

МИНИСТЕРСТВО КУЛЬТУРЫ РОССИЙСКОЙ ФЕДЕРАЦИИ
«Сибирский государственный институт искусств имени Дмитрия Хворостовского»

Приложение 2 к рабочей программе

Методические рекомендации по освоению дисциплины

«Современные информационные технологии»

для обучающихся по программе направления подготовки

53.03.02 Музыкально-инструментальное искусство,

Оркестровые духовые и ударные инструменты

Разработчики:

Ст. преподаватель Е.В. Гордеева

ст. преподаватель Шкредова И.Н.

1. Пояснительная записка

Методические указания для студентов по освоению дисциплины «Современные информационные технологии» разработаны в соответствии с ФГОС ВО по направлению подготовки 53.03.02 Музыкально-инструментальное искусство и рабочей программой дисциплины «Современные информационные технологии».

Цель методических рекомендаций – обеспечить студенту оптимальную организацию процесса изучения дисциплины, а также выполнения лабораторных работ по темам.

В современных условиях одним из важнейших требований к специалисту высокого уровня является умение самостоятельно пополнять свои знания, ориентироваться в потоке научной и культурной информации.

Приступая к изучению дисциплины «Современные информационные технологии», студенты должны ознакомиться: с рабочей программой дисциплины, с настоящими методическими указаниями, фондом оценочных средств, с учебной, научной и методической литературой, имеющейся в библиотеке Института, получить доступ в электронные библиотечные системы, получить в библиотеке рекомендованные учебники и учебно-методические пособия, завести тетрадь для конспектирования лекций, электронный носитель для хранения своих лабораторных работ, создать папку в компьютерном классе для сдачи лабораторных по всем темам дисциплины.

Для обеспечения систематической и регулярной работы по изучению дисциплины и успешного прохождения промежуточных и итоговых контрольных испытаний обучающемуся рекомендуется придерживаться следующего порядка обучения:

1. Регулярно изучать каждую тему дисциплины, используя различные формы индивидуальной работы.
2. Согласовывать с преподавателем виды работы по изучению дисциплины.
3. По завершении отдельных тем сдавать лабораторные работы преподавателю.

При регулярном выполнении текущих заданий, активном участии в семинарах и успешном прохождении межсессионной аттестации студент может претендовать на сокращение программы промежуточной (итоговой) аттестации по дисциплине.

2. Формы самостоятельной работы

Дисциплина «Современные информационные технологии» ставит своей задачей научить студентов основам использования информационных технологий в своей профессиональной деятельности.

Одной из форм самостоятельной работы по любой дисциплине является **освоение теоретического материала**. При этом рекомендуется использовать учебники, дополнительную литературу, а также конспекты лекций. Для удобства приводим здесь содержание основных тем.

Лекции

1. Понятие «информация».

Слово «информация» происходит от латинского слова *informatio*, что в переводе означает сведение, разъяснение, ознакомление. Понятие «информация» является базовым в курсе информатики, невозможно дать его определение через другие, более «простые» понятия. В геометрии, например, невозможно выразить содержание базовых понятий «точка», «луч», «плоскость» через более простые понятия. Содержание основных, базовых понятий в любой науке должно быть пояснено на примерах или выявлено путем их сопоставления с содержанием других понятий.

В случае с понятием «информация» проблема его определения еще более сложная, так как оно является общенаучным понятием. Данное понятие используется в различных науках (информатике, кибернетике, биологии, физике и др.), при этом в каждой науке понятие «информация» связано с различными системами понятий.

Информация в физике. В физике мерой беспорядка, хаоса для термодинамической системы является энтропия системы, тогда как информация (антиэнтропия) является мерой упорядоченности и сложности системы. По мере увеличения сложности системы величина энтропии уменьшается, а величина информации увеличивается. Процесс увеличения информации характерен для открытых, обменивающихся веществом и энергией с окружающей средой, саморазвивающихся систем живой природы (белковых молекул, организмов, популяций животных и так далее).

Таким образом, в физике информация рассматривается как антиэнтропия или энтропия с обратным знаком.

Информация в биологии. В биологии, которая изучает живую природу, понятие «информация» связывается с целесообразным поведением живых организмов. Такое поведение строится на основе получения и использования организмом информации об окружающей среде.

