять изображения при меньшем размере файла.

Изображение может быть обработано несколькими программами прежде, чем примет свой окончательный вид и будет напечатано или появится в мультимедиа презентации. Как правило, изображение «кочует» из программы в программу, где на ходу корректируется, улучшается, к нему добавляются спецэффекты. Таким образом, пересылка изображения из одной программы в другую жизненно необходима, так как каждая графическая программа имеет свои сильные и слабые стороны. Чтобы сделать более быстрым и эффективным этот процесс, разработчики программных средств создали ряд компьютерных форматов, которые служат своего рода общим языком, и для этого используются стандартные форматы графических файлов. Собственные форматы файлов используются, как правило, когда файл сохраняется в графическом приложении, мы запоминаем его в собственном файловом формате, т.е. формате, программном созданном для данного программного продукта. В каждом конкретном случае создатели программного продукта выбирают самый оптимальный вариант формата. Если создается файл, который будет экспортироваться в другие программы, возможно, понадобится сохранить файл в формате, отличном от того, в котором он был создан.

#### Форматы файлов векторной графики

Файлы векторных форматов содержат описания рисунков в виде набора команд для построения простейших графических объектов (линий, окружностей, прямоугольников, дуг и т.д.). Кроме того, в этих файлах хранится некоторая дополнительная информация. Разные векторные форматы различаются набором команд и способом их кодирования.

Формат PostScript (Adobe PostScript) – язык описания страниц (язык управления лазерными принтерами) фирмы Adobe. Файлы этого формата с расширением .ps или, реже, .prn содержат в себе сам документ (только то, что располагалось на страницах), все связанные файлы, использованные шрифты, а также другую информацию: цветоделение, дополнительные платы, полутоновый

растр для каждой платы, линиатуру растра и прочие данные для выводного устройства.

Формат EPS (Encapsulated PostScript) – упрощенный PostScript. Не может содержать в одном файле более одной страницы, не сохраняет ряд установок для принтера. Как и в файлы печати PostScript, в EPS записывают конечный вариант работы, хотя такие программы, как Adobe Illustrator, Photoshop и Macromedia FreeHand, могут использовать его как рабочий. EPS используется для передачи векторов и растра в издательские системы, создается почти всеми программами, работающими с графикой. Изначально EPS разрабатывался как векторный формат, позднее появилась его растровая разновидность – Photoshop EPS.

Формат PDF (Portable Document Format) предложен фирмой Adobe как независимый от платформы формат, в котором могут быть сохранены иллюстрации (векторные и растровые) и текст, причем с множеством шрифтов и гипертекстовых ссылок. Первоначальная задача PDF – передача по сети в сжатом виде проиллюстрированных и отформатированных документов – сегодня значительно расширена.

Формат AI (Adobe Illustrator Document) – может содержать в одном файле только одну страницу, имеет маленькое рабочее поле, всего 3х3 метра. В целом несколько уступает FreeHand и CorelDRAW по иллюстративным возможностям, тем не менее отличается наибольшей стабильностью и совместимостью с PostScript. AI поддерживают почти все программы, так или иначе связанные с векторной графикой. Формат Illustrator является наилучшим посредником при передаче векторов из одной программы в другую, с PC на Macintosh и назад.

Формат FH (FreeHand Document) понимает только сам FreeHand, Illustrator для Macintosh и несколько программ от Macromedia. Последние версии имеют полную кроссплатформенную совместимость. Поддерживает многостраничность. Некоторые эффекты FreeHand несовместимы с PostScript.

Формат CDR (CorelDRAW Document) – собственный формат CorelDRAW, пользоваться им чрезвычайно удобно, он имеет неоспоримое лидерство на платформе PC. Многие программы на PC (FreeHand, Illustrator, PageMaker) могут

импортировать файлы CDR. Последние версии имеют полную кроссплатформенность. В файлах этих версий применяется компрессия для векторов и растра отдельно, могут внедряться шрифты, файлы CDR имеют огромное рабочее поле (45х45 метров); начиная с 4-й версии поддерживается многостраничность.

изображений Windows. В этом формате создают изображения для ОС либо для Интернета. Он подходит для хранения векторных и растровых файлов и их последующего вывода как на экраны мониторов, так и на печатающие устройства. то что формат WMF разработан для среды Windows. Несмотря на OH поддерживается графическими программами на многих других платформах и часто используется для обмена данными с Windows-приложениями. В случае векторной Encapsulated PostScript иллюстрации формат поддерживает использование графических элементов, сохраненных как WMF.

Формат CGM (Computer Graphics Metafile) – популярный векторный формат (хотя примененный здесь алгоритм позволяет записывать и растровые картинки), используемый для переноса данных на другие платформы. Теоретически основное преимущество CGM – его независимость от аппаратной и программной платформ. CGM больше подходит для обмена данными между программами и платформами, нежели для хранения чертежей и рисунков.

Формат DXF (Drawing eXchange Format) – векторный формат поддерживают все программы автоматизированного проектирования начиная с пакета AutoCAD компании Autodesk. Однако из-за его сложности некоторые приложения "умеют" только читать DXF-файлы и не способны сохранять данные в этом формате. В DXF реализованы многие возможности, отсутствующие в большинстве других форматов, например, хранение трехмерных объектов. Необходимо отметить наличие прекрасного встроенного кодировщика текста.

#### Форматы растровых графических файлов

Растровые форматы характеризуются тем, что все изображение по вертикали и горизонтали разбивается на достаточно маленькие прямоугольники – пиксели.

В файлах растровых форматов хранятся:

– размер изображения – количество видеопикселей по вертикали и горизонтали;

 – битовая глубина – число битов, используемых для хранения цвета одного видеопикселя;

 данные, описывающие рисунок (цвет каждого пикселя рисунка и его положение), и некоторая дополнительная информация.

В файле, содержащем растровую графику, хранятся данные о цвете каждого пикселя, составляющего изображение. Чем меньше прямоугольники, на которые разбивается изображение, и чем больше их число, тем выше разрешение графического файла и тем выше его качество. Размер изображения, хранящегося в файле, задается в виде числа пикселей по горизонтали и вертикали.

Важной является также информация о количестве цветов, закодированных в файле. Цвет каждого пикселя определяется числом битов – элементарных единиц информации, с которыми может иметь дело компьютер. Каждый бит может принимать два значения – - 1 или 0. В зависимости от того, сколько битов отведено для цвета каждого пикселя, возможно кодирование различного числа цветов. Если для кодировки отвести один бит, то каждый пиксель может быть либо белым (значение 1), либо черным (значение 0). Такое изображение называют монохромным. Таким образом, кодирование цветов для изображения может выглядеть следующим образом:

4 бита информации –  $2^4 = 16$  цветов, с комбинациями битов от 0000 до 1111,

8 битов информации – 2<sup>8</sup> = 256 цветов, с комбинациями от 00000000 до 11111111,

16 битов информации –  $2^{16} = 65536$  цветов (*High Color*),

24 бита информации  $-2^{24} = 16777216$  цветов (*True Color*).

Однако следует понимать, что даже если вы используете True Color для отрисовки изображений, реального количества цветов может быть значительно меньше, поэтому не стоит забывать про технические возможности экранов.

Чем больше расширение изображения и количество цветов, содержащихся в нем, тем выше будет его вес. Например, если размер изображения 1766 x 1528, а количество цветов 16 777 216, то объем такого растрового файла будет порядка 8 МБ. Для разрешения проблемы хранения растровых файлов используют специальные процедуры, уменьшающие вес исходного файла, – сжатие. Методы сжатия файлов можно разделить на два типа:

- сжатие файла с помощью программы архиватора;
- сжатие, алгоритм которого заключен в формат файла.

В первом варианте сжатие обеспечивается наличием отдельной программы (например, WinRAR), которая благодаря своим собственным алгоритмам изменяет исходный файл, архивирует его и создает новый. Оптимизация веса файла в итоге может быть крайне значительной. Однако данный метод сжатия применим лишь для долгосрочного хранения файлов либо их переправки, так как для работы с заархивированными файлами потребуется провести процедуру разархивации файла до исходного состояния.

Если же алгоритм сжатия включен в формат файла, то соответствующие программы чтения правильно интерпретируют сжатые данные и позволяют использовать сжатый файл для постоянной работы, однако оптимизация файлового веса в данном случае будет значительно ниже, чем при первом варианте.

Перейдем к рассмотрению основных форматов растровой графики:

ВМР – в ЭТОМ формате первоначально использовалось простейшее кодирование – по пикселям (самое неэкономное), которые обходились последовательно по строкам начиная с нижнего левого угла графического изображения. Файлы этого формата входили в первые версии Windows. В этом формате использовалось только 256 цветов, т.е. пиксель представляется только одним байтом. В дальнейшем формат стал использоваться и для сохранения полноцветных изображений. Формат ВМР – один за стандартных форматов растровой графики.

*TIFF* – стандартный формат в топографической графике и издательских системах. Файлы в формате TIFF обеспечивают лучшее качество печати. Формат TIFF относится к числу наиболее универсальных и распространенных форматов растровой графики. Он создавался в качестве межплатформенного универсального формата для цветных изображений. Работа с ним поддерживается почти всеми программами для работы с точечной графикой. Может хранить графику в монохромном виде, в RGB- и CMYK-цветовых представлениях.

*GIF* – формат обмена графическими данными, который служит для записи и хранения растровых графических изображений. Этот формат отличается от других форматов растровой графики тем, что он долгое время поддерживается в Интернете. Использует индексированные цвета (ограниченный набор цветов). Лучше всего отображаются рисунки, чертежи и изображения с небольшим количеством однородных цветов, прозрачные изображения и анимационные последовательности. В GIF-изображениях используется сжатие без потери информации.

JPEG \_ предназначен для хранения изображений co сжатием. Применяющийся в нем метод сжатия изображений разработан группой экспертов в области фотографии. JPEG – один из самых мощных алгоритмов сжатия изображения. Он является практически стандартом де-факто для хранения графики полноцветных изображений. Сохраняет параметры В цветовом представлении RGB (как правило).

*JPEG 2000* – данный формат был разработан для замены JPEG. При сохранении изображения с одинаковым уровнем сжатия изображения, сохраненные в формате JPEG 2000, получаются более четкими и занимают меньше места на диске. К тому же в этом формате решена проблема появления дефектов JPEG, которые наблюдались при сохранении с большим коэффициентом сжатия.

*PNG* – формат хранения растровой графики, использующий сжатие без потерь. Это очень мощный и широко применяемый формат в Интернете и других областях компьютерной графики.

*WMF* – используется для обмена графическими данными между приложениями OC Microsoft Windows. В WMF-файлах могут храниться как векторные, так и растровые изображения.

*PSD* – внутренний формат для пакета Adobe Photoshop. Позволяет сохранять слои в изображении и поддерживает все типы графики.

Фотографические изображения, полученные с помощью устройств с высокой разрешающей способностью, часто занимают большой объем памяти. Решением проблемы хранения больших растровых изображений является сжатие, т.е. уменьшение размера файла за счет изменения способа организации данных. Методы сжатия делятся на две категории:

- сжатие файла с помощью программ-архиваторов;

- сжатие, алгоритм которого включен в формат файла.

Одним из простейших способов сжатия является метод *RLE (Run Length Encodin* – кодирование с переменной длиной строки). Действие метода RLE заключается в поиске одинаковых пикселей в одной строке. Найденные цепочки одинаковых элементов заменяются на пары (счетчик повторений, значение), что в определенных случаях существенно уменьшает избыточность данных. Алгоритм в первую очередь рассчитан на изображения с большими областями повторяющегося цвета (деловая графика, схемы, рисунки и т.п.). Недостатком такого подхода является то, что в определенных ситуациях он может вместо уменьшения приводить к увеличению размера файла (например, в некоторых случаях при сохранении цветных фотографий).

Метод сжатия LZW (Lempel-Ziv-Welch) – сжимаются данные путем поиска одинаковых последовательностей (они называются фразы) во всем файле. Выявленные последовательности сохраняются в таблице, им присваиваются более короткие маркеры (ключи). Таким образом, вместо значения цвета пикселя можно использовать индекс из таблицы. Наиболее часто встречающиеся цвета на изображении имеют меньшие индексы, а редко встречающиеся цвета размещаются в конце таблицы. Метод LZW, так же, как и RLE, лучше действует на участках однородных, свободных от шума цветов, он действует гораздо лучше, чем RLE, при

сжатии произвольных графических данных, но процесс кодирования и распаковки происходит медленнее.

*Метод сжатия Хаффмана (Huffman)*. В методе Хаффмана берется набор символов, который анализируется, чтобы определить частоту каждого символа. Затем для наиболее часто встречающихся символов используется представление в виде минимально возможного количества битов.

*Memod JPEG (Joint Photographers Experts Group)* основан на частотных представлениях изображения. В частотном представлении основную долю информации несут низкие частоты. Высокие частоты описывают шум и несущественные детали. Для сжатого представления сигнала могут использоваться различные преобразования. В результате преобразования получаем спектр. При этом низкие частоты содержатся в верхнем левом углу спектра, а высокие - в правом нижнем. Высокие частоты можно обнулить и не хранить. Используется методика сжатия с потерями. Хранится не информация о цвете пикселей, а коэффициенты разложения по некоторому базису. Положительными сторонами алгоритма JPEG является то, что пользователь может управлять соотношением размер/качество, задавая степень сжатия. С помощью алгоритма JPEG достигаются большие коэффициенты сжатия при визуально высоком качестве изображения. Отрицательными сторонами алгоритма является то, что при повышении степени сжатия изображение распадается на отдельные квадратные области. Кроме того, поскольку это алгоритм сжатия с потерями качества, то изображения, обработанные с его применением, практически не используются для анализа и дальнейшей работы.

Основные форматы растровых графических файлов

*Bit MaP image (BMP)* – универсальный формат растровых графических файлов, используется в операционной системе Windows.

*Tagged Image File Format (TIFF)* – формат растровых графических файлов, поддерживается всеми основными графическими редакторами. Включает в себя алгоритм сжатия без потерь информации (LZW).

Graphics Interchange Format (GIF) – формат растровых графических файлов. Включает алгоритм сжатия без потерь информации (LZW). Рекомендуется для хранения изображений с ограниченным количеством цветов. Используется для размещения графических изображений на Web-страницах в Интернете.

*Portable Network Graphic (PNG)* – формат растровых графических файлов, аналогичный формату GIF. Рекомендуется для размещения графических изображений на Web-страницах в Интернете.

Joint Photographic Expert Group (JPEG) – формат растровых графических файлов, который реализует эффективный алгоритм сжатия (метод JPEG) для отсканированных фотографий и иллюстраций. Алгоритм сжатия позволяет уменьшить объем файла в десятки раз, однако приводит к необратимой потере части информации. Используется для размещения графических изображений на Web-страницах в Интернете.

## Глава 2. Программирование для представления и обработки биологической и экологической информации

#### Инструменты статистического анализа

Как правило, в биологии и экологии под обработкой данных в первую очередь подразумевают статистический анализ. Удобнее всего проводить статистический анализ с помощью специализированных инструментов, к которым, кстати, не относится MS Excel, широко применяемый в этой области, причем не только студентами и школьниками. Современный статистический анализ использует сложный математический аппарат и требует гибких способов визуализации. Таблицы Excel «похвастаться» этим не могут, хотя их вполне можно применять для простейших операций с небольшими массивами данных. Удобно их использовать и для организации данных в единую таблицу.

Специализированные статистические программы можно разделить на две группы: оконно-кнопочные системы и статистические среды. Первые почти не

отличаются внешне от электронных таблиц, однако снабжены значительно большим арсеналом доступных статистических приемов. Кроме того, у них традиционно мощная графическая часть (возможных графиков больше, И управление ими более гибкое), а часто и подсистема подготовки отчетов. Многие такие программы имеют значительно меньше ограничений, чем электронные таблицы. Очень распространена в России относящаяся к этой группе система STATISTICA. Как замечено выше, она отличается мощной графической частью, т.е. имеет множество возможных вариантов графического вывода, которые при этом довольно гибко настраиваются. При этом данная система не лишена минусов, наиболее существенными которых являются ограниченный среди регион использования (преимущественно Россия) и закрытые алгоритмы вычисления (т.е. пользователю непонятно, что на самом деле в каждом конкретном случае делает программа). К тому же к системе в свое время было немало претензий по поводу «быстрых и грязных» алгоритмов работы, и есть подозрение, что ситуация не слишком изменилась.

Существуют и другие подобные системы: STADIA, PAST, SPSS, MiniTab и StarGraphics. Часть из них – коммерческие продукты, но есть и бесплатные. Также им свойственна закрытость алгоритмов и ограниченность распространения. Статистические среды используют в основном интерфейс командной строки. Пользователь вводит команды, система на них отвечает. Звучит это просто, однако сами эти программы в числе самых сложных систем обработки. Для работы с подобными системами нужны некоторые навыки программиста, зато пользователь получает полный контроль над системой: он может комбинировать любые типы анализа, записывать процедуры в скрипты, которые можно запустить в любое время, модифицировать вывод графиков, сохранять их в любые графические форматы, легко писать расширения для системы, а если она к тому же имеет открытый код, то и модифицировать систему (Шипунов, 2014).

Именно к таким открытым системам относится язык программирования R. R – язык программирования, созданный специально для научных вычислений и анализа данных с упором на визуализацию и воспроизводимость.

#### Язык программирования R

Создатели R – Росс Ихака и Роберт Джентльмэн. R появился в 1993 г. и практически сразу стал популярным по двум причинам: а) распространялся бесплатно, так как был разработан в рамках проекта GNU, и это выгодно отличало его от близкого языка S (в принципе оба языка настолько схожи, что код, написанный на S, может быть практически без изменений использован в R); б) благодаря обладал способностью развиваться использованию пакетов специальных библиотек дополнительных функций, которые предназначены для работы с данными в какой-либо узкой области. За более чем 20-летнюю историю развития R не только не устарел, но и, наоборот, расширил свои возможности и очень широкое распространение в разных сферах человеческой получил деятельности. В частности, в биологии и экологии R представляет собой универсальный инструмент статистической обработки и специфического анализа данных, включая такие области, как эволюция и филогенетика, моделирование экосистем, биоинформатика, генетика и нейробиология.

R – это многофункциональный язык. Он использует большое количество функций статистического анализа и графических функций. Большим преимуществом R является то, что графика воспроизводится в отдельном окне и может быть сохранена в различных форматах, в том числе таких распространенных, как jpg, png, bmp, pdf и др.

R позволяет анализировать сразу несколько массивов данных и комбинировать различные статистические функции.

R – это свободное кросс-платформенное программное обеспечение с открытым исходным кодом. Есть варианты для Unix и Linux, для Window и Macintosh. Исходные файлы для установки нужной версии R можно бесплатно скачать на сайте the Comprehensive R Archive Network (CRAN) вместе с инструкциями по установке (https://cran.r-project.org/).

Этот язык обладает большим количеством плюсов, хотя минусы тоже есть. Плюсы:

• Он бесплатен, его код открыт.

• Содержит огромное количество встроенных статистических тестов, математических алгоритмов, функций для анализа данных в академических и отраслевых исследованиях.

• Расширяемость: модульная система пакетов (*packages*).

• Обширное сообщество разработчиков.

• Полностью программируемая высокоуровневая графика.

• Кросс-платформенность, воспроизводимость результатов.

• R становится популярным: статьи, книги, конференции, юзергруппы.

• Обратная совместимость – код, написанный 10 лет назад, должен работать и сейчас.

## Минусы:

• R достаточно сложен как язык программирования.

• Документация некоторых методов труднодоступна без глубокого знания математических методов и статистических процедур.

• Легко написать плохой (медленный, неподдерживаемый, нечитаемый) код.

• Пакеты дополнений устаревают.

• Можно заметить некоторую непоследовательность в именовании переменных и реализации методов.

Работа с R не требует посредничества какого-либо графического пользовательского интерфейса (ГИП), и чаще всего те, кто программирует на R, ГИП не пользуются. Тем не менее существуют специально разработанные для работы с R инструменты, большинство из которых находится в свободном доступе.

Одной из наиболее популярных сред для программирования на R является RStudio. Во многом подобны ей StatET и ESS.

- RStudio http://www.rstudio.org/
- StatET http://www.walware.de/goto/statet/
- ESS (Emacs Speaks Statistics) http://ess.r-project.org/

• JGR (Java GUI for R) – http://cran.rproject.org/web/packages/JGR/index.html

Существуют и коммерческие продукты для работы с R, например, Revolution Analytics, an R service company – http://www.revolutionanalytics.com/.

R – мультипарадигменный язык, сочетающий в себе функциональное, процедурное, объектно-ориентированное и рефлексивное программирование. В нем можно использовать набор встроенных функций и переменных, а также создавать свои и комбинировать их в любые сочетания. Это придает R невероятную гибкость при профессиональном использовании.

#### Другие языки программирования

Программное обеспечение R не единственное в своём роде, и если мы говорим о больших массивах биологических данных, то непременно стоит упомянуть такой проект, как *Biopython*.

Сразу следует внести небольшое уточнение. Biopython ЭТО не самостоятельный язык программирования, a так называемая коллекция некоммерческих инструментов для вычислений В области биологии И биоинформатики, относящаяся к языку программирования Python. Возможно, вы когда-то уже слышали о Python и знаете, что это высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчиков и обладающий широким объемом полезных функций стандартных библиотек и минималистичным синтаксисом. Biopython же, в свою очередь, является полезной надстройкой языка Python, позволяющей направить вычислительную мощь языка как раз на биологические данные.

Разработка Biopython началась в 1999 г., а впервые проект увидел свет в июле 2000 г. Одними из первых разработчиков, занимающихся вопросом создания Biopython, были Джефф Чанг, Эндрю Далке и Брэд Чепмен, хотя суммарно, так или иначе, к процессу создания Biopython на сегодняшний день причастны более 100 человек. Первоначально Biopython включал в себя возможность доступа,

индексирования и обработки файлов биологической последовательности. Это все еще является основным направлением, но постепенно были добавлены модули, расширяющие стандартный функционал, чтобы охватить дополнительные области биологии. Начиная с версии 1.62 Biopython поддерживает работу на Python 3, а также Python 2. Вкратце коснемся его основных возможностей.

### Работа с большим количеством форматов

Віоруthon способен читать и писать наиболее распространенные форматы файлов для каждой из своих функциональных областей, а его лицензия разрешима и совместима с большинством других лицензий на программное обеспечение, что позволяет использовать Biopython в различных программных проектах.

#### Доступ к онлайн-базам данных

Через модуль Bio.Entrez пользователи Biopython могут загружать биологические данные из баз данных NCBI. Каждая из функций, предоставляемых поисковой системой Entrez, доступна через функции в этом модуле, включая поиск и загрузку записей.

#### Работа с филогенией

Модуль Bio.Phylo предоставляет инструменты для работы и визуализации филогенетических деревьев. Для чтения и записи поддерживаются различные форматы файлов, включая Newick, NEXUS и phyloXML. Общие древовидные манипуляции и обходы поддерживаются через объекты Tree и Clade. Примеры включают в себя преобразование и сортировку древовидных файлов, извлечение подмножеств из дерева, изменение корня дерева и анализ особенностей ветвей, таких как длина или оценка.