Понятие «информация» в биологии используется также в связи с исследованиями механизмов наследственности. Генетическая информация передается по наследству и хранится во всех клетках живых организмов. Гены представляют собой сложные молекулярные структуры, содержащие информацию о строении живых организмов. Последнее обстоятельство позволило проводить научные эксперименты по клонированию, то есть созданию точных копий организмов из одной клетки.

Информация в кибернетике. В кибернетике (науке об управлении) понятие «информация» связано с процессами управления в сложных системах (живых организмах или технических устройствах). Жизнедеятельность любого организма или нормальное функционирование технического устройства зависит от процессов управления, благодаря которым поддерживаются в необходимых пределах значения их параметров. Процессы управления включают в себя получение, хранение, преобразование и передачу информации.

Социально значимые свойства информации. Человек - существо социальное, для общения с другими людьми он должен обмениваться с ними информацией, причем обмен информацией всегда производится на определенном языке — русском, английском и так далее. Участники дискуссии должны владеть тем языком, на котором ведется общение, тогда информация будет

понятной всем участникам обмена информацией.

Информация должна быть *полезной*, тогда дискуссия приобретает практическую ценность. Бесполезная информация создает информационный шум, который затрудняет восприятие полезной информации. Примерами передачи и получения бесполезной информации могут служить некоторые конференции и чаты в Интернете. Широко известен термин «средства массовой информации» (газеты, радио, телевидение), которые доводят информацию до каждого члена общества. Такая информация должна быть *достоверной* и *актуальной*. Недостоверная информация вводит членов общества в заблуждение и может быть причиной возникновения социальных потрясений. Неактуальная информация бесполезна и поэтому никто, кроме историков, не читает прошлогодних газет.

Для того чтобы человек мог правильно ориентироваться в окружающем мире, информация должна быть *полной* и *точной*. Задача получения полной и точной информации стоит перед наукой. Овладение научными знаниями в процессе обучения позволяют человеку получить полную и точную информацию о природе, обществе и технике.

Единицы измерения количества информации.

Информация и знания. Человек получает информацию из окружающего мира с помощью органов чувств, анализирует ее и выявляет существенные закономерности с помощью мышления, хранит полученную информацию в памяти. Процесс систематического научного познания окружающего мира приводит к накоплению информации в форме знаний (фактов, научных теорий и так далее). Таким образом, с точки зрения процесса познания информация может рассматриваться как знания. Информацию, которую получает человек, можно считать мерой уменьшения неопределенности знаний. Если некоторое сообщение приводит к уменьшению неопределенности наших знаний, то можно говорить, что такое сообщение содержит информацию.

Уменьшение неопределенности знаний.

Подход к информации как мере уменьшения неопределенности знаний позволяет количественно измерять информацию, что чрезвычайно важно для информатики. Рассмотрим вопрос об определении количества информации более подробно на конкретных примерах.

Пусть у нас имеется монета, которую мы бросаем на ровную поверхность. С равной вероятностью произойдет одно из двух возможных событий — монета окажется в одном из двух положений: «орел» или «решка».

Можно говорить, что события равновероятны, если при возрастающем числе опытов количества выпадений «орла» и «решки» постепенно сближаются. Например, если мы бросим монету 10 раз, то «орел» может выпасть 7 раз, а решка — 3 раза, если бросим монету 100 раз, то «орел» может выпасть 60 раз, а «решка» — 40 раз, если бросим монету 1000 раз, то «орел» может выпасть 520 раз, а «решка» — 480 и так далее.

В итоге при очень большой серии опытов количества выпадений «орла» и «решки» практически сравниваются. Перед броском существует неопределенность наших знаний (возможны два события), и, как упадет монета, предсказать невозможно. После броска наступает полная определенность, так как мы видим (получаем зрительное сообщение), что монета в данный момент находится в определенном положении (например, «орел»). Это сообщение приводит к уменьшению неопределенности наших знаний в два раза, так как до броска мы имели два вероятных события, а после броска — только одно, то есть в два раза меньше.

В окружающей действительности достаточно часто встречаются ситуации, когда может произойти некоторое количество равновероятных событий. Так, при бросании равносторонней

четырёхгранной пирамиды существуют 4 равновероятных события, а при бросании шестигранного игрального кубика - 6 равновероятных событий.

Чем больше количество возможных событий, тем больше начальная неопределенность и соответственно тем большее количество информации будет содержать сообщение о результатах опыта.

Единицы измерения количества информации.

Для количественного выражения любой величины необходимо определить единицу измерения. Так, для измерения длины в качестве единицы выбран метр, для измерения массы — килограмм и так далее. Аналогично, для определения количества информации необходимо ввести единицу измерения.

За единицу количества информации принимается такое количество информации, которое содержит сообщение, уменьшающее неопределенность в два раза. Такая единица названа «бит».

Если вернуться к опыту с бросанием монеты, то здесь неопределенность как раз уменьшается в два раза и, следовательно, полученное количество информации равно 1 биту.

Минимальной единицей измерения количества информации является бит, а следующей по величине единицей является байт, причем $1 \text{ байт} = 2^3 \text{ бит} = 8 \text{ бит}$.

В информатике система образования кратных единиц измерения количества информации несколько отличается от принятых в большинстве наук. Традиционные метрические системы единиц, например Международная система единиц СИ, в качестве множителей кратных единиц используют коэффициент 10^n , где $n = 3, 6, 9$ и так далее, что соответствует десятичным приставкам Кило (10^3), Мега (10^6), Гига (10^9) и так далее.

Компьютер оперирует числами не в десятичной, а в двоичной системе счисления, поэтому в кратных единицах измерения количества информации используется коэффициент 2^n .

Так, кратные байту единицы измерения количества информации вводятся следующим образом:

$$1 \text{ Кбайт} = 2^{10} \text{ байт} = 1024 \text{ байт};$$

$$1 \text{ Мбайт} = 2^{10} \text{ Кбайт} = 1024 \text{ Кбайт};$$

$$1 \text{ Гбайт} = 2^{10} \text{ Мбайт} = 1024 \text{ Мбайт}.$$

Количество возможных событий и количество информации.

Существует формула, которая связывает между собой количество возможных событий N и количество информации I : $N=2^I$.

По этой формуле можно легко определить количество возможных событий, если известно количество информации. Например, если мы получили 4 бита информации, то количество возможных событий составляло: $N = 2^4 = 16$.

Наоборот, для определения количества информации, если известно количество событий, необходимо решить показательное уравнение относительно I . Например, в игре «Крестики-нолики» на поле 8×8 перед первым ходом существует возможных события (64 различных варианта расположения «крестика»), тогда уравнение принимает вид: $64 = 2^I$.

Так как $64 = 2^6$, то получим: $2^6 = 2^I$.

Таким образом, $I = 6$ битов, то есть количество информации, полученное вторым игроком после первого хода первого игрока, составляет 6 битов.

Алфавитный подход к определению количества информации.

При определении количества информации на основе уменьшения неопределенности наших знаний мы рассматриваем информацию с точки зрения содержания, ее понятности и новизны для человека. С этой точки зрения в опыте по бросанию монеты одинаковое количество информации содержится и в зрительном образе упавшей монеты, и в коротком сообщении «Орел», и в длинной фразе «Монета упала на поверхность земли той стороной вверх, на которой изображен орел».

Однако при хранении и передаче информации с помощью технических устройств целесообразно отвлекаться от содержания информации и рассматривать ее как последовательность знаков (букв, цифр, кодов, цветов, точек изображения и так далее). Набор символов знаковой системы (алфавит) можно рассматривать как различные возможные состояния (события). Тогда, если считать, что появление символов в сообщении равновероятно, по формуле можно рассчитать, какое количество информации несет каждый символ.

Так, в русском алфавите, если не использовать букву ё, количество событий (букв) будет равно 32. Тогда: $32 = 2^I$, откуда $I = 5$ битов. Каждый символ несет 5 битов информации (его информационная емкость равна 5 битов). Количество информации в сообщении можно подсчитать, умножив количество информации, которое несет один символ, на количество символов. *Количество информации, которое содержит сообщение, закодированное с помощью знаковой системы, равно количеству информации, которое несет один знак, умноженному на количество знаков*

2. Информационные процессы.

Компьютер — это электронный прибор, предназначенный для работы с информацией посредством автоматизированной регистрации, хранения, приема, передачи, обработки и воспроизведения данных.

Понятие данных. Все явления природы сопровождаются энергетическим обменом. В его ходе тела выделяют и поглощают энергию. Энергетический обмен свойствен всем известным природным процессам: физическим, химическим, биологическим и их комбинациям.