#### Графические диаграммы

Модуль GenomeDiagram предоставляет методы визуализации последовательностей в Biopython. Последовательности могут быть представлены в линейной или круговой форме, а также поддерживают многие форматы вывода, включая PDF и PNG. Диаграммы создаются путем создания дорожек, а затем добавляются функции последовательности к трекам. Перейдя по функциям последовательности и используя их атрибуты, чтобы решить, как они добавляются

к трекам диаграммы, можно контролировать внешний вид финальной диаграммы. Перекрестные ссылки могут быть проведены между различными дорожками, что позволяет сравнивать несколько последовательностей на одной диаграмме.

#### Макромолекулярная структура

Модуль Bio.PDB может загружать молекулярные структуры из файлов PDB и mmCIF и был добавлен в Biopython в 2003 г. Объект Structure является центральным в этом модуле и организует иерархию макромолекулярной структуры. Используя Bio.PDB, можно перемещаться по отдельным компонентам файла макромолекулярной структуры, например, изучать каждый атом в белке. Можно проводить общие анализы, такие как измерение расстояний или углов, сравнение остатков и расчет глубины остатков.

#### Генетика популяций

Модуль Bio.PopGen добавляет поддержку Biopython для Genepop, программного пакета для статистического анализа популяционной генетики. Это позволяет анализировать равновесие Харди-Вайнберга, неравновесность сцепления и другие особенности частот аллелей популяции.

Рабочие возможности проекта Biopython огромны, и если вас заинтересовал данный проект, вы всегда можете изучить его документацию, которая находится в свободном доступе и открыта для всех желающих. Логичным было бы проведение небольшого сравнения R и Biopython (так как Biopython – это лишь надстройка языка Python, фактически сравнение будет произведено с ним). Мы коснемся лишь самых поверхностных аспектов, без углубления в сложности и изыски программирования. Если вас это заинтересует, вы всегда можете сделать это самостоятельно.

#### Синтаксис и структура

Бытует мнение, что R по своей структуре и синтаксису сложнее, чем Python, так как не обладает определенным минимализмом. Однако подобное мнение зависит от навыков пользователя. Несомненно, для программистов Python покажется более логичным и простым, но среди рядовых пользователей распространено мнение, что язык R более специализирован под решение именно

статистических задач, способен простить многие ошибки непрофессиональных пользователей и отличается большей, так скажем, «человечностью». Ориентация языка R на векторы и матрицы в какой-то степени сближает его с пользованием Excel.

## Установка дополнительных пакетов

В Руthon ввиду наличия нескольких версий – 2. и 3. – могут возникнуть ошибки совместимости и зачастую потребуется установка каких-либо дополнительных пакетов или библиотек (Biopython точно так же требует установки разных библиотек для разных задач), что является достаточно трудоемким процессом. В R установка пакета или библиотеки контролируется пропиской одной (в редком случае нескольких) строки кода. Следует учитывать также, что все зависимости он подгружает самостоятельно.

#### Экосистема языка

R является языком программирования с довольно большим сроком жизни. Прибавляя к этому узкую специализацию, бесплатность, ориентацию по большей части на непрограммистов, мы получаем фактически колоссальную экосистему: stackoverflow, масса форумов и блогов, огромное количество учебников и учебных курсов, в том числе онлайн, но самое главное – в R реализовано все, что существует в data science, и даже сверх того. Biopython также является относительно немолодым языком, но ввиду того, что это лишь надстройка, он не имеет настолько сильно развитой экосистемы, как R. К тому же следует упомянуть, что вся документация по Biopython существует на английском языке, что также может стать для кого-то препятствием.

#### Направленность и сфера деятельности

Как уже говорилось, язык программирования R чаще используется как инструмент статистической обработки. Biopython также по большей части предназначен для решения тех или иных задач статистических исследований больших массивов данных, но принципы работы с Biopython точно такие же, как и с Python, а он уже является современным и полноценным языком программирования,

освоение которого позволит решать задачи, связанные не только с областью статистики.

#### Скорость работы

R ввиду своей специализации неэффективен в вопросах работы с памятью и оптимизации скорости выполнения задач. Он эффективно работает с векторами и матрицами, однако менее эффективен в циклах, нежели Python. Для пользователя, не искушенного в вопросах программирования, зачастую отсутствует потребность работать с циклами и т.п., а вполне достаточно алгоритмов работы с векторами.

#### Визуализация данных

Язык программирования R заточен под визуализацию данных, библиотеки построения графиков и диаграмм являются стандартными для него, Python же (и Biopython, в частности) в своей стандартной комплектации не создан для вопросов визуализации.

Таким образом, подведем небольшой итог. Если вы не собираетесь выходить за пределы анализа данных, то ваш выбор – язык программирования R. Если же вы хотите создавать приложения или модули, которые будут регулярно анализировать данные и формировать зависимости, - Python ваш выбор. Кроме того, нельзя не упомянуть о RPy – интерфейсе, позволяющем использовать преимущества обоих языков в своих областях. RPy – это модуль Python, который обеспечивает доступ к R из среды Python. Модуль доступен по адресу http://rpy.sourceforge.net. Краткий обучающий его использованию курс ПО доступен по адресу: http://www.daimi.au.dk/~besen/TBiB2007/lecture-notes/rpy.html (презентация на английском языке "RPy – R from Python»).

## Раздел 2. ИСПОЛЬЗОВАНИЕ КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ В ЭКОЛОГИИ И ПРИРОДОПОЛЬЗОВАНИИ

#### Глава 3. Графические редакторы

#### РЕДАКТОР ВЕКТОРНОЙ ГРАФИКИ CORELDRAW

Около двадцати лет назад канадская компания Corel выпустила в мир новый продукт, который назывался CorelDRAW. На сегодняшний день CorelDRAW является полноценным многофункциональным редактором векторной и растровой графики. Новые версии программы выпускаются исключительно под OC Windows. CorelDRAW – это программный комплекс, который включает в себя:

• CorelDRAW – редактор векторной графики;

• CorelPhotoPain – редактор растровой графики;

• CorelCapture – программу для захвата изображения с экрана компьютера;

•CorelTrace – программу для перевода растрового изображения в векторное.

Основными возможностями графического редактора CorelDRAW являются:

 Рисование векторной графики. В этой области можно создавать различные объекты, конструировать и оформлять иллюстрации. Также она отлично подходит для создания чертежей и работы с ними.

 Текст и работа с ним. Программа позволяет работать с разными видами текста, придумывать новые эксклюзивные шрифты. Например, без проблем создается векторный и объемный текст.

– Работа с фотографиями. Новые возможности работы с фото стали доступны в последней версии CorelDRAW. Также присущи инструменты для обработки изображений в формате гаw. Есть возможность создавать

анимационные gif-изображения и корректировать их. Есть функция поиска графики и изображений как на компьютере пользователя, так и в Интернете.

 Другие возможности, в которые входит прорисовка контуров, трассировка, рисование узоров, орнаментов и многое другое.

– Несмотря на то что CorelDRAW – это многофункциональный редактор с расширенными возможностями, он все равно продолжает совершенствоваться и развиваться. В каждой новой версии программы технологи устраняют ошибки предыдущих, добавляют новые функции, фильтры и возможности.

### Описание рабочего окна CorelDRAW

Программа CorelDRAW имеет стандартный оконный интерфейс (рис. 14).

#### Основные понятия:

Панель инструментов – панель, на которой размещены пиктограммы всех инструментов для создания и редактирования изображений.

Панель атрибутов (свойств) – панель, на которой отображаются свойства выбранного инструмента.

Палитра цветов – набор цветов для закраски рисунков.

*Рабочий стол* – рабочее пространство в центре окна, на котором располагается *рабочий лист* или *печатная страница* – ограничивающая рабочую площадь для создания и коррекции изображения, соответствующую формату выводимого на печать документа.

В *строке меню* находится набор команд для создания и преобразования изображений. Для удобства все команды разделены на группы. Каждое меню отвечает за выполнение команд отдельной группы.

#### Назначение пунктов главного меню:

Файл содержит команды ввода/выводя изображений.

*Правка* содержит команды редактирования изображений (копирование, удаление, размножение и др.).

Вид содержит команды настройки экрана.



Рис. 14. Рабочее окно CorelDraw

*Макет* содержит команды настройки рабочего листа, на котором создаются рисунки.

*Упорядочить* содержит команды упорядочения объектов, объединения нескольких объектов в один, разъединения объектов и др.

Эффекты содержит команды, реализующие различные графические эффекты.

*Растровые изображения* содержит команды для работы с растровыми изображениями.

Текст содержит команды редактирования текста.

Инструменты содержит команды настройки интерфейса пользователя.

*Окно* позволяет открыть одновременно несколько окон с рисунками и переключаться между ними в процессе работы. Кроме того, используя команды этого меню, пользователь может прятать или делать видимыми различные панели.

Помощь предназначена для вызова встроенной справочной системы.

Под строкой меню расположена *стандартная панель инструментов*. В ее состав вынесены кнопки, которые позволяют быстро выполнять соответствующие команды меню. В центре окна на рабочем столе располагается *рабочий лист* 

*(печатная страница),* выделенный тенью. Пользователь может устанавливать ориентацию рабочего листа (горизонтальная или вертикальная) и его размеры соответственно формату бумаги.

В левой части экрана располагается *панель инструментов*. На ней располагаются пиктограммы основного *набора инструментов*:

- Инструмент Указатель. Используется для выделения объектов перед их преобразованием (закраской, вращением, перемещением и др.). Выделенный объект всегда окружен маркировочной рамкой.
- Инструмент Форма. Используется для изменения формы объекта. В простейшем случае этот инструмент позволяет скруглить углы прямоугольника, а также получить дугу и сектор из эллипса.
- Инструмент Обрезка. Позволяет выбрать режим исключения или удаления частей объекта.
- Инструмент *Масштаб.* Используется для масштабирования изображения.
  - Инструмент *Свободная форма*. Используется для рисования линий. В дополнительном меню выбирается вид линии (прямая, кривая, Безье или художественная и т.д.).

Инструменты *Прямоугольник, Эллипс, Многоугольник* предназначены для создания различных видов соответствующих геометрических фигур. Дополнительные инструменты позволяют выбрать вид и отображение фигуры.

Инструмент Текст. Применяется для создания текстовых объектов.

Инструмент *Абрис*. Используется для управления свойствами контуров объекта. Дополнительно выбирается вид, цвет и размерные характеристики.

Инструмент Заливка. Применяется для заполнения объекта. Дополнительно можно выбрать вид заливки и заполнения.

Кроме этого, к пиктограмме основного инструмента привязано раскрывающееся меню *дополнительных инструментов* (вызов происходит при нажатии на пиктограмме инструмента, где есть треугольник). Такая организация уменьшает объем площади, занимаемой панелью инструментов (рис. 15).



Рис. 15. Группа дополнительных инструментов

Под стандартной панелью обычно находится Панель свойств. В отличие от других панелей, состав панели свойств является контекстно-зависимым. Это значит, что элементы этой панели определяются как используемым инструментом, так и объектом, над которым производятся действия. Вся необходимая информация задается на панели свойств. Палитра цветов, расположенная в правой части экрана, позволяет легко менять цвет объекта. Стандартные палитры, как правило, состоят из большого количества цветов, которые невозможно одновременно показать на экране. Поэтому просмотр цветом осуществляется по принципу работы с линейкой прокрутки. В строке состояния, которая находится в нижней части экрана, выводится некоторая полезная информация: координаты текущего положения курсора, какой объект выделен, цвет этого объекта и др. Пристыковываемые окна предназначены для быстрой работы с пунктами главного меню и редактирования объектов.

# Практическая работа №1. Основные приемы работы в редакторе CorelDRAW

## 1. Параметры страницы, масштабирование

Пользователь может устанавливать ориентацию рабочего листа (горизонтальная или вертикальная) и его размеры соответственно формату бумаги. Некоторые форматы заданы в CorelDRAW как стандартные. Например, A4 – 210х297 мм, A6 – 148х105 мм. Необходимо помнить, что размер рисунка, который мы видим на экране, не совпадает с его размером на печатной странице. По умолчанию рабочий лист соответствует формату бумаги A4. В этом случае рисунок на экране будет меньше, чем при печати. Если нужно создать визитную карточку размером 80х50 мм, то необходимо изменить размер рабочего листа (рис. 16).

## Изменить параметры станицы: задать формат и размер страницы

1. В панели Меню выбрать Макет/Параметры страницы.

2. Установить формат бумаги «Визитная карточка» и задать размер страницы 50х100 мм.

3. Изменить ориентацию страницы на альбомную.

4. С помощью инструмента Масштаб изменить масштаб страницы.

Параметры				×	
• Рабочее пространство	Размер				
<ul> <li>Документ</li> <li>Общие</li> </ul>	<ul> <li>Обычная бумага</li> <li>Наклейки</li> </ul>				
Страница Размер		🔘 Книжная 🔘 Альбомная			
Макет Метка	Бумага:	Визитная карточка 🔻			
Фон ⊕ Направляющие	Ширина:	91,0 миллиметры 🔻			
Сетка	Высота:	55,0 🙀 миллиметры			
Стили	Изменить размер только текущей страницы				
<ul> <li>Сохранить</li> <li>⊕ Публикация в Интер</li> </ul>	Разрешение:	300 -			
<u>⊕</u> . Общие	Выход за обрез:	0,0 миллиметры			
	Установки	принтера Сохранить специальную страницу			
	Добавить рамку страницы				
•					
			ОК Отмена	Справка	

Рис. 16. Окно меню Параметры страницы

## 2. Использование инструментов рисования

С помощью инструментов рисования создаются графические примитивы – линии и геометрические фигуры, из которых в дальнейшем строится изображение.

## Создание графических примитивов

1. Используя группу инструментов *Свободная форма*, создать разные типы линий (рис. 17).



Рис. 17. Группа инструментов для рисования линий

На уровень гладкости кривой влияет значение параметра *Гладкость* в окне диалога *Параметры*, открываемого двойным щелчком на пиктограмме инструмента *Свободная форма*. Значение этого параметра может изменяться от 0 до 100 (рис. 18).

Пар	Параметры					
	Монитор и	Кривая/Линия Безье				
	… Изменить … Привязка к объеі … Динамические на	Сглаживание свободной формы: Порог угла:	5 — пикселей			
	Предупреждения VBA	Порог спрямления:	5 🚔 пикселей			
	Память Память	Автообъединение:	5 🚔 пикселей			

Рис. 18. Параметры сглаживания

Чем больше величина параметра, тем более плавной получается кривая при рисовании. Чем меньше значение этого параметра, тем точнее кривая соответствует движениям мыши (рис. 19).



Рис. 19. Эффект сглаживания А – линия со сглаживанием; Б – линия без сглаживания

2. Нарисовать прямоугольник с помощью инструмента *Прямоугольник*, растянуть прямоугольник до нужных размеров. Если удерживать нажатой клавишу <Shift>, то прямоугольник будет рисоваться «от центра», а не «от края».

- Нарисовать квадрат. Выполнить все предыдущие действия при нажатой клавише < Ctrl>.

3. Инструмент Эллипс используется для рисования эллипсов и окружностей. Core1DRAW определяет эллипс через прямоугольник, описанный около него.

- Нарисовать эллипс. Установить курсор мыши в угловую вершину запланированного прямоугольника, в который должен быть вписан эллипс, растянуть прямоугольник до нужных размеров, внутри прямоугольника появится эллипс. Если удерживать нажатой клавишу <Shift>, то эллипс будет рисоваться «от центра», а не «от края».

- Нарисовать окружность. Выполнить все предыдущие действия при нажатой клавише < Ctrl>.

#### Выделение объектов

Правило CorelDRAW: выделить объект и только после этого выполнять над нам преобразования! Для выделения объектов перед их преобразованием используется инструмент *Указатель*.

 Выделить объект с помощью инструмента Указатель. Щелкнуть мышью на контуре объекта. Выделенный объект будет окружен маркировочной рамкой (рис. 20). Последний нарисованный объект всегда становится выделенным автоматически.



Рис. 20. Выделенный объект

2. Выделить несколько объектов. Есть несколько способов выделения группы объектов:

Способ 1

- Выделить первый объект.

- Щелкнуть мышью при одновременно нажатой клавише <Shift> на втором, третьем и т.д. объектах. Маркировочная рамка будет охватывать все выделенные объекты.

Способ 2

- Установить курсор мыши в том месте, где вы хотите расположить один из углов маркировочной рамки. Нажать левую кнопку мыши.

- Растянуть при нажатой кнопке штриховую рамку. Когда все объекты попадут внутрь рамки, отпустить кнопку мыши.

- Чтобы отменить выделение объекта из группы, необходимо выделить объект еще раз при нажатой клавише <Shift>.

#### 3. Основы работы с объектами

#### Операции преобразования

Над любым объектом или группой объектов в векторной графике можно выполнять различные преобразования, основными операциями являются:

- удаление объектов;

- перемещение объектов;

- вращение и перекос объектов;

- изменение размеров объектов;

- копирование и зеркальное отражение объектов;

- создание рисунков из простых геометрических примитивов.

Любые преобразования над объектами можно выполнять только в том случае, если они предварительно были выделены.

1. Создать на рабочем листе несколько объектов: эллипс, прямоугольник, окружность – провести операции над этими объектами.

Использование инструмента Форма для изменения формы объекта (рис.
 21). С его помощью можно:

скруглить углы прямоугольника:

- выделить прямоугольник инструментом *Форма*;

- установить курсор мыши на узел одного из четырех углов и перемещать его внутрь при нажатой кнопке мыши. Чем дальше вы переместите узел, тем более круглыми станут все четыре угла;

нарисовать сектор или дугу:

- выделить круг инструментом  $\Phi opma;$ 

- перемещать узел на контуре круга при нажатой кнопке мыши, при этом символ курсора должен находиться внутри круга; если он будет располагаться вне круга, то получится разомкнутая дуга.



Рис. 21. Использование инструмента Форма для изменения объекта

• Удалить объект.

- выделить объект;

- нажать клавишу <Delete> или воспользоваться командой Удалить контекстного меню;

- отменить последнюю операцию над объектом (меню Правка/Отменить).

• Переместить объект.

- выделить объект, установить курсор мыши на него и передвигать мышь при нажатой левой кнопке мыши.

• Выполнить вращение и перекос объекта.

- дважды щелкнуть мышью на объекте;

- маркировочные квадраты превратятся в двойные стрелки (рис. 22);

- если установить курсор мыши на одну из изогнутых стрелок, можно вращать объект при нажатой кнопке мыши в нужном направлении; если установить курсор мыши на одной из прямых стрелок, можно перемещать ее в нужном направлении до того, как объект примет желаемую форму;

- центр вращения можно переместить в любое место экрана.



Рис. 22. Вращение и перекос объекта

• Изменить размер (увеличить/уменьшить) объекта.

- выделить объект и потянуть за один из угловых маркировочных квадратов;

- объект будет увеличиваться (уменьшаться) только в одном направлении.

Получить копию объекта.

- выделить объект, нажать клавишу <+> или комбинацию клавиш <Ctrl + C>;

- копия будет располагаться над исходным объектом, переместить (снять с оригинала) копию объекта.

• Зеркально отразить объект.

- выделить объект и потянуть за один из угловых маркировочных квадратов через объект в направлении зеркального отражения;

- для получения точного зеркального отражения держите нажатой клавишу

Кроме того, для преобразования объектов имеется группа команд для точного выполнения преобразований. Для этого используется команда Упорядочить/Преобразования.

#### Методы упорядочения и объединения объектов

*Упорядочение объектов* – изменение взаимного расположения объектов относительно друг друга.

*Взаимное выравнивание объектов* – точное выравнивание одного объекта по вершинам или сторонам другого объекта.

*Группирование* – способ объединения объектов, при котором объекты остаются независимыми друг от друга, они только удерживаются вместе. Сгруппированные объекты легко разгруппировать.

Объединение – способ объединения объектов, при котором объекты сливаются в новый объект с единым контуром и заливкой. Общие части объектов становятся прозрачными. Скомбинированные объекты можно разъединить, но они при этом не сохранят своих первоначальных признаков.

*Пересечение* – способ объединения объектов, при котором не сохраняются контуры объектов и удаляются все не перекрывающие друг друга части.

*Исключение* – способ объединения объектов, при котором один объект исключается из другого.

1. Выполнить упорядочение и объединение объектов

- Изменить порядок расположения объектов с помощью команды Упорядочить/Порядок/Изменить порядок.

- Сгруппировать объекты командой Упорядочить/Сгруппировать. Теперь сгруппированные объекты можно преобразовывать как один объект.

- Разгруппировать объекты (команда Упорядочить, Разгруппировать).

- Объединить объекты (команда Упорядочить/Объединить).

- Разъединить скомбинированные объекты (команда Упорядочить/ Разъединить).

- Выполнить исключение одного объекта из другого (эту операцию удобно использовать для получения отверстий). Выделить объект, который нужно исключить. Выполнить команду *Упорядочить/Формирование*, выбрать команду *Исключить*, снять флажки с исходный и конечный объект, щелкнуть на *Исключить*, щелкнуть на объекте, из которого производится исключение.

#### 4. Сохранение и загрузка изображений в CorelDRAW

Сохранение документов в CorelDRAW похоже на аналогичные действия в любой программе, работающей под управлением Windows. Чтобы сохранить документ на диске, в меню Файл выберите *Сохранить* или *Сохранить как*. На экране появится диалог, в котором вы должны ввести имя файла и определить место на диске для сохранения файла. Кроме того, вы можете указать некоторые дополнительные параметры сохранения файла. Дополнительными параметрами в диалоге вы можете задать сохранение в формате предыдущей версии CorelDRAW или в формате иного графического редактора. Вы также можете сохранить в файле только выделенные объекты. Однако в большинстве случаев дополнительная настройка сохранения не требуется. Программа CorelDRAW использует свой собственный векторный формат файлов – *CDR*. Вместе с рисунком сохраняются и следующие настройки:

- размер страницы и её ориентация;

- интервал решётки;

- направляющие.

Если нужно сохранить рисунок в растровом формате, можно воспользоваться экспортированием, для этого выбрать в меню *Файл* команду *Экспорт* и выбрать нужный формат. Для ввода и редактирования изображений, созданных другими графическими программами, можно воспользоваться меню *Импорт*, который позволяет открыть файлы растровых и большинства векторных форматов.

## Практическая работа №2. Работа с цветом в векторной графике. Цветовые модели, заполнение объектов, использование заливки

С помощью инструментов рисования создается контур объекта, внутри которого может быть заполнение (любой цвет, штриховки или изображения). Заполненный объект трактуется как единый элемент, т.е. заполняется вся его внутренняя область.

В CorelDRAW цвет может быть задан отдельно для контура объекта и для самого объекта. Для заполнения объекта используется инструмент *Заливка*, а для контура – инструмент *Абрис*. Заливку можно разделить на следующие категории:

однородная – заполнение одним цветом или штриховкой;

*градиентная* – заполнение, использующее цветовые переходы (линейная, радиальная, коническая, прямоугольная и т.д.);

*узорчатая* – заполнение, при котором используются повторяющиеся изображения (двухцветные или полноцветные);

*текстурная* – заполнение с помощью растровых рисунков (художественные изображения).

#### 1. Использование однородной заливки

1. Запустите программу CorelDRAW и создайте новый документ. Убедитесь в том, что размер документа задан А4, ориентация – книжная.

2. Чтобы избежать ошибок, включите панель Отмена операций. Чтобы включить панель в меню Инструменты, кликните пункт Окно настройки отмены. В появившемся окне вы сможете отслеживать все ваши действия и отменять их по необходимости.

3. Выберите инструмент *Эллипс* либо в панели инструментов, либо нажатием клавиши <F7>. Нарисуйте эллипс, а затем в панели размеров задайте его ширину и высоту, равными 60 мм.

4. Выберите на панели инструментов в графе Заливка режим Диалоговое окно цвета заливки. Вызов окна возможен нажатием комбинации клавиш <Shift+F11>.
5. В раскрывшемся окне Однородная заливка на вкладке Модель выберите цветовую модель RGB, в нижней части окна кликните на панель Параметры, выберите пункт Программы просмотра цвета и в раскрывшемся меню выберите пункт RGB – трехмерная аддитивная.

6. Для заливки выделенного (текущего) объекта с использованием диалогового окна Однородная заливка (рис. 23) необходимо выбрать требуемый цвет и кликнуть ОК.



Рис. 23. Диалоговое окно однородной заливки

Важно! В цветовой модели RGB каждый из основных цветов: *Red* (красный), *Green* (зеленый), *Blue* (синий) – имеют 256 уровней яркости и кодируются целыми числами от 0 до 255. Значению 0 соответствует отсутствие данного компонента цвета, а значению 255 – максимальная его яркость. Сочетание всех возможных значений основных цветов определяет область цветового пространства модели. Для модели RGB смешиванием красного, зеленого и синего цветов можно создать более 16,7 млн цветов.

В окне Однородная заливка цветовая модель RGB представлена в виде трехмерной системы координат. Ось R (*Red*) направлена вверх, ось G (*Green*) – влево и ось В (*Blue*) – вправо. Нужный цвет создается установкой значений основных цветов. Их можно задать вводом числовых значений в числовые поля R, G и B, расположенные правее трехмерной системы, или перемещением ползунков в

виде небольших квадратов на осях основных цветов. Правее трехмерной системы расположена цветовая полоса с ползунком для изменения яркости цвета. Созданный цвет в виде цветового образца отображается в цветовом поле *Новый*, а в поле *Старый* – цвет, который выделенный объект имел раньше (рис. 24).



Рис. 24. Цветовая система координат для создания собственного цвета в модели RGB

7. Закрасьте ваш круг красным цветом. Для этого в окне *Однородная заливка* укажите значение переменной R = 255, а G и B = 0. Щёлкните по клавише *OK*. Круг будет закрашен красным цветом, следует понимать, что этот красный цвет создан по модели RGB.

8. Перекрасьте круг в зеленый, а затем в синий цвета.

# 2. Цветовые модели RGB и CMYK

1. Установите вертикальную направляющую на отметке 50 мм и горизонтальную направляющую на отметке 250. Для этого наведите курсор мыши на линейки, расположенные слева и сверху от листа, зажмите левую клавишу мыши и расположите направляющую на вашем листе исходя из заданных показателей линейки.

2. Нарисуйте круг диаметром 60 мм и расположите его центр в точке пересечения направляющих.

3. Активируйте панель *Цвет*. Для этого на панели инструментов выберите пункт *Заливка*, в раскрывающемся меню кликните на пункт *Окно настройки цвета*.

4. На панели *Цвет* в списке моделей выберите цветовую модель RGB и щелкните на пиктограмму ползунков.

5. Закрасьте круг красным цветом. Для заливки круга после выбора цвета кликните по кнопке *Заливка*, а затем по кнопке *Абрис*, чтобы назначить цвет обводке круга.

6. Создайте копию круга и закрасьте её синим цветом. Для этого установите вертикальную направляющую на отметке 75 мм. Инструментом *Стрелка* захватите нарисованный круг и переместите его вправо, совместив центр с направляющей на отметке 75 мм.

7. Закрасьте созданную копию круга синим цветом.

8. Выделите оба круга, затем перейдите в меню «Упорядочить»/ «Формирование» /«Пересечение».

9. Далее в меню Упорядочить найдите панель Порядок и в всплывающем меню выберите пункт На передний план страницы.

10. Залейте смежную область пурпурным цветом, для этого задайте переменным R и B значение, равное 255, а для переменной G значение оставьте равным 0.

11. Создайте горизонтальную направляющую на отметке 225 мм.

12. Создайте ещё одну копию круга и залейте его зеленым цветом.

13. Повторите пункты 9 и 10. При смешивании красного и зеленого цветов получается желтый.

14. Закрасьте область пересечения красного и зеленого кругов желтым цветом.

15. Проведите подобные операции для зеленого и синего кругов. Закрасьте область их пересечения голубым цветом.

16. Не забудьте про центральную область пересечения трех кругов. Смешивание основных цветов даст белый цвет в области пересечения трех кругов.

17. На новом листе нарисуйте три пересекающихся друг с другом круга диаметром 90 мм. В качестве цветовой модели используйте модель СМҮК. Используя палитру цвета, верхний круг закрасьте голубым цветом (*Cyan*), левый

круг – пурпурным (*Magenta*), правый круг – желтым (*Yellow*). Определите, какого цвета должны быть пересечения кругов, и закрасьте их необходимым цветом. Не забудьте про область пересечения всех трех кругов (рис. 25).



Рис. 25. Изображение цветовых моделей RGB и CMYK

### 3. Использование градиентной заливки

Градиентная заливка обеспечивает создание цветового перехода и часто используется для создания эффекта объема рисунка. В CorelDRAW возможно использование четырех типов градиентных заливок (рис. 26).

*Линейная* заливка обеспечивает цветовой переход вдоль прямой линии от одной стороны объекта к другой.

*Радиальная* заливка меняет цвет от внешнего контура объекта к его центру по окружности.

Коническая заливка обеспечивает переход цвета вдоль периметра объекта.

*Прямая* заливка выполняет цветовой переход от внешнего контура объекта к его центру по прямым линиям.



Рис. 26. Типы градиентных заливок

Настройки градиентной заливки позволяют также задать угол поворота цветовой полосы, количество цветовых полос и их цвет, смещение центра и края полос перехода (рис. 27).

Градиентная заливка		
Тип: Прямые 🔻	Параметры	
Сдвиг центра	Угол: 0,0	
По 0 🔶 %	Шаги; 256 🔄 🖱	
По вертикали: 0 🚔 %	Сдвиг 0 👘 %	
Цветовой переход		
Двухцветный Пастройка     Династройка     Династройка     Династройна     Династро     Династр		
Из: 🔳 🔽		
B:		
Центр:	50	
Заготовки: 🗸 🖓 📼 Параметры PostScript		
	ОК Отмена Справка	

Рис. 27. Диалоговое окно градиентной заливки с параметрами настроек

# Выполнение рисунка «Микроскоп» с применением градиентной заливки и интерактивных инструментов

1. Запустите программу CorelDRAW и создайте новый документ. Убедитесь в том, что размер документа задан А4, ориентация – книжная.

2. Создайте составляющие части рисунка, используя инструменты *Основные фигуры*, *Прямоугольник*, *Эллипс:* 

2.1. Начнем с основания и держателя штатива, выбираем инструмент *Основные фигуры*, на панели атрибутов находим меню *Правильные фигуры* и выбираем трапецию (рис. 28).

□       □	<u>О</u> кно <u>С</u> правка		
	€ 10,0 0 😤		Сверхт 🔻 🕃 🗐
	150	× □□□©©© ≈♡∿\\0	250

Рис. 28. Меню Правильные фигуры на панели атрибутов



Рис. 29. Трапеции, нарисованные с помощью инструмента Основные фигуры

2.3. Выделяем трапецию держателя и в контекстном меню выбираем *Преобразовать в кривую*, и далее с помощью инструмента *Форма* выполняем перекос объекта (рис. 30).



Рис. 30. Преобразование объекта в кривую

2.4. Для того чтобы закруглить верхнюю часть трапеции, необходимо выполнить преобразование верхнего отрезка полученной трапеции в кривую линию и, используя инструмент *Форма*, придать ей нужную форму, при этом отрезок между узлами будет иметь форму кривых Безье, управляя которыми, задаем необходимую форму и радиус (рис. 31).



Рис. 31. Преобразование отрезка в кривую и управление кривыми Безье

2.5. Аналогично создаем остальные делали рисунка и располагаем их в нужном порядке (рис. 32). Таким образом мы собираем сложный рисунок из простых геометрических примитивов.



Рис. 32. Сборка рисунка из составляющих примитивов

3. Используя заполнение в виде градиентной заливки, создадим эффект объема деталей микроскопа.

3.1. Выделите одну из деталей рисунка, из группы дополнительных инструментов Заливка выберите Градиентная заливка, в диалоговом окне выберите тип заливки, установите цветовой переход и угол цветовой линии. Для прямоугольных деталей будем использовать тип заливки – линейная, а для круглых – радиальная.

3.2. Для того чтобы выполнить градиентную заливку цилиндрических частей микроскопа, необходимо настроить несколько цветовых переходов. Для этого в диалоговом окне градиентной заливки установите *Цветовой переход – Настройка*, на пиктограмме с помощью маркировочных указателей задайте дополнительные цветовые линии, при этом активная линия отмечена черным цветом указателя, для нее можно выбрать нужный цвет из предложенной палитры либо в отдельном окне заливки во вкладке *Другие* (рис. 33).



Рис. 33. Настройка градиента с несколькими цветовыми лучами

3.3. Обратите внимание, что для создания объема необходимо установить контрастные цвета для цветовых линий. Далее задайте угол наклона полученных цветовых линий в соответствии с углом наклона объекта. Для этого либо измените значение параметра *Сдвиг края* в нужную сторону или при помощи мыши измените направление цветовой линии на пиктограмме в углу окна с образцом заданного градиента заливки (рис. 34).

	Градиентная заливка
<u>~</u>	Тип: Линейные  Тип: Параметры Угол: 45,8
	По срузонтали: 0 10 10 10 10 10 10 10 10 10 10 10 10 1
	Цветовой переход © Двухцветный <sup>®</sup> Настройка
	Заготовки: • Фр Параметры PostScript ОК Отмена Справка

Рис. 34. Окно настройки градиента и полученный результат заполнения объекта

3.4. Таким образом, примените градиентную заливку ко всем частям рисунка, при этом не забывайте, что у объекта нужно задать также соответствующий цвет абриса. Обратите внимание, что при использовании градиентной заливки цвет объекта становится неоднородным и поэтому абрис может выделяться на рисунке, в таком случае можно удалить абрис совсем, для этого в группе инструментов *Абрис* выберите *Без абриса* (рис. 35) или на панели свойств инструмента *Абрис*.



Рис. 35. Группа инструментов Абрис

3.5. При выполнении заполнения деталей рисунка следите за взаимным расположением объектов относительно друг друга, для того чтобы изменить положение объекта в меню *Порядок*, выберите одно из предложенных команд – на передний план, на задний план, а также можно расположить объект на уровень назад или вперед относительно исходного положения.

3.6. При необходимости перемещения или заполнения группы объектов перед этим их удобно сгруппировать. Для того чтобы в дальнейшем работать с ними по отдельности, можно их разгруппировать (рис. 36).



Рис. 36. Заполненные заливкой детали рисунка «Микроскоп»

3.7. Для того чтобы рисунок выглядел еще более объемно, к нему можно добавить эффект тени. Для этого выделите все детали рисунка вместе и выполните их группировку для того, чтобы наш рисунок стал единым объектом. Далее в группе Интерактивные инструменты выберите Интерактивная тень (рис. 37).



Рис. 37. Группа интерактивных инструментов

3.8. Затем задайте тень рисунку, точкой начала тени должно являться основание изображенного предмета (рис. 38).



Рис. 38. Добавление интерактивной тени к рисунку «Микроскоп»

3.9. На панели свойств инструмента тень вы можете установить угол тени, непрозрачность, размытие и затухание тени для того, чтобы она выглядела наиболее естественно.

3.10. И в завершение добавим фон нашему рисунку. Нарисуйте прямоугольник размером во весь рабочий лист. Выделите его, в группе инструментов Заливка выберите заливку узором, в диалоговом окне заливки Узор отметьте пункт *Регистр* и выберите понравившийся рисунок заливки; кроме того, здесь также можно установить дополнительные параметры для отображения узора: скосить, повернуть, изменить размер узора (рис. 39).



Рис. 39. Добавление узорчатой заливки к объекту

3.11. В завершение вызовите контекстное меню выделенного объекта и в пункте *Порядок* выберите *На задний план страницы*. Заполненный узором прямоугольник переместится за наш рисунок и будет играть роль фона (рис. 40).



Рис. 40. Готовый рисунок «Микроскоп»

# Практическая работа №3. Работа с текстом в CorelDRAW

Текстовые объекты в CorelDRAW бывают двух основных типов (рис. 41):

Абзацный текст – Используется для создания заголовков, надписей, подписей. Можно применять различные художественные эффекты.

*Строчный текст* – Используется для создания больших текстовых блоков фиксированной ширины. Позволяет настроить более сложное форматирование, чем у строчного текста.

# Так выглядит художественный текст

Так выглядит абзацный текст

Рис. 41. Типы текстовых объектов CorelDRAW

# Создание текстовых объектов

Для создания текстовых объектов используется инструмент *Текст* панели Инструментов (рис. 42). Чтобы создать *строчный* текст, следует:

– выбрать инструмент *Текст*;

 щелкнуть на свободном месте страницы: после этого будет зафиксировано положение текстового курсора;

– набрать текст.

Чтобы создать абзацный текст, следует:

– выбрать инструмент Текст;

– создать с его помощью прямоугольную рамку: после этого будет зафиксирована рамка для абзацного текста, в которой появится текстовый курсор;

– набрать текст.



Рис. 42. Инструмент Текст

Если необходимо создать строчный текст в несколько строк, необходимо использовать клавишу <Enter> для перевода курсора на новую строку. При

создании абзацного текста клавишу <Enter> следует использовать только для создания нового абзаца, а строки в пределах абзаца будут создаваться автоматически, подобно тому, как это происходит в Microsoft Word.

# Редактирование текстовых объектов

Редактировать текст можно несколькими способами.

1-й способ. Дважды щелкнуть мышью в том месте текстового объекта, где необходимо внести изменения, а затем начать редактирование.

2-й способ. Выделить текстовый объект с помощью инструмента Указатель, а затем выбрать команду меню *Текст/Редактировать текст*. Далее начать правку в специальном диалоговом окне *Редактировать текст*.

3-й способ. Выделить текстовый объект, а затем воспользоваться инструментом *Правка текста* на панели свойств.

# Форматирование текстовых объектов

Форматирование текстовых объектов в CorelDRAW можно осуществлять:

а) с помощью специальных диалоговых окон;

б) с помощью специальных докеров;

в) с помощью панели свойств (рис. 43).



Рис. 43. Форматирование символов с помощью панели свойств

#### Форматирование отдельных символов

Для того чтобы форматировать отдельные символы текстового объекта, нужно выделить эти символы. Это можно сделать в режиме редактирования текстового объекта, используя те же приемы выделения текста, которые используются, например, в Microsoft Word.

Более быстрый способ выделить несколько отдельных символов в произвольном порядке – воспользоваться инструментом *Форма*:

– выделить объект;

– выбрать инструмент  $\Phi o p M a$ ;

– щелкать, удерживая клавишу <Shift>, на маленьких квадратных маркерах, расположенных слева внизу от символов. При этом на панели свойств можно будет задавать форматирование для выделенных символов, а также можно их свободно перемещать мышью и задавать для них произвольную заливку (рис. 44).



Рис. 44. Панель свойств инструмента Форма

### Выравнивание текста по траектории

В качестве базовой линии текстового объекта можно выбрать произвольную траекторию, для того чтобы расположить текст по заданной траектории, можно воспользоваться несколькими способами.

1-й способ. Создать произвольную траекторию. Затем выбрать инструмент *Текст,* подвести его к контуру траектории и после того, как курсор примет вид буквы *А* с волнистой линией, щелкнуть мышью, чтобы зафиксировать текстовый курсор и начать ввод текста.

2-й способ. Создать произвольную траекторию и текстовый объект с необходимым содержимым. Выделить текстовый объект, выбрать команду меню *Текст/Подогнать под траекторию*, а затем щелкнуть мышью по траектории.

3-й способ. Создать произвольную траекторию и текстовый объект с необходимым содержимым. Правой кнопкой мыши переместить текстовый объект на траекторию и после того, как курсор примет вид прицела, отпустить мышь: в появившемся контекстном меню выбрать команду *Подогнать текст под траекторию*. Рекомендуется применять такое выравнивание только для строчного текста.

### Выравнивание абзацного текста под рамку

Рамка для абзацного текста может быть произвольной формы, а не только прямоугольной.

– Создать произвольную замкнутую форму и абзацный текст с необходимым содержимым. Правой кнопкой мыши переместить текстовый объект на замкнутую форму. В появившемся контекстном меню выбрать команду Поместить текст внутрь.

– Чтобы абзацный текст полностью занял отведенную для него рамку, выделите его и выберите команду меню *Текст/Рамка абзацного текста/Подогнать текст под рамку*.

### Преобразование текста в кривые

Любой текстовый объект можно преобразовать в кривые с помощью уже известной нам команды *Преобразовать в кривые*. Для этого необходимо несколько усложнить уже имеющиеся текстовые символы. Преобразование текста в кривую дает в такой ситуации возможность свободного изменения формы символов. Следует учитывать, что абзацный текст, как правило, преобразуется в группу кривых.

# Разбиение текста

Любой текстовый объект можно разделить на несколько независимых текстовых объектов.

– Для этого нужно выделить текстовый объект и выбрать команду меню Упорядочить/Разбить текст. Обратите внимание, что каждая строка текстового объекта теперь представляет собой независимый текстовый объект.

– Если выделить любую из получившихся строк и вновь разбить ее на части, то каждое слово в строке будет представлять собой независимый текстовый объект, так же можно сделать каждый символ отдельным объектом.

# 1. Создание визитной карточки

1. Откройте новый документ формата А4 книжной ориентации.

2. Создайте прямоугольник размером 90 х 50 мм (он нужен только для указания габаритов и будет удален).

3. Включите инструмент *Текст*.

4. Шрифтом Arial, кеглем 8 наберите строчными буквами как один блок Artistic Text текст, как в примере, представленном на рис. 45.



Рис. 45. Образец создания абзацного текста

5. Назначьте выключку по центру *Ctrl+E>*. На остальные параметры текста пока не обращайте внимания.

6. Вырежьте через буфер обмена слова «ФГБОУ ВО Пермский государственный национальный исследовательский университет» и поместите их как отдельный текстовый блок. С помощью диалогового окна *Текст/Регистр* (<Shift+F3>) назначьте для них вариант верхний регистр и разделите его на два абзаца, установите кегль 9 пт, выключка по центру, и разместите вверху прямоугольника.

7. Вырежьте слова «Кафедра ботаники и генетики растений» и преобразуйте их в отдельный текстовый блок. Назначьте этому тексту жирное очертание.

8. Аналогичным образом создайте отдельный текстовый блок в два абзаца со словами «Петров Петр Петрович». Назначьте тексту кегль 12, начертание – жирный, выключка по центру.

9. Слову «преподаватель» в отдельном текстовом блоке назначьте кегль – 9, начертание – курсив.

10. Оставшимся словам назначьте начертание жирный и разделите их на два блока, каждый по два абзаца, выключка по центру. В одном блоке будут содержаться адрес и телефон, в другом – электронный адрес и адрес сайта.

11. При помощи инструмента *Прямоугольник* создайте разделительные линии, как на рис. 46.



Рис. 46. Создание визитной карточки

12. Расположите все созданные и отформатированные тексты используя команды Упорядочить / Сгруппировать / Преобразование / Положение / Относительное / Применить к копии (рис. 47).

* Преобразо	вание — 🔺 🗙	
<b>+</b> 0	2 II L	
Положени	e:	
Г.: 97,93	21 🚔 мм	
B.: 0,0	мм	
🔽 Относительное		
Применить к копии		
Применить		

Рис. 47. Пристыковочное окно инструмента Преобразование

13. В результате получаем печатный лист с копиями визитной карточки (рис.48).

·····	·····
Петров Петр	Петров Петр
Петрович	Петрович
c Mone, yii, Nyveper, 3-15 overal, apologithmetra,	ствонь ук вунерик, к15 ослаг россуфилато.
1. SC20007 WWW.ppers	к экспертор WWW росси
C 61-02/10/10/10/02/07/10/22/07/10/02/02/07/10/10/20/07/10/10/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/07/10/20/02/20/00/200/00/	THE SAME THE RESERVENTS CONFIDENTIAL AND DESCRIPTION OF THE SAME AND THE SAME
Петров Петр	Петров Петр
Петрович	Петрович
с Полак, ул. Бугарак, д.13.	с Пола, ул. Бунрая, д.13.
1. ЭКЗВИЯТ	з эконого — Молания
УУУШИДСК	може рассейских с
<ol> <li>впосилотический постранстванный мыроснизаний 1</li></ol>	Г НЕОУ ПОНИЧЕСКИЙ ПОСИДИСТВИНЫЙ МАДИСКАЛЬНЫЙ Г 1 ИССЛЯДСКИЛИНСКИЙ МИТВИСКИМ П Вафиара Полинии и понтики распиева Вафиара Полинии и понтики распиева
Петров Петр	Петров Петр
Петрович	Петрович
с.Поль. уз. Бунрак, а.13 1. ЭКЗВЕР УМУЖДала	с Парак, ул. Бунфан, а 13. онлал, росуфеанта. 1. 2628/007 ЖИМ (расс
Г. ОТНОГ ЛОТИРИСКИЙ ГОСИДИИСТВИНИИЙ НИЦИСКИХНИМ Г.	Г Изаку по пониский посиди Спаниный надиональный 1
1. ИССТВАДИИЛЬВООЙ НИХВИСИТИТ	1 исследовильноский нителестикт
Вофевда Болании и пинтики растики	Кофицал Билинан и полтании растики
Петров Петр	Петров Петр
Петрович	Петрович
с Пола, уз. Бунра, 213 онал. госорфията. 1. Эконог WWW.gu.co	с Пола, ул. Бунфан, а 13 1. ЭКОНИЯТ — МАНИДАЛА. МАНИДАЛА
1 Выропрания и полная распесатиля и полная полная и полная полн Полная полная полн	Г Набола Палаский по сприблажный мироскийский "1 1 исследский какономисский пиланский т Вафиара Полинание платовие располя
Петров Петр	Петров Петр
Петрович	Петрович
с Палав, ул. Бунярая, а 13 селет, <u>консеферат ок</u> 1. 2008/07 WWW.804.04	с Пазав. ул. Бухорая, а Та з. экснедината. Эксперията.
Г отвор полничиский постранстванный мыроснизаний 1 исстадовизаньской пиланости I Каревра Бланеван и пентови распеса	Г техот по телевой по сприставные и изресни зама 1 1 исследски закоой интерства Варада Болекка и телетока растика Стал
Петров Петр	Петров Петр
Петрович	Петрович
с Полас. ул. Бунфон, 3.13 онлаг, арходфият.о.	с Полас ул. Бучерок, а 13. оновал россуфскала.
1. Эксперт — МУМ Архон.	1. ЭКОНИЕТ
Сарадо в изаема и полная растия Карада в изаема и полная растия	Г НЕОГЛАЛИНСКИЙ ГОЛУКИТЫКАЛ РАДИСКАЛЫТ"   ИССЛЬДСКИЗНЬСНИЙ ИНТВИГОТТИ Вафтара Боланкен и понтоке растичий
Петров Петр	Петров Петр
Петрович	Петрович
- Doons yo kyopes, a 19 ernet, according to.	C Thome, yo. Byregine, all constrained as a second or constrained as WWW assess

Рис. 48. Печатный лист визитных карточек

# 2. Создание круглой печати

1. Создайте новый документ формата А4.