Энергия распространяется в форме сигналов. При взаимодействии сигналов с веществом происходит изменение состояний и свойств материальных тел. Если изменение зафиксировано, говорят о том, что произошла регистрация сигнала. Так образуются данные. *Данные - это зарегистрированные сигналы любой физической природы.*

Текст на бумаге — это зарегистрированный результат взаимодействия бумаги и красителя. Запись на магнитной ленте — это зарегистрированные результаты изменения магнитного поля вблизи записывающей магнитной головки. Кратеры на Луне — это зарегистрированные результаты взаимодействия космических тел с ее поверхностью. Кратеры на Земле — это зарегистрированные результаты активности внутреннего вещества планеты.

Информация. Когда говорят о том, что компьютеры предназначены для работы с информацией, имеют в виду, что информация содержится в данных. Компьютеры работают только с данными, а информация образуется в момент взаимодействия данных с информационными методами.

Информационные методы. Чтобы получить информацию из данных, надо обладать соответствующими информационными методами. Некоторые информационные методы, основанные на органах чувств, свойственны большинству людей от рождения. К ним относятся: зрение, слух, осязание, обоняние и вкус. Некоторые информационные методы развиваются в процессе обучения, например чтение, знание языка общения, аналитическое мышление и др. Информационный метод определяет способ взаимодействия с данными с целью получения содержащейся в них информации.

Роль информационных методов. Отсутствие необходимого информационного метода не позволяет получить информацию даже при наличии данных. Например, книга представляет собой набор данных, зафиксированных в типографии краской на бумаге. Чтобы получить из нее информацию, читатель должен обладать зрением, знанием русского языка и его азбуки, умением читать и мыслить логически, а также понимать использованные термины. Но даже и этот длинный перечень не представляет полный набор необходимых методов. Например, необходимо еще располагать естественным или искусственным методом для создания достаточной освещенности. Если хотя бы одного из перечисленных методов не хватает, то либо для него подбирают адекватную замену, либо данные, содержащиеся в книге, не станут информацией.

Мы не случайно выделили слова адекватную замену. Например, если читатель книги не знаком с русским языком, то он может привлечь переводчика, но информация, которую он при этом получит, будет не совсем той, которую получит читатель, владеющий русским языком. То есть содержательная часть информации зависит как от содержательной части данных, так и от использованных информационных методов.

Информационные процессы. Информация образуется в момент взаимодействия данных с информационными методами. Такое взаимодействие называется информационным процессом. Для человека частные информационные процессы сливаются в один общий непрерывный информационный процесс — он постоянно сопровождает общение людей с внешним миром, даже во сне. У объектов живой природы общий информационный процесс прекращается вместе с процессом обмена веществ, то есть после смерти.

Процессы, связанные со сбором, хранением, поиском, обработкой, кодированием и передачей информации, называют информационными процессами.

Технические устройства также могут участвовать в информационных процессах. Простейшие примеры: теле- и радиоприемники. Для них информационный процесс прекращается вместе с прекращением рабочего энергетического обмена, то есть после выключения устройства.

Содержательная часть данных: сведения и команды. От содержательной части данных (хотя и не только от нее, но и от использованного информационного метода) зависит содержательная часть информации. Люди уделяют повышенное внимание содержательной части данных и различают в ней две составляющие: сведения и команды.

Для человека сведения и команды достаточно близки. Например, перечень инструкций, описывающих порядок вычисления корней квадратного уравнения, можно рассматривать как совокупность команд, исполнение которых приводит к правильному решению, но можно рассматривать и как сведения о приемах решения уравнений.

Для технических устройств различие между данными, содержащими сведения и команды, более существенно. Например, телевизионный приемник, принимающий сигналы из эфира или от видеоманитофона, одновременно обрабатывает оба эти типа данных. Данные, содержащие сведения, он преобразует в форму, удобную для органов чувств человека, а данные, содержащие команды, используются для управления самим приемником.

Естественные и искусственные информационные методы. Зрение, слух, мышление и другие информационные методы, относящиеся к биологическим объектам, считаются

естественными. Некоторые из них свойственны только человеку, некоторые — всем живым организмам. Общими для живой природы являются генетические методы наследования, связанные с передачей информации от предков потомкам через данные, хранящиеся в цепочках ДНК.

В тех случаях, когда человеку недостаточно естественных методов для работы с данными, он может использовать искусственные методы. Широко известно использование телескопов, микроскопов, радиоприемников, телевизионных приемников, магнитофонов и других устройств. Например, человек не обладает естественными органами чувств, способными регистрировать радиоволны. В этом случае он использует радиоприемник, с помощью которого получает информацию из данных, регистрируемых антенной.