2. Нарисуйте три концентрические окружности без заливки диаметром 60, 45 и 35 мм. Для этого, как только нарисуете первую окружность, переведите курсор в центр окружности, зажмите клавиши <Shift> и <Ctrl> и рисуйте следующую окружность.

3. Щелкните правой кнопкой по первой окружности, выберите в всплывающем меню пункт *Свойства*. Во вкладке *Абрис* установите значение ширины 0,5. Повторите для двух других окружностей, устанавливая значение ширины абриса для второй окружности равным 0,1; для третьей – равным 0,5 соответственно.

4. Наберите три отдельных текстовых блока шрифтом *Arial*, кеглем 12 пт., соблюдая регистр, по примеру:

ПРЕПОДАВАТЕЛЬ КАФЕДРЫ БОТАНИКИ И ГЕНЕТИКИ

ПЕТРОВ ПЕТР ПЕТРОВИЧ

Выпускник ПГНИУ

5. Выберите текст «ПРЕПОДАВАТЕЛЬ КАФЕДРЫ БОТАНИКИ И ГЕНЕТИКИ», задайте ему кегль 20, затем перейдите в меню *Текст/Текст вдоль пути* и разместите текст внутри первой окружности (рис. 49).



Рис. 49. Образец расположения текста вдоль пути

6. Тексту «Петр Петрович...» задайте кегль 14 и повторите операцию 6, разместив текст во второй окружности.

7. Тексту «Выпускник ПГНИУ» задайте кегль 12 и разместите его внутри третьей окружности.

8. При помощи инструмента *Форма* распределите тексты по длине всех окружностей (рис. 50).



Рис. 50. Создание печати

# 3. Создание логотипа

1. Создайте новый файл формата А6, альбомной ориентации.

2. Наберите слово «Био» шрифтом *Arial*, кеглем 100, регистр верхний. Назначьте начертание – жирный.

3. Преобразуйте текст в кривые.

4. Инструментом *Форма* добавляйте новые узлы, удаляйте ненужные, изменяйте тип сегментов между узлами. Перемещайте узлы, отредактируйте форму кривой, придав ей желаемый вид (рис. 51).



Рис. 51. Пример создания логотипа

# РЕДАКТОР РАСТРОВОЙ ГРАФИКИ АДОВЕ РНОТОЅНОР

#### Введение в программу Adobe Photoshop

Adobe Photoshop – это многофункциональный графический редактор, разработанный и распространяемый фирмой Adobe System. Предназначен в основном для работы с растровыми изображениями, однако имеет определенное количество инструментов для работы с векторной графикой. Данное программное обеспечение занимает лидирующее положение среди графических редакторов растровой графики и распространяется на большинство современных платформ, таких как: macOS, Windows, iOS, Windows Phone, Android. Изначально Adobe Photoshop создавался как средство для работы с полиграфией, однако благодаря разнообразным возможностям стал широко использоваться в веб-дизайне. В программе поддерживается значительное количество цветовых моделей:

• RGB,

• Градации серого,

• CMYK,

• LAB,

• Doutone,

• 256-цветовая палитра,

- Черно-белые,
- Многоканальность.

Осуществляется поддержка обработки изображений с глубиной цвета 8 битов (256 градаций на один канал), 16 битов (используется 15 битов плюс один уровень – 32769 уровней) и 32 бита. Таким образом, среда Photoshop широко применяется фотографами, дизайнерами, художниками, иллюстраторами и др. Однако ввиду больших возможностей Photoshop является довольно сложной программой для освоения и требует определенного уровня навыков для комфортной работы с ним; следует также упомянуть, что чаще всего пользователи не реализуют все заложенные возможности данного ПО, а работают только с теми областями, которые непосредственно нужны им. К подобным возможностям можно отнести: работа по оптимизации четкости изображения, ретушь и коррекция графических файлов, цветовая коррекция, создание коллажей, непосредственное создание рисунков.

Переходим к интерфейсу программы Adobe Photoshop и ознакомимся с её составляющими частями. Как только вы запустите программу, перед вами откроется окно, содержащее все основные панели (рис. 52).



Рис. 52. Основные панели программы Adobe Photoshop

Строка меню – стандартный элемент для всех оконных приложений. Для нее есть стандарты, принятые во всем мире, и Photoshop не стал исключением в том смысле, что она начинается с *File* (Файл) и заканчивается *Help* (Справка). Состоит из следующих меню:

• *Файл* – этим меню приходится пользоваться в основном в начале и конце работы, так как большинство его функций связаны с созданием, открытием и сохранением файлов. Лучше сразу научиться пользоваться клавиатурными сокращениями для запуска этих команд — вы сможете сэкономить много времени;

• *Редактирование* – в этом меню находятся основные команды редактирования, многие из которых есть и в других программах: *Undo* (Отменить), *Paste* (Вставить), *Copy* (Копировать) и т. д.;

• Изображение – меню, созданное для работы непосредственно с изображением. Многие его команды придется применять почти в каждой работе;

- Слой работа со слоями;
- Шрифт работа со шрифтами;

• Выделение – создание и редактирование выделения;

• *Фильтр* – в этом меню находятся фильтры — дополнительные модули Photoshop, выполняющие самые разные функции;

- *3D* создание и работа с 3D-объектами;
- Просмотр различные инструменты для удобства работы и просмотра;

• *Окно* – с помощью этого меню можно управлять отображением палитр и размещением документов на экране;

• Справка – команды справочной системы.

Палитра инструментов – с этим элементом программы вам придется работать больше, чем со всеми остальными вместе взятыми. Здесь находятся те инструменты, при помощи которых вам предстоит делать основную работу. Подробнее инструменты будут рассмотрены ниже. Пока обратите внимание на то, что на кнопках многих инструментов нарисованы стрелки. Это значит, что за кнопкой скрывается всплывающая панель с дополнительными инструментами. Чтобы открыть ее, нужно щелкнуть мышью на инструменте и не отпускать какое-то время кнопку мыши. Когда вы выберите на всплывающей панели инструмент, он появится на основной палитре вместо предыдущего. Это сделано для минимизации размеров палитры.

Панель параметров – здесь отображаются все изменяемые настройки инструментов. В зависимости от выбора инструмента возможные опции меняются.

Палитра цветов – служит для быстрого выбора цвета и изменения оттенков.

Палитра слоёв – служит для работы со слоями. Понятие «слой» в программе Photoshop имеет точно такое же физическое значение, как и в обычной жизни, – это прослойка или пластинка, являющаяся составной частью чего-то целого.

Важным моментом при работе с программой Adobe Photoshop является использование «горячих» клавиш, которые значительно упрощают ваши действия и уменьшают время, затраченное на совершение того или иного процесса. Постарайтесь с первого же дня вашей работы с Photoshop поменьше использовать мышь при выборе команд в меню. Обратите внимание на то, что напротив большинства из них написаны клавиатурные сокращения, нажав которые, вы

выполните то же самое, но гораздо быстрее. Привыкайте сразу к правильной работе, иначе потом будет трудно переучиваться. Для вашего удобства запомните следующие сочетания «горячих» клавиш, представленных в табл. 2.

Комбинация клавиш	Действие
Ctrl + A	выделить ВСЕ
Ctrl + C	копировать
Ctrl + V	Вставить
Ctrl + N	новый документ
Shift + Ctrl + N	новый слой
Ctrl + S	Сохранить
Shift + Ctrl + S	сохранить как
Alt + Shift + Ctrl + S	сохранить для Web
Ctrl + Z	вернуть назад
$\overline{Alt + Ctrl + Z}$	вернуть на несколько шагов назад
Shift + Ctrl + Z	вернуть на шаг вперед
Ctrl (-)	уменьшение изображения
Ctrl (+)	увеличение изображения
$\overline{Alt + Ctrl + 0}$	натуральный размер изображений
Alt + колесико мыши	изменение масштаба
Ctrl + колесико мыши	движение по горизонтали
Колесико мыши	движение по вертикали

Таблица 2. Часто применяемые сочетания «горячих» клавиш программы

Adobe Photoshop

Практически вся ваша деятельность в программе Adobe Photoshop так или иначе будет связана с инструментами. Именно они позволяют совершать большинство рабочих операций, будь то выделение, рисование или ретуширование. Доступные инструменты программы перечислены в табл. 3.

Таблица 3. Инструменты Adobe Photoshop

►+	Перемещение – данный инструмент предназначен для выбора и
	переноса слоев по документу. Быстрая клавиша <v>.</v>
[]]	Прямоугольная область – инструмент выделения области
	документа. Допускает выделение прямоугольником, овалом,
	колонкой и строкой. Быстрая клавиша <m>.</m>
<i>\\</i>	<i>Лассо</i> – более продвинутый инструмент выделения области
	документа сложной формы. Имеет три разновидности:
	Лассо – выделенная область определяется траекторией движения
	курсора;
	Полигональное лассо –выделение, осуществляемое через вершины
	(выделенные точки) замкнутого многоугольника;
	Магнитное лассо – инструмент сам определяет границы
	выделяемого объекта.
	Быстрая клавиша <l>.</l>
×.	Быстрое выделение «Магическая палочка» – инструментом
	«магическая палочка» выделяют области одного цвета, причем
	точность выделения можно задавать изменением чувствительности
	«палочки». Быстрая клавиша <w>.</w>
¥,	Кадрирование – инструмент, предназначенный для кадрирования
	области документа. Быстрая клавиша <c></c>
ø.	Пипетка – набор инструментов «пипетка», линейка, цветовой
	эталон и счетчик. Быстрая клавиша <i>.</i>
æ.	Инструмент для коррекции необходимой области. Включает в себя
	точечную восстанавливающую кисть, восстанавливающую кисть и
	заплатку. Быстрая клавиша <j>.</j>

1.	Кисть – инструмент, предназначенный, как следует из названия,	
	для рисования. Имеет две разновидности: карандаш и кисть. Обе	
	разновидности допускают дополнительные настройки. Быстрая	
	клавиша <b>.</b>	
<u> </u>	Штамп – инструмент клонирования выделенной области, полезен	
	при ретушировании фотографий. Быстрая клавиша <s>.</s>	
Z.	Архивная кисть – оригинальный инструмент, взаимодействующий с	
	историей изменения документа. При наведении кисти	
	просматривается предыдущее состояние определённой области	
	документа. Быстрая клавиша <y>.</y>	
Ø.	<i>Ластик</i> – инструмент, аналогичный обычному ластику для	
	рисования. Имеет три разновидности: ластик, фоновый ластик,	
	волшебный ластик. Быстрая клавиша <e>.</e>	
۵.	Заливка – инструмент предназначен для заливки выделенной	
	области градиентным или однородным цветом. Клавиша <g>.</g>	
•	Размытие – инструмент размывания или наведения резкости	
	изображения. Имеет и третью разновидность – «палец»,	
	предназначенный для вытягивания изображения.	
	Осветление – инструмент осветления или затемнения области	
	рисунка. Быстрая клавиша <o>.</o>	
ø.	<i>Перо</i> – многофункциональный инструмент «перо»,	
	предназначенный для построения и редактирования пути. Быстрая	
	клавиша <p>.</p>	
T	Текст – инструмент для набора и редактирования текста как в	
	горизонтальном, так и в вертикальном направлении. Быстрая	
	клавиша <t>.</t>	
A.	Контур – выделяет фигуры или сегменты, отображая узловые	
	точки, линии направления и точки направления. Быстрая клавиша	
	<a>.</a>	

æ.	Фигура – инструмент для создания геометрических фигур. Быстрая	
	клавиша <u>.</u>	
*	Рука – переводит курсор в форму руки при перемещении по	
	документу. Быстрая клавиша <h></h>	
Q	Масштаб – инструмент увеличивает или уменьшает изображение	
	документа. Быстрая клавиша <z></z>	
<b>⊡</b> +	Цвета плана – инструмент, устанавливающий изначальные цвета	
	фона и переднего плана. Стрелки меняют между собой	
	установленные цвета фона и переднего плана. Клавиша <d></d>	
	Значение цвета плана – показывает текущие значения цветов фона	
	и переднего плана	
	Быстрая маска – инструмент включения /выключения режима	
	быстрой маски. Быстрая клавиша <q></q>	
ē,	Переключение вида отображения. Быстрая клавиша <f></f>	

Обратите внимание на то, что это не весь перечень доступных инструментов программы Photoshop, для компактизации пространства разработчиками Photoshop была принята мера вложения инструментов друг в друга. Чтобы посмотреть вложенные инструменты, совершите длительный клик по пиктограмме инструмента – раскроется вложенное меню, где вы сможете выбрать дополнительные инструменты.

Как уже говорилось paнee, основным объектом в Adobe Photoshop, с которым совершается работа, является слой. Практически все изображения, с которыми совершаются те или иные действия в программе, подобны слоёному пирогу и состоят из некоторого количества слоёв. Таким образом, в ПО существует возможность редактирования режимов наложения слоёв друг на друга. От режима наложения, указанного в элементах управления панели параметров, зависит, какое изображении влияние будут испытывать пиксели В при использовании определенного инструмента рисования или редактирования. При описании

визуальных эффектов режима наложения используются перечисленные далее термины, обозначающие цвета:

- Основной цвет это исходный цвет в изображении.
- *Совмещенный цвет* это цвет, накладываемый с помощью инструмента рисования или редактирования.
- *Результирующий цвет* это цвет, полученный в результате наложения.

При использовании режимов наложения слои начинают взаимодействовать друг с другом, в результате чего получается интересный эффект. Доступные режимы наложения представлены на рис. 53.

Нормальный	Normal
Растворение	Dissolve
Подложка	Behind
Очнстка	Clear
Замена темным	Darken
Умножение	Multiply
Затемнение основы	Color Burn
Линейный затемнитель	Linear Burn
Темнее	Darker Color
Замена светлым	Lighten
Осветление	Screen
Осветление основы	Color Dodge
Линейный осветлитель	Linear Dodge (Add)
Светлее	Lighter Color
Перекрытие	Overlay
Мягкий свет	Soft Light
Жесткий свет	Hard Light
Яркий свет	Vivid Light
Линейный свет	Linear Light
Точечный свет	Pin Light
Жесткое смешение	Hard Mix
Разница	Difference
Исключение	Exclusion
Цветовой тон	Hue
Насыщенность	Saturation
Цветность	Color
Свечение	Luminosity

Рис. 53. Режимы наложения

Многие пользователи просто прогоняют слой по всем режимам в поисках лучшего варианта. Конечно, будет намного удобнее, если хотя бы немного разбираться в этих режимах и понимать, для чего нужен каждый из них. Итак, когда мы устанавливаем режим смешивания, то слой, которому изменили режим, начинает взаимодействовать с нижестоящим. Рассмотрим подробнее доступные режимы наложения. *Нормальный* – уже из названия этого режима становится понятно, что это обычное состояние слоя. Здесь нет взаимодействия между слоями (активным и нижестоящим).

*Растворение* – этот режим в случайном порядке удаляет некоторые пиксели активного слоя, что придает эффект пористости или шума.

Подложка – в этом режиме редактирование или рисование осуществляется только на прозрачной части слоя. Он применяется для работы только со слоями, для которых отменен выбор параметра «Блокировать прозрачность», и аналогичен рисованию на обороте прозрачных областей листа прозрачной монтажной основы.

*Очистка* – в этом режиме обеспечивается редактирование или рисование каждого пиксела и преобразование его в прозрачный. Этот режим доступен для инструментов *Фигура* (когда выбрана область заливки<sup>□</sup>), инструмента *Заливка*, инструмента *Кисть*, инструмента *Карандаш*, команды *Выполнить заливку* и команды *Выполнить обводку*. Для использования этого режима необходимо перейти к слою, для которого отменен выбор параметра *Блокировать прозрачность*.

Замена темным – этот режим проявляет только темные оттенки верхнего слоя на нижнем. Его используют для наложения одного слоя на другой, если же вы хотите затемнить, то лучше воспользоваться режимом, следующим по списку, – Умножение.

*Умножение* – это один из самых распространенных режимов. Он накладывает цвета верхнего слоя на цвета нижнего. Цвет становится гуще, от этого изображение темнеет.

Затемнение основы – этот режим путем повышения насыщенности и контрастности дает сильный эффект. Он также немного затемняет, поэтому следует быть аккуратным с этим режимом, так как цвета с ним принимают неестественный вид и принтер не сможет правильно отобразить их.

*Линейный затемнитель* – этот режим затемняет нижний слой, чтобы проявить цвет верхнего слоя путем уменьшения яркости.

Замена светлым – с помощью этого режима получается противоположное действие режиму Замена темным. Проявляет светлые пиксели. Для осветления изображения лучше использовать следующий режим.

*Осветление* – этот режим производит те же вычисления, что и режим У*множение*, но использует при этом инвертированные значения, чтобы проявить конечный результат. Этот режим осветляет изображение. Его можно применять для осветления темных фотографий путем дублирования и установления верхнему слою режима *Осветление*.

*Осветление основы* – похож на режим *Осветление*, но черный цвет на верхнем слое не влияет на результат, другие же цвета будут слегка подцвечивать нижние путем повышения насыщенности и понижения контраста, что позволит подобрать подходящий тон. Хорош для создания сильных эффектов.

*Линейный осветлитель* – то же самое, что и предыдущий режим, но здесь цвета сочетаются путем повышения яркости.

Перекрытие – этот режим равномерно смешивает цвета обоих слоев, хорошо поддается изменениям прозрачности. Он отсеивает светлые части и умножает темные. При низком уровне прозрачности он похож на *Нормальный* режим, но с более интенсивными цветами. Хорошо подходит для наложения текстуры на изображение.

*Мягкий свет* – этот режим осветляет и затемняет цвета изображения в зависимости от цвета верхнего слоя. Это почти то же самое, что фильтр рассеянный свет. Хорошо подойдет для корректировки тонов, а понижение прозрачности даст еще более изысканный эффект.

Жесткий свет – похож на предыдущий режим. Почти не используется, так как *Мягкий свет* более контролируемый. Он будет затемнять изображение, если наверху темное, осветлять, если светлое, и повышать контрастность.

*Яркий свет* – этот режим затемняет или осветляет цвета изображения в зависимости от того, какие цвета на верхнем слое. Создается впечатление, что изображению добавили различные значения контраста.

*Линейный свет* – если верхний слой яркий, то нижний слой будет осветляться, если темный, то затемняться.

*Точечный свет* – с помощью этого режима можно получить разные результаты, работает путем перемещения цветов на изображение в зависимости от того, светлые или темные цвета на активном слое.

*Жесткий микс* – дает похожий результат, контраст здесь на максимуме. Получается очень яркое изображение, причем крупными кусками.

*Разница* – подходит для создания различных эффектов. Он реверсирует тона и цвета, яркость здесь зависит от разницы пикселей.

Исключение – более мягкая версия режима Разница, контраст небольшой.

*Цветовой тон* – здесь цвета верхнего слоя смешиваются с насыщенностью и яркостью нижнего слоя. Это дает сильный эффект. Альтернативный режим – это *Цветность*.

*Насыщенность* – с этим режимом уровни насыщенности нижнего слоя меняются на соответствующие пиксели верхнего слоя. Пригодится для того, чтобы заставить объект принять цвет или текстуру другого.

*Цветность* – используется только яркость нижнего слоя, а цвет и насыщенность идут от верхнего слоя.

Свечение:

Светлее – сравнивает суммарную величину значений всех каналов для совмещенного и основного цветов и отображает цвет с наибольшим значением. В режиме Светлее третий цвет не создается, что может быть результатом наложения Замена светлым, поскольку для создания результирующего цвета выбираются наибольшие значения каналов как основного, так и совмещенного цветов.

*Темнее* – сравнивает суммарную величину значений всех каналов для совмещенного и основного цветов и отображает цвет с наименьшим значением.

Важно! Для 32-битных изображений доступны только следующие режимы смешивания: Обычный, Растворение, Замена темным, Умножение, Замена светлым, Линейное осветление основы, Разница, Цветовой тон, Насыщенность, Цвет, Яркость, Светлее и Темнее.

# Практическая работа №4. Базовые принципы работы в Photoshop.

Работа со слоями

Начиная изучать Photoshop, каждый сталкивается с таким явлением, как «Слои». Постараемся разобраться, для чего же они необходимы.

### 1. Палитра слоёв

Начнем с того, что откроем палитру слоёв в Photoshop. Если она у вас еще не открыта, то идите во вкладку *Окно* и выберите из списка *Слой* или вызовите палитру нажатием клавиши  $\langle F \rangle$ 7, повторное нажатие на клавишу скроет данную палитру. Открыв любой документ (изображение) в Photoshop, вы увидите, как в палитре слоев появилась иконка с этим изображением. Слой в документе один, и в палитре слоев тоже будет отображаться один слой.

Слои можно сравнить со стопочкой книг, листов из тетрадки или альбомом. Вы складываете, а вернее накладываете слои друг на друга, как складываете книги в стопочку. Вы можете поменять книги (слои) местами, можете убрать из стопки книгу или добавить ещё одну. Для тех, кто совсем незнаком со слоями, сравнение со стопкой книг немного поможет в этом разобраться.

# 2. Добавление слоя-изображения

А теперь рассмотрим это на примере. Откройте файл «Для практики» (изображение в формате PNG, и потому оно на прозрачном фоне). Можете выделить лошадь «Ctrl+A», скопировать её «Ctrl+C», затем создать новый документ (по умолчанию после того, как вы скопировали изображение, размеры нового изображения будут соответствовать скопированному, а в разделе размер будет указано *Буфер обмена*). После этого вставить «Ctrl+V» лошадь в файл с изображением.

На изображении в правом углу расположилась лошадь. В палитре слоев мы теперь можем наблюдать два слоя. Первый – это слой «Фон» (он находится внизу) и

второй – слой с лошадью («Слой 1»), он находится выше слоя Фон, а в правом углу иконки в палитре слоев вы можете видеть маленькую лошадь на прозрачном фоне.

#### 3. Выделение (активность) слоя

Вы можете видеть также, что верхний слой, который мы только что вставили, окрашен в синий цвет; это говорит о том, что этот слой выделен (или активен, т.е. он рабочий) и вы можете с ним работать: перетаскивать, удалять, добавлять эффекты слоя и т.д. Другими словами, все изменения, которые мы будем производить, коснутся только выделенного (активного фона). Чтобы выделить другой слой, в данном случае это само изображение (Фон), нужно кликнуть по нижнему слою мышкой, и выделится нижний слой. Рабочим в палитре слоев является выделенный слой (окрашенный в синий цвет).

#### 4. Видимость слоя

Мы можем выключить слой с лошадью (т.е. сделать его невидимым), нажав на иконку с глазом, рядом с иконкой слоя с лошадью). Слой с лошадью стал невидимым, чтобы вернуть ему видимость, нужно снова нажать на глаз. Слой с лошадью стал видимым. Для наглядности и для закрепления знания проделайте то же самое с нижним слоем «Фон». Сначала отключите видимость слоя, нажав на иконку с глазом, потом включите.

Таким образом, мы можем отключать видимость любого слоя, находящегося в палитре слоев, если он нам временно не нужен или в данный момент мешает видеть наши действия на мониторе.

### 5. Перетаскивание слоя

Что же мы еще можем делать со слоем с лошадью («Слой 1»)? Мы можем перетаскивать нашу лошадь по всему файлу, куда нам заблагорассудится, но для этого в палитре слоев должен быть выделен слой с лошадью и следует воспользоваться инструментом *Перемещение* из палитры инструментов.

Первоначально изменим размер нашей лошади, нажав комбинацию клавиш <Ctrl+T>; как вы видите, по периметру изображения появились маленькие квадратики, их перетягивание изменит размер изображения (для того чтобы сохранялись пропорции файла во время перетаскивания квадратиков, зажмите клавишу <Shift>).

Перетащим лошадь в нижний угол, и вы увидите, что и в палитре слоев произошли некоторые изменения: на иконке слоя с лошадью она тоже теперь находится в нижнем левом углу. То есть всё, что мы делаем в документе с нашими изображениями, обязательно отражается на слое в палитре слоев.

### 6. Дублирование слоя

Что ещё мы можем сделать с нашей лошадью в палитре слоёв? Конечно же, продублировать. То есть создать точно такой же слой с лошадью. Для этого мы должны:

или нажать правой кнопкой мыши на слой с лошадью в поле документа и из списка выбрать *Cosdamь dyбликат слоя*;

или в верхней панели опций выбрать: Слой / Дублирование слоя;

или же перетащить иконку слоя в нижний правый угол на иконку, похожую на квадратик с загнутым уголком.

И самый простой и чаще всего применяемый способ – это сочетание клавиш <Ctrl+J>. Теперь в нашей палитре слоёв три слоя: Фон, слой с лошадью («Слой 1») и только что созданный слой с лошадью, который фотошоп по умолчанию назвал «Слой 1 копия». Новый слой с лошадью автоматически был создан над выделенным слоем (по умолчанию фотошоп создаёт новый слой или дубликат слоя над выделенным слоем).

Итак, у нас в палитре слоёв три слоя, а в самом файле мы видим всего-то наш фон и лошадь, но это только потому, что Photoshop создаёт дубликат слоя строго над тем же слоем, что у нас был. Потяните инструментом *Перемещение* лошадь в сторону и вы это увидите.

#### 7. Переименование слоя

Слоёв в палитре становится все больше, и если при работе с файлом вы создаете много слоёв, то, чтобы не запутаться, можно слои переименовать: сделайте двойной клик по названию слоя в палитре слоёв и впишите любое описательное название. Допустим, первый слой с лошадью, который у нас по умолчанию называется «Слой 1», переименовываете в нижнюю лошадь, а продублированную лошадь «Слой 1 копия» в «Вторая лошадь».

#### 8. Разблокирование слоя

Заметьте, что при двойном клике на нижний слой «Фон» у нас всплывает диалоговое окно. Происходит это потому, что слой заблокирован (справа на слое стоит замочек). Заблокированный слой мы не можем двигать, применять к нему некоторые действия. Для разблокирования слоя дважды кликнете по слою в палитре слоёв, всплывёт диалоговое окно, в котором вы можете также поменять название слоя (например, на «Фоновка»), нажимайте *ОК*. Слой разблокируется, и изменится название слоя.

### 9. Создание нового пустого слоя

Чтобы создать новый пустой слой, нужно нажать на иконку в виде квадратика с загнутым листочком в основании палитры слоёв. Photoshop автоматически создаст слой над активным слоем («Вторая лошадь»). Или же в верхней панели опций нажать на *Слой / Новый слой*. (Опять же во всплывшем окне вы можете поменять название слоя и некоторые другие параметры.) Или же создать новый слой сочетанием клавиш <Shift+Ctrl+N>.

### 10. Удаление слоя

Чтобы удалить ненужный слой, правой кнопкой мыши нажимаем в палитре слоёв на слой, который хотим удалить, и из списка выбираем Удалить слой. Или в верхней панели опций выбираем Слой / Удалить слой. Или же в палитре слоёв мышкой нажимаем на слой и перетаскиваем в корзинку, которая находится в
основании палитры слоёв или же просто нажимаем на корзину. (Внимание! Слой, который мы хотим удалить, должен быть активным!) Слой удалится. Ещё один способ удаления слоя. При активном инструменте *Перемещение* выделяем слой, от которого хотим избавиться, и нажимаем клавишу <Delete>.

#### 11. Меняем слои местами

Слои в палитре слоёв можно менять местами. Перетаскивать наверх или вниз. Все эти манипуляции производятся мышью. Просто захватываете мышью нужный слой и перетаскиваете его вниз или вверх.

#### 12. Режимы смешивания

Рядом со кнопкой *Обычные* нажмите на стрелочку, выпадет меню со всем списком режимов смешивания. Суть режимов смешивания заключается в том, что пиксели верхнего слоя смешиваются с пикселями нижнего слоя, создавая порой оригинальные или необычные эффекты. По умолчанию всегда стоит режим смешивания *Обычные*, чтобы понять, как работают режимы смешивания, лучше с ними поэкспериментировать.

#### 13. Непрозрачность слоя

В этом же ряду находится *Непрозрачность* слоя. Понижая непрозрачность, мы понижаем непрозрачность слоя. Пример с лошадью: с понижением непрозрачности слоя лошадь становится полупрозрачной.

#### 14. Заливка

Опция Заливка выполняет практически то же, что и Непрозрачность: снижает непрозрачность слоя, но в отличие от Непрозрачности опция имеет одну замечательную особенность: снижая непрозрачность слоя, она оставляет видимыми все стили слоя, применяемые к изображению. Приведем пример: к лошади применим Стили слоя / Тень и Внешнее свечение. Для этого дважды щелкнем по иконке слой в палитре слоев и в открывшемся меню поставим галочки в разделе

*Тень* и *Внешнее свечение*. Верхняя лошадь имеет те же параметры, что и нижняя, только у верхнего слоя мы оставили Заливку 100%, а у нижнего понизили до 0%.

#### 15. Блокировка слоёв

Следующий ряд Закрепить. У нас их четыре.

При нажатии на первую блокировку (Квадрат) мы не сможем редактировать прозрачные пиксели, при нажатии на вторую блокировку (Кисточка) нельзя будет редактировать прозрачные и непрозрачные пиксели. Третий значок (Крестик со стрелками) заблокирует объекты слоя от перемещения, и четвёртый значок (Замок) заблокирует всё. Нельзя будет редактировать, двигать, удалять слой, но можно будет перемещать между слоями. Если вы выберите один из вариантов блокировки, то на выбранном слое справа появится значок замка.

#### 16. Связывание слоёв

Если вам нужно двигать несколько слоёв, чтобы они не сдвигались относительно друг друга, их можно связать. Выделите нужные слои в палитре слоёв. Если слои располагаются последовательно друг над другом, то, нажав на верхний слой и удерживая нажатой клавишу <Shift>, нажмите на нижний слой, который нужно связать. Все слои в этом промежутке выделятся синим цветом. (Если нужно связать слои, находящиеся не друг над другом, а в разных слоях палитры слоев, то нажмите на один слой, а потом, удерживая нажатой клавишу <Ctrl>, нажимайте на нужные слои). Далее нажмите на иконку, напоминающую скрепку в основании палитры, слои свяжутся. Вы это увидите, взглянув на палитру слоёв, – на связанных слоях появится значок связывания. Таким же образом вы можете развязать слои. Выделите связанные слои и нажмите на иконку связывания, слои развяжутся.

## 17. Стили слоя

Замечательная опция, применяемая к слоям. К любому из наших слоёв мы можем применить Стили слоя.

Для этого необходимо, конечно, выделить слой и в палитре слоёв нажать на иконку *Стили слоя*. Она находится в основании палитры слоёв fx. С помощью стилей слоя вы можете создать тень, внутреннее или внешнее свечение, поменять цвет и текстуру, применить градиент или создать рамку вокруг изображения. В списке вы видите весь перечень стилей слоя, которые можно применить к изображению.

#### 18. Маска слоя

С помощью этой иконки можно к слою применить *Маска слоя*. Маска слоя хороша тем, что позволяет исправлять ошибки, совершённые при удалении фона и прочих операциях.

Применим к слою *Маска*, иконка слоя маски находится в самом низу палитры слоев **Г**, далее выбираем инструмент *Кисть*, предварительно указав цвета переднего и заднего планов как черный и белый (кликнув по иконке **Г**). Далее указав основным цвет черный, можно удалять ненужные элементы изображения, при переключении цвета на белый можно восстанавливать удаленные фрагменты. Удалить Маску можно, перетянув на иконку с корзиной.

#### 19. Корректирующий слой

Корректирующий слой отвечает за изменение цвета, оттенков, насыщенности на изображении. Выберите слой, поверх которого хотите расположить корректирующий слой. Нажмите на иконку *Слой / Новый корректирующий слой*. Выберите из списка команду *Яркость / Контрастность*. Установите нужные вам параметры и нажмите *OK*. В любой момент вы можете выключить корректирующий слой или удалить, так как он создаётся в виде отдельного слоя. Основное изображение при этом останется неизменным.

### 20. Группирование слоёв

Когда у вас собралось достаточно много слоёв и переименование уже не помогает в них разобраться, можно слои сгруппировать в отдельные папочки. Для

создания папки нужно нажать на иконку в виде папки и перетащить туда нужные слои. Или выделить нужные слои в палитре слоёв и перетащить их на иконку с папкой. Либо выделить нужные слои, а затем нажать комбинацию клавиш

При создании папки вы можете дать ей название для вашего удобства, дважды щелкнув по имени группы. В палитре слоёв можно отключить видимость папки, соответственно невидимыми станут все слои, находящиеся в ней. Можно также перетаскивать папку, при этом перетаскиваться будут все слои. Слои, находящиеся в папке, можно разгруппировать, нажав правой кнопкой мыши по папке и выбрав из списка *Разгруппировать слои*, а также создать дубликат папки или же удалить папку со всем ее содержимым.

## 21. Слияние слоёв

Слить все видимые слои можно, нажав на сочетание клавиш <Ctrl+Shift+E> или нажав правой кнопкой мыши на слой и выбрав *Слить видимые слои*. Если вам нужно слить не все, а несколько слоёв, выделите их в палитре слоёв, нажмите правой копкой мыши по одному из слоёв и из списка выберите *Слить слои*.

# Практическая работа №5. Выделение областей. Инструменты выделения

Навыки выделения областей либо фрагментов имеют большое значение при работе с растровыми изображениями. Когда на изображении есть выделенная активная область, то редактировать можно лишь её, оставляя невыделенную часть изображения защищенной.

Выделение фрагментов в растровой графике принципиально отличается от такового в векторных редакторах. Для растровой графики не существует объектов, изображение формируется из пикселей разного цветового диапазона. Таким образом, перемещение фрагмента растровой графики первоначально требует сообщения для растровой программы, какую именно группу пикселей требуется

переместить – выделить ее. Границы выделенной области отмечаются мерцающей пунктирной дорожкой, которая формирует перемещаемую рамку (рис. 54).



Рис. 54. Выделенная и маскированная области

Изображение, находящиеся за зоной выделения, называется маскированной областью, и оно недоступно для редактирования или изменений, пока активна область выделения (рис. 55).



Рис. 55. Удаление фрагмента изображения: а) с выделением; б) при отсутствии выделения

Чем сложнее и разнообразнее будет изображение, с которым вам необходимо работать, тем более трудоемким будет процесс выделения необходимых областей, однако в Adobe Photoshop существует масса инструментов, облегчающих процедуру выделения. Познакомимся с ними подробнее.

Самый простой способ создать выделение в Adobe Photoshop – это воспользоваться режимом *Быстрой маски (Quick Mask)*. Для перехода из обычного режима рисования в режим *Quick Mask* необходимо нажать на кнопку III внизу палитры инструментов или воспользоваться «горячей» клавишей <Q>.

В режиме Quick Mask необходимо инструментом Карандаш или Кисть с жестким краем закрасить черным цветом (который в режиме Quick Mask/Быстрой маски будет представлен полупрозрачным красным) те элементы, которые необходимо удалить с изображения.

После этого следует вернуться в обычный режим рисования, нажав на кнопку внизу палитры инструментов, и инвертировать выделение, выбрав из меню *Выделить* команду *Инвертировать*. Инвертировать выделение необходимо, так как при работе в режиме *Quick Mask* в выделение попадает та часть изображения, которая не была закрашена. Можно, конечно, изначально в режиме быстрой маски закрашивать то, что на изображение не должно попасть в выделение, а ненужные объекты и дефекты оставлять незакрашенными, но, скорее всего, этот процесс будет более трудоемким (рис. 56).





Рис. 56. Выделение объектов при помощи Quick Mask

Кроме того, в Adobe Photoshop для выделения части изображения можно использовать любой из инструментов выделения: Прямоугольное выделение, Эллиптическое выделение, Лассо, Полигональное лассо, Магнитное лассо, Волшебная палочка.

Инструменты *Прямоугольное выделение* и *Эллиптическое выделение* на панели инструментов скрыты за одной пиктограммой. На панели инструментов отображается пиктограмма последнего выбранного инструмента. Для того чтобы открыть всплывающее меню, необходимо нажать правой кнопкой мыши на стрелке справа, внизу от данной пиктограммы (рис. 57).

Rectangular Marquee Tool м Elliptical Marquee Tool M Single Row Marquee Tool Single Column Marquee Tool

Рис. 57.Всплывающее меню инструмента

Инструмент *Прямоугольное выделение* предназначен для выделения прямоугольных или квадратных областей. Для того чтобы выделить прямоугольную область изображения, необходимо:

1. Активизировать инструмент выделения *«рямоугольное выделение»* нажав на панели инструментов пиктограмму . или (если последним выбранным инструментом выделения был не инструмент *Прямоугольное выделение*) выбрать инструмент из всплывающего меню.

2. Подвести курсор мыши в точку на изображении, где, как предполагается, будет находиться угол воображаемого прямоугольника, и нажать левую кнопку мыши.

3. Продолжая удерживать левую кнопку мыши нажатой, переместить курсор мыши по диагонали к противоположному углу и отпустить кнопку мыши.

Для того чтобы выделить квадратную область изображения, необходимо при выделении удерживать клавишу <Shift>. Следует учесть, что если уже имеется какая-то выбранная область, то при использовании клавиши <Shift> произойдет ее наращивание. Поэтому нужно нажимать <Shift> только тогда, когда уже начнется выделение области (рис. 58).



Рис. 58. Прямоугольное выделение

Инструмент Эллиптическое выделение предназначен для выделения эллипсов и окружностей. Для того чтоб выделить эллиптическую область на изображении, необходимо:

1. Выбрать на панели инструментов инструмент Эллиптическое выделение, нажав на пиктограмму , или (если последним выбранным инструментом выделения был не инструмент Эллиптическое выделение) выбрать инструмент из всплывающего меню.

2. Подвести курсор мыши в точку на изображении, где, как предполагается, будет находиться угол воображаемого прямоугольника, внутрь которого будет вписан эллипс, и нажать левую кнопку мыши.

3. Продолжая удерживать левую кнопку мыши нажатой, переместить курсор мыши по диагонали к противоположному углу и отпустить кнопку мыши.

Для того чтобы выделить окружность, необходимо при выделении удерживать клавишу <Shift>. Следует учесть, что если уже имеется какая-то выбранная область, то при использовании клавиши <Shift> произойдет ее наращивание. Поэтому следует нажимать <Shift> только тогда, когда уже начнется выделение области.

Если при выделении прямоугольной или эллиптической области на изображении удержать нажатой клавишу <Alt>, то контур будет образовываться от центра к границам, а не от одного угла к другому (рис. 59).



Рис. 59. Эллиптическое выделение

Инструменты *Лассо*, *Полигональное лассо*, *Магнитное лассо* на панели инструментов скрыты за одной пиктограммой. Инструмент *Лассо* предназначен для создания выделений произвольной формы. Для того чтобы выделить на изображении область произвольной формы, необходимо:

1. Выбрать на панели инструментов инструмент *Лассо*, нажав левой кнопкой мыши на пиктограмме *Р*, или выбрать инструмент из всплывающего меню.

2. Подвести курсор мыши к объекту, который необходимо выделить, и, нажав левую кнопку мыши, обвести объект и замкнуть контур.

Инструмент Полигональное лассо так же предназначен для создания выделений произвольной формы, но при этом контур выделения состоит из прямолинейных сегментов. Для того чтобы выделить на изображении область с помощью инструмента Полигональное лассо, необходимо:

1. Выбрать на панели инструментов инструмент *Полигональное лассо*, нажав на пиктограмму *У*, или выбрать инструмент из всплывающего меню.

2. Подвести курсор в любую точку рядом с объектом, который нужно выделить, и щелкнуть левой кнопкой мыши – это будет первая точка контура.

3. Переместить курсор мыши на следующую точку контура недалеко от первой и снова щелкнуть левой кнопкой мыши. Программа автоматически построит между двумя точками отрезок прямой.

4. Продолжая таким образом проставлять точки, обвести весь объект и замкнуть контур.

Инструмент *Магнитное лассо* предназначен для выделения областей произвольной формы. При использовании инструмента *Магнитное лассо* не требуется очень точно следовать контуру выделяемого объекта. Если объект имеет достаточно хороший контраст по отношению к окружающему его фону, то линия границы выделяемой области будет сама притягиваться к краю объекта. Для того чтобы выделить на изображении область с помощью инструмента *Магнитное лассо*, необходимо:

1. Выбрать на панели инструментов инструмент *Магнитное лассо*, нажав на пиктограмму , или выбрать инструмент из всплывающего меню.

2. Подвести курсор мыши к краю объекта, который необходимо выделить.

3. Кликнуть левой кнопкой мыши и начать медленно перемещать курсор вдоль линии планируемого контура выделения. Особое внимание следует уделять опорным точкам, появляющимся по мере формирования контура и при щелчках мышью, а если положение какой-либо опорной точки не соответствует запланированному, достаточно нажать клавишу <Delete>, чтобы удалить ее, вернуть курсор до положения предыдущей опорной точки и повторить его передвижение вдоль линии выделения.

4. Замкнуть контур, т.е. соединить конечную опорную точку с начальной, подведя к ней курсор или просто дважды щелкнув кнопкой мыши.

Инструмент Волшебная палочка предназначен для выделения области с или подобными по цвету пикселями. Параметры одинаковыми допуска устанавливаются с помощью параметра Допуск на панели опций данного инструмента. Чем больше величина этого параметра, тем больше цветов попадает в выделенную область. Значение допуска меняется от 0 до 255. При допуске, равном 0, выделяется область одного цвета, при допуске, равном 255, выделятся все цвета, т.е. все изображение. Для выделения области изображения, залитой одним цветом, необходимо:

1. Выбрать на панели инструментов инструмент *Волшебная палочка*, нажав на пиктограмме <u></u>

2. Подвести курсор мыши к той точке изображения, которая должна быть включена в область выделения, и щелкнуть левой кнопкой мыши. В результате вокруг точки появится контур выделения, ограничивающий область, цвет которой совпадает с цветом указанной точки с учетом допуска, заданного параметром *Допуск*.

Гибкость использования перечисленных инструментов обеспечивается тем, что выделяемые ими области можно складывать, вычитать друг из друга, получать пересечение двух нарисованных рамок.

Для того чтобы добавить новое выделение к старому, необходимо перед использованием инструмента нажать на клавиатуре клавишу <Shift> и, удерживая ее, осуществить выделение.

Для того чтобы вычесть новое выделение из старого, необходимо перед использованием инструмента нажать клавишу <Alt>. Совместное использование клавиш <Shift+Alt> приводит к тому, что выделенной остается область совмещения (наложения) старого и нового выделений (рис. 60).





Рис. 60. Варианты выделения областей инструментом *Волшебная палочка* с разным уровнем параметра *Доступ* 

После того как вы выделили объект, можно провести с ним разного рода трансформации: увеличение/уменьшение размеров, вращение, отражение и т.п. Для того чтобы провести процедуру трансформации, после выделения объекта нажмите комбинацию клавиш <Ctrl+T> (рис. 61).





Рис. 61. Возможности трансформации выделенных областей

Для трансформации выделенной области потяните за одну из точек на образовавшейся рамке. При непосредственном наведении курсора на точку и принятии им вида направленной в разные стороны стрелки можно изменять размер объекта. Чтобы строго соблюдать пропорции изменяемого объекта, в процессе изменения зажмите клавишу <Shift>, это обеспечит соблюдение пропорций ширины и высоты области. Трансформация размеров области с одновременно зажатой клавишей <Ctrl> обеспечит изменение перспективы объекта. Если чуть отодвинуть курсор от узловой точки, то курсор примет вид двоякоизогнутой стрелки. Зажатие левой клавиши мыши в этот момент и дальнейшее движение мышью обеспечат вращение объекта относительно его центральной точки. Для того чтобы снять с области рамку трансформирования, нажмите клавишу <Enter>. Более подробный перечень действий, связанных с трансформацией выделенной области (рис. 62).



Рис. 62. Все возможные способы трансформации выделенной области

Таким образом, вы познакомились с основными способами выделения областей изображения и процедурами их дальнейшей трансформации. Полученные навыки можно использовать для работы по созданию коллажей.

## Практическая работа №6.

## Основные приемы работы с изображениями в Adobe Photoshop

## 1. Создание коллажа

# Для этого вам понадобятся три основных изображения: Heбo.jpg, Луна.jpg и Птичка.jpg.

1. Откройте изображение неба (файл Небо.jpg) в Photoshop. Будем превращать его в лунную композицию.

2. Скопируйте этот слой. На панели *Окно* на единственном пока слое с небом кликните правой кнопкой и выберите *Создать дубликат слоя*.

3. Теперь нажимаем Изображение /Коррекция / Цветовой фон /Насыщенность и выставляем примерно такие настройки (ставим галочку *Тонирование*): оттенок: 196, насыщенность: 22, яркость: 0.

4. Открывается изображение с луной – Луна.jpg.

5. Теперь нужно вырезать её. Выбираем инструмент Волшебная палочка, кликаем по черному фону, а затем инвертируем выделение (Выделение» / Инверсия).

6. Копируем луну.

7. Перейдем обратно на изображение с небом. Для этого выберите пункт меню *Окно* – Небо.jpg. Вставим луну.

8. В случае необходимости слегка подправляем контуры луны. Находясь на слое с луной (!), нажмите комбинацию <Ctrl+T>. Измените размер луны.

9. Луна пока не вписывается в наш фон. Но это можно подправить. Зажимаем <Ctrl> и кликаем на маленькое изображение нашей луны в списке слоёв. Теперь при активном выделении нажимаем: *Выделение/Модификация/Граница*» и выбираем значение в *3px*.

10. Теперь надо слегка размыть луну: *Фильтры/Размытие/Размытие по Гауссу»*. Выставляем радиус в *1px*. Снимаем выделение <Ctrl+D>. Теперь границы луны слегка размыты.

11. Применим пару эффектов слоя с луной. Для этого два раза кликаем на наш слой с луной (делаем это в списке слоёв) и выставляем настройки:

• *Внешнее свечение*: Непрозрачность 43%, вместо желтого цвета ставим белый. Размах 7%, размер 111 рх.

• Внутреннее свечение: Непрозрачность 55%, вместо желтого белый цвет. Размах 1%, размер 103 рх. Для того чтобы убедиться, какие эффекты дают нам свечения, попробуйте «повыключать» эффекты, нажимая на «глазки» на панели Слои.

12. Открываем изображение с птичкой – Птица.jpg.

Отделить её от фона не составит особого труда, т.к. фон белый. Выбираем Волшебную палочку. На панели свойств выставляем Чувствительность (Допуск) на 50%. Выделение / Инверсия.

13. Копируем птицу.

14. Перейдем обратно на изображение с небом. Для этого выберите пункт меню *Окно* – Небо.jpg. Вставим птицу.

15. Теперь при активном выделении вокруг птицы, если вы его уже сняли, зажмите <Ctrl> и кликните на маленькую иконку изображение слоя с птичкой. Выбираем: *Выделение/Модификация/Граница* и выбираем значение в 2 *px*.

16. Теперь надо слегка размыть. Нажимаем: *Фильтры/Размытие/Размытие по Гауссу*. Выставляем радиус в 0,7 *рх*.

17. Заблокируем слой с птицей, нажав на кнопочку в шашечку на панели Слои.

18. Используя инструмент *Лассо*, сделаем грубое выделение левого крыла птицы.

19. Нажмём <Ctrl+J> (дубликат крыла на новый слой), а потом ещё раз (сделаем копию слоя с крылом). У нас должно появиться в списке слоёв два слоя с этим крылом. Самое верхнее мы скрываем. Для этого нажимаем на глазик слева от маленькой картинки слоя и переходим на слой с крылом, который находится ниже. Нажимаем: *Фильтр/Размытие/Размытие в движении*. Выставляем параметры. Угол – 20%, смещение: 200 рх.

20. Теперь перемещаем слой с нашим размытым крылом под слой с птицей и стиль смешивания слоя с *Нормального* на *Жесткий свет*.

21. Затем перемещаемся на самый верхний слой с крылом и делаем его видимым. Нажимаем: *Фильтр/Размытие/Размытие в движении*. Выставляем параметры. Угол – 80%, смещение: 50 рх.

22. Установите стиль смешивания этого слоя на Осветление.

23. Преобразовываем второе крыло. Убедитесь, что вы находитесь на слое с птичкой. С помощью лассо грубо выделяем второе крыло. Теперь нажимаем <Ctrl+J> (копируем слой со вторым крылом). Назовите этот слой «Крыло 2».

24. Копируем наш слой «Крыло 2». Копию «Крыло 2» помещаем под слой с птичкой.

25. Теперь будем вращать наше крыло. Для этого нажимаем <Ctrl+T>. Нужно повернуть крыло, чтобы получилось, как бы 2 крыла.

Не стоит отбрасывать крыло далеко от тела. Оно должно соприкасаться у основания. Примените трансформацию, нажмите <Enter>.

26. Теперь нажимаем: Фильтр/Размытие/Размытие в движении. Ставим в настройках: угол – 80 градусов и смещение: 85 рх. Меняем стиль слоя на Жесткий свет.

27. Если копию «Крыло 2» размещали при трансформации над крылом, то теперь «Крыло 2» нужно разместить чуть ниже. Для этого нажимаем <Ctrl+T>. Примените трансформацию, нажмите <Enter>.

28. После этого применяем: Фильтр/Размытие/Размытие в движении, только в этот раз ставим настройки 20 и 35 соответственно. Меняем стиль смешивания слоя на Осветление.

29. Теперь слегка подправим изображение. Переходим на слой с птицей и жмём: Изображение/Регулирование/Экспозиция и выставляем вот такие настройки [0.18 / -0.0020 / 1.08].

30. Остаётся только добавить текст и поиграть с настройками стиля слоя. Шрифт используйте *Arno Pro*, а текст напишите, какой вам больше нравится.

31. Сохраните рисунок в формате PSD и JPG с названием «Полнолуние».

## 2. Тоновая коррекция изображения и фильтры в Adobe Photoshop

Изображения можно получить разными путями, будь то сканирование, фотография с цифровой камеры и т.п. Зачастую такие изображения всегда имеют разного рода дефекты: затемненные или засветленные участки, отсутствие должного контраста или размытые и нечёткие детали. Качество изображения, и фотографии в частности, определяется грамотным соотношением света и тени. Если это соотношение соблюдено, то фотографии «оживают», в них чувствуется рельеф, повышается восприятие. Погрешности в освещении и контрасте называют тоновыми, а процедуру их корректировки – *тоновой коррекцией*.

Тон компьютерного изображения – это яркость пикселей, из которых оно состоит. Пиксели изображения имеют различную яркость. Та часть полного диапазона яркостей, которая использована в изображении, носит название *тонового диапазона*. Чем шире тоновый диапазон изображения, тем «глубже» цвета и лучше проработка деталей. В идеале в изображении должны быть использованы все значения яркостей (тона). Если разбить весь тоновый диапазон на три неравные условные части, то самая темная часть будет называться тенями, самая светлая – подсветкой, а между ними расположится большой диапазон средних тонов (рис. 63).



Рис. 63. Гистограмма изображения

По горизонтальной оси располагаются значения яркости от 0 до 255, а по вертикали – количество пикселей каждого уровня яркости. Тени – это самая темная часть изображения с малым количеством яркости. Светлые тона – самая светлая с наибольшим количеством яркости. Между тенями и светом располагаются средние тона. Основная задача тоновой коррекции – грамотное распределение яркости пикселей в изображении.

Для коррекции тона зачастую используют команды:

- Яркость/Контраст
- Уровни
- Кривые

## Яркость/Контраст

Команда *Яркость/Контраст* – это самый простой способ тоновой коррекции всего изображения. Подобная простота связана серьезными ограничениями, поскольку эта коррекция не позволяет осуществлять изменение яркости и контраста на выбранных тоновых диапазонах и отдельных цветовых каналах (рис. 64).

Яркость/Контрастность		×
Яркость:	이	ОК
Контрастность:		Отмена
		🔽 Просмотр
		🗌 Использовать прежние

Рис. 64. Яркость/Контраст

В диалоговом окне *Яркость/Контраст* есть всего две полосы, в которых перемещением ползунка изменяется яркость и контрастность в диапазоне от -100 до +100. Теоретически параметры яркости и контраста независимы, но на практике изменение одного из них влечет за собой изменение и другого. Увеличение или уменьшение яркости ведет к уменьшению контраста, увеличение контраста способствует увеличению яркости, а уменьшение контраста – к уменьшению яркости изображения.

## Уровни

В диалоговом окне *Уровни* представлена гистограмма распределения пикселей по градациям яркости (рис. 65).



Рис. 65. Уровни

По нижней оси гистограммы расположены три ползунка, отмечающие положение черного, белого и средне-серого цветов изображения. В правом и левом полях отображаются уровни белого и черного, измеряемые в градациях серого от 0 до 255. Среднее поле показывает гамму в условных единицах от 0 до 10. Исходная гамма изображения равна единице.Устанавливаемые значения тонов называются входными уровнями. Их можно задавать численно в полях *Входные значения*. При интерактивной настройке числа в этих полях соответственно меняются. Пипетки с различными наполнителями (чёрный, белый, и серый) служат для определения участков, которые вы хотите считать самыми тёмными, участков, которые вы хотите считать самыми светлыми, и полутонов (серая). Полутоновая пипетка принимает участие в коррекции цветных изображений.

Коррекция уровней позволяет изменять основные тоновые характеристики, настраивать диапазон тонов и менять положение нейтральной точки, т.е. гамму изображения. В окне *Уровни* удобно выполнять простейшие операции по коррекции контраста и яркости изображения как в средних тонах, так и в общем. Управляя выходными уровнями, можно регулировать наиболее яркий и темный тона изображения, предназначенного для печати. Важно помнить, что если изображение предполагается печатать, то необходимо учитывать недостатки воспроизведения светов и теней с помощью печатной машины.

## Кривые

С помощью диалогового окна *Кривые* можно осуществлять практически любую тоновую коррекцию, работа с этим окном позволяет вносить исправления даже в крайне дефектные изображения. Основную часть окна занимает график распределения пикселей. В отличие от гистограммы, представляющей абсолютные значения распределения, график кривых выстраивается относительно (рис. 66).



Рис. 66. Кривые

По оси X расположены градации яркости на входе – яркость пикселей в данный момент, а по оси Y – значения на выходе – яркость пикселей по результату коррекции. Состояние тонов при открытии окна изображается как линия под углом 45°, т.е. отсутствие коррекции. Стандартно для изображений в модели RGB яркость возрастает снизу-вверх и слева-направо – слева внизу расположены темные тона, а справа вверху – светлые тона.

Тоновая кривая должна неизбежно быть кривой хотя бы потому, что она так называется. На самом деле она сама не может быть ступенчатой, потому что ступеньки на кривой повлекут выпадение полутонов и плохое качество картинки. Режим создания кривой вручную позволяет нарисовать кривую от руки.

## Фильтры

Фильтры используются для очистки и ретуширования фотографий, создания специальных эффектов, которые придают изображению вид наброска или картины в особом стиле, а также специфических трансформаций с использованием эффектов

искажения и освещения. Все фильтры, предлагаемые компанией Adobe, содержатся в меню *Фильтр*.

Чтобы применить фильтр, выберите соответствующую команду в меню *Фильтр*. При этом следует помнить некоторые важные рекомендации:

• Фильтры применяются к активному, видимому слою или выделению.

- Некоторые фильтры работают только с RGB-изображениями.
- К 8-битным изображениям могут быть применены все фильтры.

• Некоторые фильтры полностью обрабатываются в оперативной памяти. Если объема доступной оперативной памяти недостаточно для обработки эффекта фильтра, то может быть выдано сообщение об ошибке.

Фильтр может применяться к активному слою или к смарт-объекту. Фильтры, применяемые к смарт-объекту, являются обратимыми и могут быть в любой момент перенастроены.

• Чтобы фильтр мог бы использоваться ко всему слою, необходимо сделать этот слой активным или выделенным.

• Чтобы применить фильтр к области слоя, следует выделить эту область (рис. 67).



a



б

Рис. 67. Варианты изменения изображения после применения фильтра: а) изображение без фильтра; б) изображение с фильтром «Мозаичные фрагменты»

Эффекты фильтров используются в том порядке, в каком они выбраны. Можно переупорядочить фильтры после их применения, перетаскивая их имена в списке примененных фильтров. В результате переупорядочения эффектов фильтров вид изображения может кардинально измениться. Применяемые фильтры можно также удалять, выбрав фильтр и щелкнув значок *Удалить слой*.

Команда *Ослабить* позволяет изменить непрозрачность и режим наложения любого фильтра, инструмента рисования, стирания и цветовой коррекции. Режимы наложения команды *Ослабить* представляют собой подмножество соответствующих режимов, задаваемых в параметрах инструментов рисования и редактирования.

### Практическая работа № 6.

## Исправление дефектов на фотографии, ретушь лица

Для качественного освоения техники ретуши требуется значительное время. Обязательно учитывайте базовые советы при процедуре ретуши:

• Сохраняйте как можно больше текстуры кожи. Для этого пользуйтесь маленькими кисточками (с небольшим радиусом) с высокой мягкостью.

• Не рассматривайте природные пятна кожи, например, веснушки и родинки, как недостатки и дефекты. Они придают модели особенность и изюминку.

• Во всем надо знать меру и понимать, когда следует остановиться.

1. Откройте исходное изображение (фото должно быть в хорошем качестве, с разрешением не менее 1000х1000 пикселей), и сделайте копию фонового слоя. На этом корректирующем слое-копии мы и будем работать. Слой «Фон» не изменяем – с ним мы сможем сравнить получившееся изображение «до» и «после».

2. Удаляем мелкие дефекты: прыщики, пигментные пятина и т.п. Используем инструмент *Точечная восстанавливающая кисть*, задаем ей размер и жесткость, аккуратно удаляем дефекты точечными движениями. Помните: любой из инструментов *«Кисть+*Alt» переводит его в режим *Мишень* для взятия образца

цвета с целью закраски редактируемой области, а *Точечная восстанавливающая* кисть автоматически настраивается на окружающий дефект фон.

3. Работа с глазами. Выберите инструмент *Осветлитель*, задайте ему размер, примерно равный радиусу радужки, и экспонирование (интенсивность осветления) примерно 50% (можно попробовать разные его значения для достижения желаемого эффекта).

4. Измените цвет глаз. Для этого используем инструменты выделения:

• выберите инструмент *Лассо*, при нажатии клавиши <CapsLock> пиктограмма *Лассо* сменится на «мишень» для удобства работы с мелкими деталями. Выделите инструментом *Лассо* радужку глаза по внешнему контуру, исключите из области выделения зрачок, выделив его с помощью <*Лассо*+Alt>;

• добавьте к выделяемой области радужку другого глаза и так же исключите из него зрачок;

• сгладьте область выделения, чтобы избежать неровностей выделения и грубых линий: меню *Выделение/Модификация/Растушевка* радиус растушевки в зависимости от амплитуды неровностей;

• для изменения цвета выделенной области накладываем новый корректирующий слой (меню Слои / Новый корректирующий слой /Цветовой баланс), задаем ему имя, например, «Глаза»;

• с помощью ползунков в меню Цветового баланса выберите нужный оттенок, но следите за естественностью цветов.

5. Губы. Задайте оттенок губам, используя инструменты и методы аналогично тому, как мы делали это с глазами. Вновь следите за естественностью выбранного оттенка. Попробуйте изменить оттенок губ с помощью другого режима корректирующих слоев, например, *Тон / Насыщенность*.

6. Удаление лишних деталей. Для удаления лишних деталей, например, волосков, можно воспользоваться инструментом Штамп.

7. Избавление от дефектов кожи. Одной из основ ретуши является избавление от дефектов кожи. Именно об этом и пойдет речь дальше. Воспользуемся быстрой техникой «Degrunge».

8. Техника «Degrunge» выполняется в несколько этапов:

Этап 1. Разделение на части

• Объедините все слои, кроме фона.

• Продублируйте полученный слой дважды.

• Первый назовите «Размытие», второй «Мелкие детали», третий «Неровности».

1) Работа со слоем «Размытие»

• На первой копии слоя примените фильтр *Размытие по Гауссу*. Радиус увеличиваем до тех пор, пока неровности и дефекты не исчезнут. Будьте внимательны, этот шаг очень важен! Подбирайте радиус правильно.

• Радиус будет равен примерно 6, он подбирается в зависимости от размера изображения в пикселях. Запомните значение радиуса, оно вам понадобится в дальнейшем шаге.

2) Работа со слоем «Мелкие детали»

• На второй копии слоя применяем фильтр Цветовой контраст (Фильтр/Другое/Цветовой Контраст) и подбираем такой радиус, чтобы мелкие детали были видны, но неровности еще не проявились. Это не всегда легко, но в первом приближении можно разделить радиус размытия на три, т.е. около 1.

3) Работа со слоем «Неровности»

• На третьей копии нужно применить фильтр Цветовой контраст с радиусом, который мы использовали для размытия, т.е. 6, затем размыть полученный слой по Гауссу с радиусом, который применяли для фильтра Цветовой контраст на слое «Мелкие детали», т.е. 1.

Этап 2. Применение эффекта «Векторной маски»

1. Теперь организуем слои. Расположите слой *Размытие* над фоновым, выше него слой «Неровности», верхний слой будет «Мелкие

детали». Для слоев «Неровности» и «Мелкие детали» установите режим смешивания *Линейный свет* и непрозрачность 50%.

2. Мы снова получили оригинальное изображение.

3. Выделите слой «Неровности» и добавьте к нему *Векторную маску*, появится изображение маски напротив значка слоя.

4. Выберите инструмент *Кисть*, установите для него размер, равный самому узкому месту на корректируемом изображении, минимальную жесткость и непрозрачность 50%.

5. На панели цветов задайте основной черный цвет, второстепенный белый.

6. Мягкой черной кистью закрашивайте места, где хотите устранить неровности, но держитесь подальше от границ и деталей, которые должны остаться четкими: глаз, бровей, губ, теней на лице в области носа и т.д.

7. После достижения желаемого результата объедините корректирующие слои в группу *Ретушь*.

8. Теперь можно сравнить полученный результат с исходным изображением.

# Практическая работа № 7. Инструмент «Текст». Работа с текстом

Всем нам хорошо знаком такой инструмент, как *Текст*. Как только ПК прочно вошли в повседневную жизнь, большая часть деятельности, связанной с компьютером, проводилась именно благодаря этому инструменту.

Вы знакомы с ним ещё с таких программ, как «Блокнот», «MS Word», «OpenOffice» и мн. др. Не обошёл текст стороной и Adobe Photoshop. Стоит понимать, что инструмент *Текст* является одной из наиболее сильных и разработанных сторон ПО Adobe Photoshop. Однако работа с текстом не подразумевает только печать символов, возможности редактора Photoshop

значительно шире, и для грамотной деятельности следует внимательнее рассмотреть данный инструмент.

Клавиша инструмента *Текст* располагается на панели инструментов Adobe Photoshop и активируется однократным кликом по пиктограмме **Т**. либо при помощи клавиши <T>. Обратите внимание, что панель инструмента текст включает в себя вложенное меню, где располагаются ещё несколько режимов с текстовыми данными (рис. 68).



Рис. 68. Режимы инструмента Текст

• *Горизонтальный текст* – стандартный режим текста, располагающегося по горизонтали.

• *Вертикальный текст* – размещение текста по вертикальной оси (используется, например, при написании фраз на некоторых восточных языках).

• *Горизонтальный текст* – маска – создание быстрой маски, написание горизонтальным выделением.

• *Вертикальный текст* – маска – написание вертикальным выделением.

Текст можно расположить в необходимой области при помощи двух различных способов:

1. Выбираем инструмент *Текст* и кликаем по нужной нам области в пределах рабочего пространства. После появления мерцающего курсора можно вводить текст (вводить можно как с клавиатуры, так и копировать из

других источников). После окончания ввода нажимаем <Ctrl+Enter> и наш текст будет готов.

2. Размещаем наш текст внутри рамки выделения. Выделяем область, включаем инструмент *Текст* и располагаем его внутри выделенной области, а затем уже вводим необходимые нам данные. Принципиальное отличие от первого варианта заключается в том, что текст не может выйти за пределы выделенного пространства, что очень удобно при необходимости точного размещения текстовых данных.

Перемещение и трансформация текста обеспечивается точно таким же способом, что описан при перемещении либо трансформации выделенных областей, и активируется нажатием горячих клавиш <Ctrl +T>.

## 1. Форматирование текста

Форматирование текста обеспечивается благодаря параметрам меню *Символ*. Чтобы перейти в это меню, выберите *Окно/Символ* (рис. 69).



Рис. 69. Форматирование текста

Данное меню содержит две вкладки: *Символ* и *Абзац*. Рассмотрим более подробно инструменты первой вкладки, отвечающие за работу с символами (табл. 4).

Иконка	Работа с символом
Tahoma 🗸	Позволяет выбрать гарнитуру шрифта
Regular 🗸	Начертание шрифта
<b>T</b>	Размер шрифта
IA	Расстояние между строками текста
АŢV	Расстояние между символами внутри заданной пары
₽¥.	Межзнаковое расстояние
IT	Масштабирование текста по вертикали
Ĩ	Масштабирование текста по горизонтали
<u>A</u> <sup>a</sup> <sub>t</sub>	Позволяет регулировать положение символов относительно строки
Цвет:	Выбирает цвет шрифта
T T TT Tr T <sup>1</sup> T <sub>1</sub> T	Панель выбора различных стилей шрифта: жирный, курсив, подстрочный, зачеркнуты и т.п
аа резкое ~	Сглаживание текста

Вторая вкладка включает в себя следующие инструменты (табл. 5)

	Панель позволяет выбирать
	различные варианты выравнивания:
	по левому краю, по центру, по
	ширине и т.п.
+≣ 0 пт	Отступ от левого края
重 + Опт	Отступ от правого края
*Ē 0 пт	Отступ первой строки позволяет
	оформлять «красную строку»
т 0 пт	Отступ перед абзацем
,≣ 0 пт	Отступ после абзаца

## 2. Деформация текста

Данный инструмент позволяет провести различную деформацию текста. Для активации инструмента кликните правой кнопкой мыши по тексту с активированным инструментом *Текст*, из выпадающего меню выберите пункт *Деформировать текст* (рис. 70).

Деформировать текст	×
Стиль: Не показывать 💌	ОК
🕼 Горизонтальный 🏾 С Вертикальный	Отмена
Изгиб; 🥼 %	
Искажение по горизонтали:  %	
Искажение по вертикали; 🥼 %	

Рис. 70. Деформирование текста

В раскрывающейся вкладке *Стиль* существует большое количество различных вариантов деформирования вашего текста. Регулирование ползунков в пунктах *Изгиб*, *Искажение по горизонтали*, *Искажение по вертикали* с выбранным стилем деформации позволит значительно преобразовать ваш текст (рис. 71).

- Hotoshop	genteo10th
Деформароваль техст X   Сливь: Спань: Спань: С   Ф Горовонгальный С Отмена Отмена   Ислажение по горовонгали: 440 % Ус   Ислажение по горовонгали: 440 % Ус	Асформераналь гекст Стиль: [2]Закрученный У Истиль: [2]Закрученный И Истиль: 555 % Иссажение по горяоснатался ( Иссажение по вертикалия: ( 177 % Иссажение по вертикалия: ( 177 %

Рис. 71. Некоторые варианты деформации текста

## 3. Текст по контуру

Одной из интересных возможностей Adobe Photoshop является возможность размещения текста по контуру. Создайте любую кривую при помощи инструмента *Перо* либо *Свободное перо* . Подведите курсор к контуру, указатель изменит форму, кликните левой кнопкой мыши и введите необходимый текст (рис. 72).

voto shop

Рис. 72. Вариант отображения текста по контуру

Преобразуйте в фигуру. Данная возможность позволяет преобразовать текст в векторную маску слоя, которую можно редактировать, подобно любым другим векторным фигурам. Создайте текст, затем перейдите в панель *Слои/Шрифты/Преобразовать в кривые*. Теперь если кликнуть по букве и потянуть за опорные точки, то можно изменить размер контура буквы (рис. 73).



Рис. 73. Преобразование в фигуру

# Практическое задание № 8. Работа с текстом. Текст из травы

Урок состоит из трех частей – фон, текст и немного дополнительных эффектов. Вам потребуются несколько изображений «Бумага.jpg», «Трава.jpg» и «Бабочка.jpg».

## 1. Создание фона

1. Создаем новый документ в Photoshop размером 1920 на 1200 рх.

Приступим к рисованию радиального градиента с помощью инструмента *Градиент* (клавиша <G>) начиная от светло-зеленого (#ADBF41) и заканчивая средне-зеленым (#328A26).

2. Сделаем этот фон более реальным, используя для этого готовую текстуру бумаги, открыв изображение «Бумага.jpg» в Photoshop. Теперь нужно сделать текстуру черно-белой <Ctrl+Shift+U> и растянуть ее так, чтобы она полностью закрывала наш документ.

3. Далее переводим этот слой в режим *Перекрытия* и ставим прозрачность на 70%, чтобы смешать текстуру с нашим зеленым фоном.

4. Чтобы сделать текстуру еще лучше, копируем слой с текстурой, поворачиваем его на 180 градусов и делаем его прозрачным на 20%. Далее берем еще текстуру бумаги и проделываем с ним то же самое, размещая каждый следующий слой над предыдущим. Все это нужно делать для того, чтобы текстура смотрелась более реалистичной и появились мелкие детали. Как можно видеть, получилось шесть слоев.

5. Теперь необходимо скопировать самый первый зеленый фон и разместить копию над всеми слоями, после этого поставить прозрачность на 40%.

6. Создаем новый слой, который будет выше всех. Используя большую мягкую черную кисть (клавиша <B>), добавляем немного темного цвета по краям картинки. Стоит вернуть прозрачность на 30% и перевести в режим *Перекрытия*. Затем можно скопировать этот слой и применить к копии *Размытие по Гауссу* при 32 рх. В результате углы станут более мягкими.

## 2. Сохдание текста

7. На панели слоев создаем новую группу слоев (щелчком на папке на панели слоев) с именем *Текст*. На панели инструментов Photoshop выбираем инструмент *Текст*, задаем на панели опций гарнитуру и размер шрифта и пишем какое-нибудь слово.

8. А сейчас используем картинку с травой для создания травяного эффекта. Открываем «Трава.jpg». Трава красивая, но в верхней части немного темнее, чем в середине. Нам нужен большой прямоугольный кусок этой травы для создания травяной текстуры. Используя инструмент *Прямоугольник*, выделяем часть изображения с самой красивой травой, копируем <Ctrl+C> и вставляем <Ctrl+V> в документ с текстом (новый слой с травой должен быть над текстовым слоем). Переименовываем слой в «Текстура травы».

9. Вы видите, что размер текста и травяной текстуры не совпадают. Чтобы получить хороший результат, нужно использовать для изображения травы режим *Свободное трансформирование* (<Ctrl+T>), увеличить или уменьшить размер текстуры, а затем продублировать этот слой и сдвинуть в слое-дубликате траву вправо, чтобы закрыть весь текст. Если необходимо, повторяйте это до тех пор, пока текст не будет закрыт полностью. После этого нужно объединить все слои с травяной текстурой в один слой.

10. Слой с травой нужно сделать невидимым, а затем создать новый слой над текстовым.

11. Теперь создадим траву, при помощи которой будем украшать текст. В Photoshop уже есть кисть травы, но она не подходит для создания текста. Поэтому мы создадим новую кисть. Берем инструмент *Карандаш* и рисуем травинку, заливаем ее черным цветом и сохраняем как кисть: Редактирование//*Определить кисть*, открываем окно выбора кистей и выбираем эту кисть. Затем открываем окно *Кисти* (или нажимаем F5) и задаем настройки для кисти:

 Динамика формы: колебания размера – 100%, минимальный диаметр – 22% и колебание угла – 100%;

• Рассеивания: рассеивание – 45, счетчик – 5.

• Размеры и жесткость кисти используйте по своему усмотрению.

12. Выбираем текстовый слой и растрируем его. Выбираем инструмент Волшебная палочка, выделяем наш текст, затем сочетанием клавиш <Ctrl+Shift+I> инвертируем выделение. Переходим на чистый слой и проводим нашей травяной кистью по контуру букв, создавая по периметру текстуру травы. Затем выбираем слой с нарисованной по контуру травой и объединяем их. После этого делаем видимым слой с травяной текстурой, делаем его активным и с помощью контекстного меню выбираем *Создать обтравочную маску*. Если вас устраивает полученный эффект, объединяем два слоя – текст и травяную текстуру.

13. Можно остановиться на этом, а можно еще сделать следующее: в стилях слоя задать тень для травяного текста. Выбираем *Стили слоя/Тень* и задаем для нее такие параметры:

- Режим наложения (Умножение),
- Угол (120),
- Размах (20),
- Размер (8).

### 3. Создание дополнительных эффектов

14. Добавим элементы на нашу картинку с текстом. Для начала открываем изображение «Бабочка.jpg» в Photoshop, вырезаем белый фон с помощью инструмента Волшебная палочка (клавиша <W>), затем выполняем Выделить/Модифицировать/Сжать и выполняем сжатие на 1px. Инвертируем выделение <Ctrl+Shift+I> и копируем бабочку в наш главный рисунок. Для вырезания божьей коровки из файла «Божья коровка.jpg» используем инструмент Лассо, обводим нужную часть и копируем в главный рисунок.

15. Теперь уменьшаем бабочку и божью коровку пропорционально, добавляем к ним тень и расставляем так, чтобы они уравновешивали друг друга на рисунке.

## Глава 4. Система статистического анализа и визуализации данных R

## НАЧАЛО РАБОТЫ С RSTUDIO. ОРГАНИЗАЦИЯ РАБОЧЕЙ СРЕДЫ

## Рабочее пространство

Рабочее пространство – это текущая рабочая среда R в памяти вашего компьютера, которая включает в себя любые созданные пользователем объекты (векторы, матрицы, функции, таблицы данных или списки). В конце каждой сессии вы можете сохранить рабочее пространство, и оно автоматически загрузится при следующем запуске программы. Команды интерактивно вводятся в ответ на приглашение к их вводу. Можно использовать стрелки вверх и вниз для перемещения между введенными ранее командами. Это позволяет вызвать предыдущую команду, отредактировать ее и вновь выполнить ее, нажав клавишу Enter.

Текущая рабочая директория – это та директория, где находятся файлы данных и куда по умолчанию сохраняются результаты. Функция getwd() позволяет узнать, какая директория в данный момент является рабочей. Вы можете назначить setwd( рабочую директорию при помощи функции ). Если появляется необходимость импортировать файл, который находится не в рабочей директории, нужно написать полный путь к нему. Всегда заключайте в кавычки названия файлов и директорий. R – это чувствительный к регистру клавиатуры язык. Вы можете либо вводить команды по одной – приглашение на ввод команды (>), либо запускать набор команд из исходного файла.

При открытии RStudio вы увидите четыре рабочих окна. Нижнее левое – это консоль (*Console*), туда можно непосредственно вводить команды, нажимать <Enter> и получать результат (рис. 74).

Над консолью тоже можно писать команды и коды целиком. <Enter> здесь только переносит на нижнюю строку. Для запуска кода необходимо нажать *Run*. Справа вверху находятся две вкладки *Cpeda* (*Environment*) и *История*: (*History*).



Рис. 74. Рабочие окна RStudio. Начало работы

Среда показывает наше окружение, а История сохраняет все команды, использованные в течение одной рабочей сессии. Внизу справа находится несколько вкладок. *Файлы (Files)* – это менеджер файлов, который позволит нам пользоваться переходами, не покидая RStudio. *Графики (Plots)* предназначены для визуализации. *Пакеты (Packages)* – список библиотек, которые мы используем. *Помощь (Help)* – встроенная справка.

Если необходимо воспользоваться справкой или найти ответ на какой-либо вопрос, то можно выбрать два варианта действий:

использовать встроенную справку: help ("sin") или help.search("logarithm");

использовать электронные ресурсы, например, *Kickstarting R*, https://cran.rproject.org/doc/contrib/Lemon-kickstart/index.html или *R-analytics*, http://ranalytics.blogspot.ru/.

Как пользоваться справкой:

- 1. Открыть RStudio.
- 2. В правом нижнем секторе перейти во вкладку *Help*.

3. Ввести в секторе *Консоль* (*Console*, левый нижний сектор рабочего пространства) запрос: *help* ('\*\*\*'), или *help* ("\*\*\*"), или ?\*\*\*, или ?'\*\*\*'

Справка появится в правом нижнем секторе.

R спроектирован как расширяемый язык, основанный на пакетах (*packages*). Даже само ядро языка, с которым мы будем работать, является пакетом *base*. Вместе с основным пакетом происходит автоматическая установка еще нескольких пакетов, часть из них уже подключена к сессии, а другие можно подключать без необходимости установки. Остальные специализированные пакеты можно и нужно скачивать с официального сайта CRAN. Кроме того, можно пользоваться *github* и *bitbucket*, эти ресурсы не требуют строгой документации к пакетам. Некоторые пакеты есть только на них.

Чем отличаются пакеты от библиотек? Модули расширений, которые необходимо скачивать, – это пакеты. А место на диске, где они хранятся, – это библиотеки. Можно посмотреть список библиотек, набрав команду:

*libPaths*() – в результате мы получим адреса на диске, где хранятся эти пакеты.

installed.packages() покажет пакеты, которые у вас установлены.

Как подключить установленный пакет? Используем два вида команд:

library(название пакета) или require (название пакета).

После подключения будут доступны дополнительные функции.

Иногда необходимо скачать новый пакет и установить его на компьютер. Для установления используется функция *install.packages('название nakema', dependencies* = *TRUE*). Второе условие показывает, что автоматически загружаются и другие пакеты, которые нужны для нормальной работы запрашиваемого. Визуализация зависимостей пакетов друг от друга доступна по адресу: http://nycdatascience.com/most-popular-r-packages-and-r-package-dependency-visualization/

Если необходимо обновить пакеты, то нужно использовать команду update.packages ().

Наконец, некоторые пакеты могут работать неправильно (или конфликтовать друг с другом), особенно если перед этим произошло обновление версии R. Проверка осуществляется командой *sessionInfo*(), которая показывает все установленные у вас пакеты.

#### Сохранение результатов

Чаще всего, работа с кодом не исчерпывается одной сессией. Работу можно сохранить разными способами.

• Через верхнее меню команда *Save File*. Появится окно, где можно выбрать директорию, где будет храниться скрипт, и самостоятельно набрать название.

• Иногда нужно сохранить не только скрипт, но и некоторые исходные данные вместе с ним. Для этого используют функцию *write.csv*(df, 'df.csv') – при реализации этой команды через консоль появится файл в формате .csv

• Иногда нужно сохранить некоторые дополнительные переменные и функции, расчет которых заново потребует много времени. Для этого есть команда *Save Workspace* во вкладке Среда (Environment). В этом случае сохраняется все содержимое окна Среда (Environment).

• Если нужно сохранить только некоторые переменные, а не всю рабочую среду, то используем команду в консоли *save*(my\_file, file = 'my\_file.RData').

## Переменные и функции

Итак, любые команды в R исполняются построчно. Набирается команда и нажимается <Enter>.

Если код состоит из нескольких строк, то каждая из строк исполняется отдельным интерпретатором. При одновременном выполнении нескольких строк необходимо объединять их.
# Практическая работа №8. Переменные, операторы, функции

#### 1. Упражнение «Калькулятор»

Ввести команду: 18+4 (или любую другую) и нажать <Enter>

Намного эффективнее использовать переменные. Используем оператор присваивания «<-». Слева от оператора указывается имя переменной (может быть любым, уникальным), справа – ее значение. Так можно создать любые переменные. Типы данных очень разнообразны: вектора, матрицы, таблицы данных и списки.

#### 2. Упражнение «Переменные»

- a<- 5 b<- 8 c<- a+b
- c

Можно создать два типа переменных в зависимости от окружения. Окружение переменной – это пространство, где ее можно найти. Все переменные, созданные в консоли, попадают в глобальное окружение. Переменные из глобального окружения доступны для любых программ во вкладке *Среда (Environment)*. Локальные переменные существуют в текущей области, их значение может быть использовано как параметр программы, но напрямую любая другая запущенная программа к ним обратиться не может. Локальные переменные утрачиваются после выполнения текущей программы.

При работе с фрагментами сложного кода, более одной строки, лучше перейти к вводу команд в левой верхней вкладке. При этом будет удобно отслеживать историю их использования. Кроме того, можно будет повторно запускать некоторые команды и редактировать их. Для запуска команды необходимо нажать Run (правый верхний угол в левой верхней вкладке).

#### 3. Упражнение «Логические операторы»

a> 3 TRUE a > 8 FALSE a == 5

TRUE

Обратите внимание, что один знак равенства соответствует присвоению переменной какого-либо значения, а два знака – это оператор сравнения, т. е. логический оператор.

Встроенные функции уже содержатся в R. Можно использовать любые из них в соответствующей ситуации.

#### 4. Упражение «Функция»

- 1) ls () позволяет посмотреть глобальное окружение.
- 2) rnorm (15, mean = 5, sd = 2)

Обратите внимание, если необходимо присвоить значение переменной, то используется оператор присваивания, если нужно указать значение аргумента функции, то – оператор равенства. Можно также написать свою собственную функцию. Выбираем для своей функции какое-либо новое название, затем в круглых скобках указываем аргументы функции, а в фигурных – тело функции (ее код).

Указываем название новой функции, затем оператор присваивания, затем ключевое слово function, которое подтверждает то, что создается именно функция, а не обычная переменная. В данном примере аргументов функции не существует, поэтому круглые скобки остаются пустыми. Код здесь представляет собой переменную у в локальном окружении с постоянным значением 2. Затем идет команда return, которая возвращает значение у.

### 5. Упражнение «Пишем собственную функцию»

```
returntwo <- function ( ) {
y<-2
return (y)
}
returntwo ()
При запуске такой функции будет напечатана цифра 2.
Или:
addten<- function(x) {
x <- x+10
}
```

```
addten (a)
```

а

Значение переменной **a** не изменилось. Это связано с тем, что присвоение нового значения произошло только в локальном окружении, а в глобальном оно не изменилось. Для того чтобы изменения отразились в глобальном окружении, необходимо использовать команду *return*.

```
addten<- function(x) {
return ( x+10)
}
morea <- addten (a)
morea
или
<<- оператор глобального присваивания.
addten <- function (x) {
moreb <<- x+10
}
addten (b)
b
moreb
```

#### 6. Удаление лишних переменных

Удалить лишние переменные можно функцией *гт*. Например:

rm (moreb)

Восстановить переменные, удаленные с помощью функции гт, невозможно.

#### 7. «Напишите функцию»

Напишите функцию, которая принимает на вход два числа и возвращает результат деления первого на второе:

```
divide <- function x, y) {
[ваш код]
}
Проверьте результат:
divide (164, 4)
```

#### Практическая работа №9. Векторы и матрицы

Базовой структурой данных в R является вектор. На векторах основаны более сложные структуры: матрица, список, фактор, датафрейм. *Векторизация* – это ключевая концепция языка. Вектор представляет собой индексированный набор данных одного типа. То есть каждый элемент имеет свой номер, и все элементы должны быть сходны между собой. R не делает различия между скалярными и векторными величинами, т. е. скаляр здесь – это вектор длины 1.

Порядок индексации в этом языке начинается с 1, поэтому к последнему элементу вектора можно обратиться по индексу n, a не n-1, как во многих других языках программирования.

Так как элементы вектора должны быть одного типа, то вектора называются атомарными (*atomic*). Выделяют следующие типы векторов:

- *Logical* Логические (TRUE/FALSE).
- *Integer* Целые (целые числа).
- *Numeric/double* (числа с плавающей точкой).
- *Complex* (комплексные числа).

- *Character* (строки).
- *Raw* (байтовые последовательности).

Для создания вектора можно использовать функцию с:

x < -c(5, 8)

Создание числовых последовательностей возможно с помощью оператора «:», который задает последовательность с шагом «1» с определенными границами:

x <- c (5:8)

Тип вектора можно определить при помощи функции typeof:

```
tree<-c("oak", "birch"," pine")
typeof (tree)
```

Можно включить в вектор элементы разных типов, например: dou<- c (FALSE, 1.5) typeof (dou)

Мы получим тип *double*, это значит, что элемент FALSE был автоматически приведен к 0.

Если изменить вектор *dou* так, чтобы в нем преобладал тип данных строки, то меньшее количество элементов другого типа будет приведено к типу большинства, т. е. число 5 превратится в символ "5".

dou <- c(5, b, "abc")

В некоторых случаях элемент не приводится к новому типу, а просто не распознается. В этом случае на его месте появляется символ NA.

Можно задавать шаг последовательности:

seq (1, 2, by = 0.25)seq (3, 4, length.out = 5)

Для повторения существующих векторов используется фунция *rep*:

```
rep (1:3, times = 3)
rep (1:3, each = 4)
rep (1:3, length.out = 5)
```

Для обращения к определенным элементам вектора, необходимо указать индекс элемента. Например:

x[1] x[3] x[c(1,3,4)]

#### 1. Упражнение «Вектор»

Сохраните в новой переменной вектор из 20 целых чисел с шагом 2.0.

Выведите на экран с помощью команды print 2, 5, 7, 9 и 17 элементы вектора.

При операциях над векторами эта операция проводится над каждым элементом вектора. Можно выполнять, например, любые арифметические операции. Также можно выполнять логические операции, например:

x >= 3

В качестве ответа будет получен вектор из FALSE и TRUE. Операции сравнения можно использовать и при индексации. Например, возьмем вектор из 20 элементов, который вы создали при выполнении упражнения. Назовем его vector\_1.

vector\_1[vector\_1>=10]

Обратите внимание, что выводятся только те элементы, для которых это неравенство выполняется со значением TRUE.

vector\_1[vector\_1>=10& vector\_1 < 30]

#### 2. Сохраните результат в новую переменную

Правила переписывания (*recycling*). Иногда мы работаем с векторами разной длины. Есть несколько правил работы с ними.

• Длина результата равна длине большего из векторов.

• Меньший вектор дублируется (переписывается) несколько раз, чтобы длина переписанного вектора совпала с длиной большего.

• Если длина большего вектора не делится нацело на длину меньшего, то выдается предупреждение (не ошибка).

Сложим два вектора: 1:5 + 0:1. Второй вектор состоит всего из двух, поэтому появится предупреждение. Чаще всего вектором меньшей длины является скаляр (вектор из одного элемента). Приоритет операторов при этом сохраняется: 5:8<sup>2</sup>; *это не то же самое, что* (5:8)<sup>2</sup>.

Если необходимо совершить операцию с каким-то одним элементом вектора (или нескольких элементов), то порядковый номер этих элементов указывается в квадратных скобках после названия вектора: f[4, 8]. Отрицательное индексирование возвращает все элементы, кроме указанных: f[-5].

Логическое индексирование – все элементы, которые соответствуют значению TRUE:

*f*[*rep*(c(TRUE, FALSE), 5)]

При помощи логического индексирования удобно выбирать элементы, которые соответствуют определенным условиям.

f[f<18 &f>5]

Функция *length* показывает атрибут вектора (число элементов). Другие атрибуты: *names, dimnames, dim* и др.

*attr*(x, 'author''' <- "Caesar" – эта функция создает новый атрибут, здесь – автор.

attributes(x) обнуляет существующие атрибуты вектора.

Векторы можно использовать очень широко. Например, можно получить ряд последовательного возведения в степень 2 через вектор:

x<-2^(0:10)

#### 3. Условия и циклы в R

В R, как и в других языках программирования, широко используются разные виды условий, циклы.

if-else

if ("condition") {<do something>} else {<do another thing>}

*condition* – это выражение, результатом которого будет логический вектор длины 1 (TRUE или FALSE, NA приведет к ошибке), в фигурных скобках – произвольные команды. Обратите внимание, что нельзя переносить *else* на новую строку.

```
if (sqrt(2) > 1.5) {
  print ("Greater!")
  } else {
  print ("Less!")
  }
```

Если условие необходимо проверить на целом векторе, то нужно использовать другой оператор *ifelse* (пишется слитно). В задаче выясняется, является ли заданный вектор (х) нестрого убывающим или нестрого возрастающим. В этих случаях будет напечатано слово 'TRUE', иначе – 'FALSE'. Попробуйте самостоятельно разобраться с каждой строкой кода. Для того чтобы узнать значение функций, можно воспользоваться подсказкой в консоли: ?diff, ?prod

```
is_monotone <- function(x) {
b<-diff(x,1)
    a<-c(b>=0)*1
    d<-c(b<=0)*1
    y<-c(b == 0)*1
    ifelse (prod(y)== 1, 'FALSE',
        ifelse (prod(a)== 1, 'TRUE',
        ifelse (prod(d) == 1, 'TRUE', 'FALSE')))
}</pre>
```

Цикл *repeat*. Повтор идет до того, пока не будет достигнуто пороговое значение, после чего оператор *break* выводит из цикла.

```
i <- 0
repeat {
i <- i+runif(1)
print(i)
if (I >5) break
}
```

Цикл *while*. Этот цикл может быть не выполнен ни разу, в отличие от *repeat*, который будет выполнен хотя бы один раз.

```
i <-2^14
while (i > 1000) {
i <- i/2
print (i)
}
```

Цикл *for*. Один из самых популярных циклов. Внутри цикла необходимо принудительно вызывать функцию *print*, иначе результат не будет отображен. В некоторых случаях этот вид цикла ведет к неэффективному коду.

```
for (i in 1:8) {
    if (i %% 2 == 0) print (i)
    }
    for (i in letters) {
        if (i == 'b')next
        if (i == 'd') break
        print (i)
    }
```

#### 4. Пример неэффективного кода

Многие функции в R векторизованы. Следующий код содержит лишний цикл *for*. Кроме того, не указана длина результата, поэтому результат перезаписывается время от времени.

Задача fizz-buzz:

```
y <- vector(mode = 'character', length = 100)
y <- character (100)
for (i in 1:100) {</pre>
```

```
if (i%% 15 ==0) {
y[i] <- `fizz buzz'
} else if (i %% 3 == 0) {
y[i] <- `fizz;
} else if (i %% 5 == 0) {
y[i] <- `buzz'
} else {
y[i] <- i
}
}</pre>
```

Второй способ решения этой задачи (эффективный):

*Матрица* – двумерный массив данных одного типа, организованных в ряды (*row*) и столбцы (*col*). Для создания матрицы можно воспользоваться функцией *matrix*.

Попробуйте создать разные матрицы, варьируя условия.

#### 5. Упражнения «Матрицы»

matrix (1:6, nrow = 2) matrix (1:6, nrow = 2, byrow = TRUE) matrix (5:6, nrow = 2, ncol = 5) У матрицы есть специальный атрибут *dim*, который отвечает за ее размерность.

```
m<- matrix (1:6, ncol = 3)
dim(m)</pre>
```

Такой же результат можно получить, запросив отдельно количество строк и столбцов в матрице:

```
c(nrow(m), ncol(m))
```

С матрицами можно проводить те же арифметические операции, что и с векторами. Их можно также индексировать.

#### 6. Упражнение «Арифметические операции с матрицами»

```
m1<- matrix (1:4, nrow = 2)
m2<- matrix (c(1, 2, 2, 3), nrow = 2)
m1
m2
m1+m2
m1*2
m1*m2
```

Матрицы можно присоединять друг к другу. Для этого используются команды *rbind/cbind*. Первая команда «удлиняет» матрицу, так как присоединение идет по рядам, число столбцов не изменяется. Во втором случае матрица «расширяется», а число рядов остается прежним. С помощью этих команд можно объединять любое количество матриц и векторов.

При индексации необходимо учитывать две размерности, это означает, что каждый элемент имеет два индекса, которые обозначают столбец и ряд. Если использовать только один индекс, то выделится весь столбец или весь ряд:

m<-matrix (1:10, ncol= 5)
m[1, 3]
m[1, ]
m[,3]</pre>

Иногда необходимо применить какую-либо функцию не ко всей матрице, а только к определенным столбцам или строкам. Для этого используется функция *apply*. Она имеет три аргумента: матрица, индекс (1 – построчно, 2 – по столбцам) или функция.

```
f<- function(x) sum(x^2)
apply(m,1,f)</pre>
```

Последнее выражение означает, что функция применяется к матрице *m*, построчно, *f* – название используемой фукнции. В упражнении используется матрица m, которая была задана ранее.

apply(m,1,f)

#### 7. Датафреймы

Самый важный тип данных, с которым работает R, – это таблицы данных или датафреймы (*data frame*). Датафрейм представляет собой одномерный список из векторов одинаковой длины. Таким образом, каждый датафрейм состоит из списка столбцов, причем внутри одного столбца все данные должны быть одного типа, а вот столбцы уже могут отличаться друг от друга. Столбцы соответствуют

переменным, а строки – наблюдениям. Для наглядности попробуем самостоятельно создать датафрейм.

Допустим, мы можем создать три вектора, описывающие присутствующих в этой аудитории:

age<- с(введем сюда цифры, обозначающие возраст студентов) gender<-с (введем сюда обозначение пола: F или M) name<- с(введем сюда имена, обязательно в кавычках: "Olga") data<- list(age, gender, name) data[[1]][1] data[[2]][3] df <- data.frame(Name = name, Age = age, Gender = gender) typeof(df)

Команда покажет, что df – это *list*, т. е. список данных. Мы можем делать в R статистическую обработку данных.

#### Практическая работа №10. Элементарная статистика

В векторе age отберите только те наблюдения, которые отклоняются от среднего не больше, чем на одно стандартное отклонение. Сохраните эти наблюдения в новую переменную и напечатайте ее при помощи команды print.

Функции:

mean(x) – среднее значение вектора х

sd(d) – стандартное отклонение вектора х

abs(n) – абсолютное значение числа n

#### 1. Загрузка данных в R

Можно не только работать с созданными самостоятельно в R датафреймами, но и использовать уже готовые данные. Доступ к ним можно получить с помощью команды загрузки данных (даны в форме справки, впереди знак вопроса):

?read.table – эта функция имеет много аргументов, на описании которых необходимо остановиться.

Аргументы: *file* – это название файла с данными; *header* – позволяет указать, является ли верхняя строка всех столбцов названием (тогда указать значение TRUE, по умолчанию – FALSE); *sep* – указывает, каким знаком разделены столбцы в данных, dec – каким знаком разделены десятичные дроби. Последние два аргумента принципиально важны, если вы работаете с файлами .csv, созданными в англоязычных и русскоязычных версиях Exel. По умолчанию в R *sep* имеет значение «,», а *dec* – «.». Разделение столбцов в русскоязычных версиях осуществляется с помощью точки с запятой, разделение десятичных дробей – запятыми.

?read.csv – аналогичная функция, но с другими значениями по умолчанию.

Чтобы функция сработала, убедитесь, что

- указано расширение загружаемого файла,
- название файла взято в кавычки,
- загружаемый файл находится в рабочей директории.

Если файл находится не в рабочей директории, есть три способа решить эту проблему:

 изменить рабочую директорию (в RStudio: Session -> Set Working Directory -> Choose Directory...);

- переместить файл в актуальную рабочую директорию;
- указать полный путь к файлу. Например:

```
read.csv("C:/Users/username/documents/myexp.csv")
read.csv('C:/R/myexp.csv')
mydata<-read.csv('C:/R/myexp.csv')
mydata
head(mydata)
```

(Обратите внимание, что путь к файлу на вашем компьютере может быть другим!). Если выполнить команду *mydata*, то мы получим просто массив данных, которые трудно будет воспринимать визуально. Если выполнить команду *head*(mydata), то в верхней части появятся заголовки столбцов по 6 строк (повторяться может несколько раз).

Можно посмотреть только названия переменных: names(mydata).

Есть другие способы, которые позволяют посмотреть эти данные.

*View*(mydata)

В консоли появится дополнительная вкладка, где данные будут представлены в виде таблицы. Команда выводит до 1000 наблюдений.

*str*(mydata) – позволяет увидеть тип переменных. Все данные, которые записываются буквами, относятся в формате csv к типу переменных Factor.

summary(mydata) – делает элементарный стат. анализ.

К формату сsv можно привести данные других форматов, чаще всего это делают для таблиц Exel. Можно использовать и формат .txt. Отличия будут в команде, которая пишется для загрузки этого файла в среду R. Создадим файлы myexp.txt и myexp.csv прямо в программе Excel. Для этого необходимо проделать следующее:

• Нажать на вкладку Файл в левом верхнем углу.

• Выбрать Экспорт и найти опцию Изменить тип файла.

• Кликнуть на нее (появится список различных форматов).

• Выбрать .txt (разделитель — знак табуляцией '*tab*') или .csv (разделитель запятая).

• Прописать расположение и имя файла.

Теперь необходимо лишь импортировать полученные файлы в среду R. Для этого воспользуемся командами:

для .csv:

mydata1 <- read.csv("C:/Users/username/documents/myexp.csv", sep = ";")

для .txt:

```
>mydata2 <- read.table("C:/Users/username/documents/myexp.csv", sep = "\t")
```

Сейчас для загрузки файлов в форматах программы Excel (.xsl или .xlsx) существует R-пакет "xlsx", который позволяет загружать их без предварительного преобразования.

Сначала необходимо загрузить пакет "xlsx" в среду R:

install.packages("xlsx", dep = T)

и подключить этот пакет при помощи команды

library("xlsx")

Для загрузки таблицы необходимо прописать путь и имя файла с указанием номера листа, в котором находится таблица:

имя таблицы в R <- read.xlsx("путь к файлу/имя файла.xlsx", sheetIndex = "номер листа")

В нашем случае имя файла "myexp.xlsx", а имя таблицы данных в среде R – "mydata3":

mydata3 <- read.xlsx("C:/Users/username/documents/myexp.xlsx", sheetIndex = 1)

Обратите внимание, что таблицы Exel, с которыми планируется работать дальше в среде R, должны быть приведены к единой размерности, четко структурированы и, желательно, содержать всю информацию на латинице.

#### 2. Анализ датафрейма

С датафреймом можно проделывать практически все операции, которые мы учились делать с векторами и матрицами. Во-первых, поскольку датафрейм является списком, к нему применимы методы индексации списков. Кроме того, его можно индексировать как двумерную матрицу. Если сейчас вернуться к тому датафрейму, который мы только что сделали, и перейти во вкладку *df* в левом верхнем углу, то увидим таблицу, куда внесены все данные.

*mydata*\$*score* – выбирает отдельную переменную по имени. Значения каждой переменной представляют собой вектор.

b<-mydata\$score mean(b) summary(b) mydata\$score\*2 и т. д.

Можно добавить в датафрейм новые переменные, например: mydata\$new\_varible<-0 (вместо 0 может быть что угодно, любой вектор).

Индексирование:

mydata[1,1] mydata[5,] mydata[c(2, 193, 225), 1] mydata[,2:5] Таким образом, можно обращаться к любому количеству, к любым элементам таблицы, опираясь на принципы индексации, рассмотренные для векторов и матриц.

Можно выделять отдельными таблицами часть большого датафрейма. Для этого подходит команда *subset*:

mydata2<- subset(mydata, gender == 'female')
mydata3<- subset(mydata, gender == 'male')</pre>

Здесь будут работать также команды *rbind* и *cbind*, которые позволяют объединять датафреймы:

mydata4<- rbind(mydata2, mydata3)

Еще один источник данных, с которыми можно работать, – это библиотека datasets, которая загружается автоматически. Чтобы просмотреть весь список доступных датасетов, выполните команду *library*(help = "datasets"). Некоторые из этих датасетов мы будем использовать, они доступны по имени в любой момент работы. Например, если вы захотите использовать датасет *mtcars*, можете просто выполнить команду *mtcars*, чтобы увидеть данные в консоли.

Команда *data*(mtcars) добавит датасет в рабочую среду.

Команда *help*(mtcars) выведет информацию о датасете.

Команда *my data <- mtcars* запишет датасет в новую переменную.

#### 3. Упражнение «Работа с датасетом»

В переменной *df* сохранены данные *mtcars*. В датафрейме *df* создайте новую переменную под названием *even\_gear*, в которой будут единицы, если значение переменной (*gear*) четное, и нули, если количество нечетное.

Напечатайте эту переменную при помощи команды *print(df\$even\_gear)* Вам может помочь оператор деления с остатком:

5 %% 2 == 1

TRUE

В переменной df сохранены данные *mtcars*. Создайте новую переменную new\_var в данных, которая содержит единицы в строчках, если в машине не меньше четырёх карбюраторов (переменная *carb*) или больше шести цилиндров (переменная *cyl*). В строчках, в которых условие не выполняется, должны стоять нули.

Напечатайте переменную при помощи команды: print(df\$new\_var)

# Практическая работа № 11.

#### Группировка наблюдений за загрязнением воздуха

В переменную *df* сохранены встроенные данные *airquality*. В новую переменную сохраните subset исходных данных, оставив наблюдения только для месяцев 7, 8 и 9.

При помощи функции *aggregate* рассчитайте количество наблюдений (не пропущенных) по переменной *Ozone* в каждом месяце. Для определения количества наблюдений используйте функцию *length*(). Воспользуйтесь записью с помощью формулы, при которой пропущенные значения не учитываются:

aggregate – позволяет применить функцию к определенной группе значений. aggregate (x = df\$hp, by = list(df\$vs), FUN = mean) – df\$hp разбита на две группы по значениям vs и в группах рассчитано среднее.

## *Colnames*() – показывает названия столбцов датафрейма. Можно присвоить этой функции новые значения через вектор:

aggregate(hp~vs, df, mean) aggregate(hp~vs+am, df, mean) aggregate(x = df\$hp, by= list(df\$vs, df\$am), FUN = mean) aggregate (x = df[,-c(8,9)], by = list(df\$am), FUN = median) aggregate(df[,c(1,2)], by = list(df\$am, df\$vs), FUN = sd) aggregate(cbind(mpg, disp) ~ am+vs, df, sd)

*is.na* – игнорирует пропущенные значения, *na.rm* – исключает пропущенные значения (na.rm = T).

#### Визуализация данных в **R**

Для представления результатов данных в статьях или презентациях необходима их визуализация в виде таблиц или графиков, гистограмм. R позволяет строить графики и гистограммы любой сложности. Элементарная графика возможна с помощью автоматически установленных пакетов, для более сложных манипуляций потребуется скачать и установить новые пакеты, например, ggplot2, GrapheR, ggraph и др.

Базовые графики и гистограммы можно построить, используя команды:

barplot, boxplot, plot, hist, pie.

Рассмотрим подробнее некоторые из этих функций. *Pie* строит круговую диаграмму. Характеристики этой диаграммы можно задать в скобках, варьируя значения аргументов. В примере указаны значения аргументов по умолчанию, их можно вообще не прописывать в функции, если вы не хотите изменить их значения.

*pie*(x, labels = names(x), radius = 0.8, clockwise = FALSE, init.angle = if(clockwise) 90 else 0, density = NULL, angle = 45, col = NULL, border = NULL, lty – NULL, main = NULL),

*labels* – подписи осей, *radius* – размер диаграммы, *clockwise* – размещение по часовой стрелке или против, *col* –цвет, *main* – название диаграммы.

#### 1. Упражнение «Круговая диаграмма»

Составьте круговую диаграмму на основании встроенного датасета *огоп*. Задайте самостоятельно радиус диаграммы, цвет секторов. Укажите название диаграммы и сделайте подписи. Одним из самых востребованных способов визуализации является гистограмма, *hist*.

hist (x, breaks = "Sturges", freq = NULL, probability = !freq, include.lowest = TRUE, right = TRUE, density = NULL, angle = 45, col = NULL, border = NULL, main = paste("Histogram of", xname), xlim = range(breaks), ylim = NULL, xlab = xname, ylab, axes = TRUE, plot = TRUE, labels = FALSE, nclass = NULL, warn.unused = TRUE, ...).

Работа с функцией *hist* аналогична предыдущему упражнению. Дополнительно можно подписать оси координат с помощью команд *xlab* и *ylab*.

#### 2. Упражнение «Гистограмма»

Составьте гистограмму на основании любого встроенного датасета. Задайте самостоятельно плотность гистограммы, размеры и цвета столбцов. Укажите название гистограммы и сделайте подписи, не забудьте подписать и данные, и оси координат. Можно использовать и другие графические функции. Так, *plot(x)* рисует график значений вектора x, упорядоченных вдоль оси x.

*plot*(x,y) изображает график зависимости у от х.

*barplot*(x) представляет собой столбчатую диаграмму значений вектора х.

Также можно усложнить ваши диаграммы и графики, добавив к ним графические функции низкого уровня.

#### 3. Графические функции низкого уровня

*points*(x, y) – добавляет к графику точки с координатами x и y.

lines(x, y) – то же, но для линий.

*text*(x, y, labels, ...) – добавляет к графику текст, указанный при помощи аргумента labels, в место с координатами х и у.

*segments*(x0, y0, x1, y1) – изображает линию, проходящую из точки с координатами (x0, y0) в точку с координатами (x1, y1).

*arrows*(x0, y0, x1, y1, angle= 30, code=2) – то же, что и *segments*(), но для стрелок; указатель стрелки будет изображен в точке с координатами (x0, y0), если *code* = 1, в точке (x1, y1), если *code* = 2, и с обеих сторон стрелки, если *code* = 3; аргумент angle управляет тем, насколько заострен указатель стрелки.

abline(h = y) - рисует горизонтальную линию, отходящую от координаты; abline(v = x) - рисует вертикальную линию, отходящую от координаты x.

*rect*(x1, y1, x2, y2) – рисует прямоугольник, чьи левая, правая, нижняя и верхняя грани задаются координатами x1, y1, x2 и y2 соответственно.

*legend*(x, y, legend) – добавляет легенду в точке (x, y) с символами, которые задаются при помощи аргумента legend.

*title()* – добавляет к графику заголовок (также можно добавить подзаголовок).

*axis*(side, vect) – добавляет к графику ось внизу (side = 1), слева (2), сверху (3) или справа (4).

#### 4. Управление графическим параметрами

Часть графических параметров уже использовалась нами при выполнении самостоятельных заданий в этом разделе. Обратите внимание, что многие из

перечисленных ниже параметров могут быть установлены глобально при помощи команды *par*(...) или локально в виде аргументов графических функций.

*adj* – контролирует выравнивание текста (0 – по левому краю, 0.5 – по центру, 1 – по правому краю).

bg – устанавливает цвет фона (например, bg = "red"); список 657 доступных цветов можно просмотреть при помощи команды *colors*().

*cex* – управляет размером шрифта и символов; следующие параметры имеют аналогичное действие в отношении: чисел на координатных осях – cex.axis, меток осей – cex.lab, основного заголовка графика – cex.main и подзаголовка – cex.sub

col – задает цвет символов и линий (в виде названий цветов "red", "blue" и т.п. – см. colors() или в виде "RRGGBB" – см. rgb(), hsv(), gray() и rainbow()); так же, как и для *cex*, имеются варианты col.axis, col.lab, col.main, col.sub.

*font* – целое число, задающее стиль текста (1: нормальный, 2: курсив, 3: жирный, 4: жирный курсив); как и для *cex*, имеются варианты font.axis, font.lab, font.main, font.sub.

# Специальные возможности R в области экологии и природопользования

Как уже не раз говорилось, R – это модульный язык, в который постоянно вносятся изменения и дополнения в виде пакетов специальных функций. Ряд пакетов предназначен специально для анализа данных в области экологии и природопользования. Даже базовая версия R содержит широкий спектр функций для использования в области охраны окружающей среды. Эта, уже имеющаяся, функциональность дополняется множеством пакетов, доступных на сайте CRAN, которые предоставляют специальные методы, например, метод кластерного анализа.

Ниже приведен список некоторых наиболее интересных и специализированных пакетов и функций. Кроме этих пакетов, существует огромное количество пакетов с дополнительными возможностями для статистического анализа, которые не будут здесь описаны, поскольку являются универсальными и могут использоваться в различных областях науки.

Пакет *EnvStats* является преемником модуля S-PLUS EnvironmentalStats. Этот пакет данных позволяет совершать действия над неиндексированными данными, выполнять оценку достоверности и отображать ее результаты, сравнивать концентрации химических веществ со стандартами (обратите внимание, что стандарты международные) с использованием доверительных интервалов, сравнивать между собой сразу несколько образцов, использовать метод Монте-Карло для анализа данных.

Пакет *BiodiversityR* представляет собой графический интерфейс для анализа биоразнообразия сообщества. Пакет *untb* представляет собой набор утилит для анализа данных о биоразнообразии, включая симуляцию экологического дрейфа в рамках Унифицированной теории биоразнообразия и биогеографии Хаббелла и расчет различных диагностических функций, например, кривой видового накопления Престона.

Пакет *primer* предоставляет множество функций для моделирования экологических данных и базовой теоретической экологии, включая функции, связанные с демографическими матричными моделями, метапопуляцией и исходными моделями, моделями хозяев-паразитоидов и болезней и др.

Пакет *vegan* предоставляет широкий спектр функций, связанных с экологией, таких как индексы разнообразия (включая так называемые числа Хилла), ранжированные диаграммы обилия, логарифмические ряды Фишера, модель разломанного стержня (распределение обилия видов с максимально возможной в природе равномерностью), модель обилия Хаббелла и др.

Пакет *adehabitat* дополняет *ade4* и предоставляет набор инструментов для анализа выбора мест обитания животных.

Пакет *popbio* можно использовать для построения и анализа возрастных или сценарных матричных моделей популяции.

Пакет *openair* содержит немало инструментов для анализа и интерпретации параметров временных серий загрязнения воздуха.

Ряд пакетов предназначен для оценки численности животных и связанных с этой оценкой параметров. К таким пакетам относятся: *Rcapture, secr, SPACECAP, DSpat, RMark, mrds, marked, Distance* и *dsm.* 

Несколько специальных пакетов предназначено для анализа данных в области почвоведения: soiltexture может описывать и анализировать текстуру грунта, его классификацию и преобразования; адр содержит набор алгоритмов, связанных с моделированием почвенных ресурсов, классификацией почв, агрегацией профиля почвы и визуализацией; *HydroMe* оценивает параметры в моделях инфильтрации и водоудержания. Существует также несколько пакетов, относящихся к области климатологии: *seas* реализует ряд функций для анализа и визуализации сезонных данных; *RMAWGEN* представляет собой набор функций для пространственной стохастической генерации временных рядов температуры И осадков с моделей; *Interpol.T* проводит авторегрессионных использованием векторных интерполяцию минимальных ежечасную ежедневных И максимальных температурных рядов.

1. *Божко А. Н.* Photoshop CS: технология работы: учеб.-справ. изд. / А. Н. Божко. М.: Кудиц-образ, 2004. 624 с.

2. Бондаренко М. Правильный вектор: бесплатные инструменты для создания векторной графики / М. Бондаренко, С. Бондаренко. 2013. 3DNews Daily Digital Digest (www.3dnews.ru); URL: https://3dnews.ru/774197 (дата обращения: 01.11.2017).

3. Васильев В. Е. Компьютерная графика: учеб. пособие / В. Е. Васильев, А. В. Морозов. СПб.: СЗТУ, 2005. 101 с.

4. *Гумерова Г.Х.* Основы компьютерной графики: учеб. пособие / Г. Х. Гумерова. Казань: Изд-во КНИТУ, 2013. 87 с.

5. Заика А. Photoshop для начинающих / А. Заика. М.: РИПОЛ классик, 2013. 200 с.

6. *Залогова Л. А.* Компьютерная графика. Элективный курс: учеб. пособие / Л. А. Залогова. М.: Б:ИНОМ. Лаборатория знаний, 2005. 212 с.

7. Искусство дизайна с компьютером и без. М.: Кудиц-образ, 2005. 204 с.

8. *Кабаков Р.* R в действии. Анализ и визуализация данных на языке R / P. Кабаков. М.: ДМК-Пресс, 2012. 588 с.

9. *Кнут Д*. Э. Компьютерная типография / Д. Э. Кнут. М.: Мир: АСТ, 2003.
668 с.

10. Корнеенко О. Е. Компьютерные информационные технологии: практ. пособие для студ. экон. спец. / О. Е. Корниеко, О. И. Полякова. Гомель: ГГУ им. Ф. Скорины, 2008. 78 с.

11. Коцюбинский А. О. Компьютерная графика / А. О. Коцюбинский, С. В. Горошев. М.: Технолоджи-3000, 2001. 752с.

12. *Основы* информатики и информационные технологии: учебно-метод. комплекс для студ. ист. фак.: в 2 ч. Ч. 2 / Е. Н. Балыкина, Е. Э. Попова, Д. Н. Бузун. Минск: БГУ, 2008. 96 с.

13. Павлова Л. Д. Информатика. Учебный курс: учеб. пособие / Л. Д. Павлова, О. А. Кондратова, Н. В. Балицкая // Современные проблемы науки и образования. 2009. № 1. URL: https://www.science-education.ru/ru/article/view?id=865 (дата обращения: 14.12.2017).)

14. Пономаренко С. И. Adobe Photoshop 6.0: Наиболее полное руководство /С. И. Пономаренко. СПб.: БХВ - Петербург, 2002. 832 с.

15. Филиппович А.Ю. Компьютерные шрифты: виды, характеристики и технологии создания: лекции по дисциплине «Искусство шрифта» / А. Ю. Филиппович. М.: Моск. гос. ун-т печати им. Ивана Фёдорова, 2012. URL: http://it-claim.ru/Education/Course/Fonts/lecture/2TLectureFonts.pdf (дата обращения: 01.11.2017).

16. *Херн Д*. Компьютерная графика и стандарт OpenGL / Д. Херн. 3-е изд. М.: Вильямс, 2005. 1168 с.

17. Шипунов А. Б. Наглядная статистика. Используем R! / А. Б. Шипунов, Е. М. Балдин. М.: ДМК Пресс, 2009. 298 с.

18. Шитиков В. К., Розенберг Г. С. Рандомизация, бутстреп и методы Монте-Карло. Примеры статистического анализа данных по биологии и экологии. Тольятти: Кассандра, 2013. 314 с.

19. *Adobe Photoshop CS*. М.: ТРИУМФ, 2004. 576 с.

20. *Biopython* Tutorial and Cookbook. URL: http://biopython.org/ DIST/docs/tutorial/Tutorial.html (дата обращения: 11.01.2018).

21. *Official* git repository for Biopython. URL: https://github.com/ biopython/biopython (дата обращения: 11.01.2018).

22. *Matloff N*. The Art of R Programming / *N*. *Matloff*. San Francisco: No Starch Press, Inc. 2011. 402 p.

23. *Cotton R*. Learning R. URL: https://avaxsearch.pro/?q=learning%20R (дата обращения: 09.12.2017).

24. *Paradis E.* R for Beginners. URL: http://www.pdfdrive.net/r-for-beginners-the-comprehensive-r-archive-network-e1543329.html (дата обращения: 17.12.2017).

25. *Kickstarting* R. URL: https://cran.r-project.org/doc/contrib/Lemon-kickstart/index.html (дата обращения: 11.01.2018).

26. *CRAN* project. URL: https://cran.r-project.org/ (дата обращения: 11.01.2018).

27. *Packages* for Ecology and Environments. URL: https://cran.r-project.org/web/views/Environmetrics.html (дата обращения: 22.12.2017).

Учебное издание

Данилова Мария Александровна Васильева Юлия Сергеевна Красильников Виталий Павлович

# КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ В ЭКОЛОГИИ И ПРИРОДОПОЛЬЗОВАНИИ

Под общей редакцией М. А. Даниловой

Учебное пособие

Редактор Л. А. Богданова Корректор Л. И. Семицветова Техническая подготовка материалов: М. А. Данилова

> Объем данных 8 Мб Подписано к использованию 12.12.2018

Размещено в открытом доступе на сайте www.psu.ru в разделе НАУКА / Электронные публикации и в электронной мультимедийной библиотеке ELiS

Издательский центр Пермского государственного национального исследовательского университета. 614990, г. Пермь, ул. Букирева, 15