Аппаратные и программные информационные методы. Аппаратные информационные методы представлены техническими устройствами — приборами. Мы можем говорить о том, что микроскоп предоставляет человеку информационный метод для получения данных от объектов, не видимых невооруженным глазом. Реализован этот метод аппаратно. Так же аппаратно, с помощью приборов, реализованы методы прослушивания радиопередач и просмотра телевизионных передач.

Значительно позже появился новый класс искусственных методов работы с данными — программные методы. Возьмем автоматическую стиральную машину. Она способна без участия человека выполнить замачивание, стирку, полоскание и отжим белья. В ходе работы машина руководствуется информацией, поступающей от устройства управления, в которое предварительно заложены данные, представляющие собой команды.

Компьютеры в информационных процессах. Важная особенность компьютеров, отличающая их от других приборов, предназначенных для работы с информацией, заключается в том, что современный компьютер использует все три класса информационных методов одновременно.

1. Данные обрабатываются аппаратно.
2. Характер этой обработки зависит от работающих программ.
3. Результат обработки представляется в виде данных, пригодных для взаимодействия с естественными информационными методами, которыми располагает человек, или с искусственными методами, которыми располагают другие устройства, подключенные к компьютеру.

Благодаря гибкости компьютеры могут взаимодействовать не только с человеком, но и с любыми другими техническими устройствами, в том числе и с другими компьютерами — так образуются компьютерные сети. Именно благодаря неограниченным возможностям работы с любыми устройствами компьютер и стал универсальным прибором, способным выполнять столь разнородные функции, как регистрация, хранение, обработка, прием, передача и воспроизведение данных.

3. Информационные основы процессов управления

В повседневной жизни мы всюду сталкиваемся с управлением: рабочий управляет станком, учитель — учениками, дирижер — оркестром, программист — работой компьютера и ходом выполнения программы. Вспомним слова из детской песенки: «Чтоб водить корабли, чтобы в небо взлететь надо многое знать, надо много уметь...»

Главное, надо знать: зачем выполняется управление? Например, летчик, садясь за штурвал самолета, должен заранее знать, куда и зачем он летит. Врач, назначая больному лекарство, должен быть уверен в том, что оно поможет ему выздороветь. Водитель автобуса обязан обеспечить доставку пассажиров к месту назначения. Работая на компьютере, пользователь

стремится представить информацию в удобной для работы форме. Все это означает, что для управления надо знать конкретную цель, ожидаемый результат.

При этом важно понимать, что тот, кто управляет кем-либо или чем-либо, должен обладать исходной (предварительной) информацией.

Например, для летчика исходной информацией является:

- навыки и сведения, полученные в процессе обучения летной профессии, то есть профессиональные знания;
- задание на конкретный полет в устной или письменной форме; разработка по картам предстоящего маршрута со штурманом;
- данные о состоянии летательного аппарата на момент взлета;
- данные о предполагаемых метеоусловиях.

Для водителя автомобиля, например, исходная информация - это:

- профессиональные знания по управлению автомобилем и о правилах дорожного движения;
- сведения о состоянии дороги и автомобиля перед поездкой;
- маршрут поездки.

Таким образом, всегда должен существовать объект управления, который может быть представителем как живой, так и неживой природы. В рассматриваемых примерах — это оркестр, ученики, компьютер, самолет, автомобиль.

Управление каким-либо объектом живой или неживой природы осуществляет человек или устройство, которые обладают исходной информацией: сведениями о существующей обстановке или ситуации, профессиональными знаниями (если это человек), сведениями о самом объекте управления и пр. Человек или устройство, получив необходимую исходную информацию, оказывает управляющее воздействие на объект управления. Так, например, дирижер, учитель, программист, летчик, водитель управляют соответствующими им объектами: оркестром, учениками, компьютером, самолетом, автомобилем.

Однако только исходной информации недостаточно для успешного управления. В процессе управления должна быть использована информация о фактическом состоянии объекта управления, например о текущем состоянии самого автомобиля или самолета и об обстановке на дороге или в полете. Такая информация называется текущей, или рабочей. Текущая информация о состоянии объекта управления должна постоянно поступать к человеку или устройству, которые управляют этим объектом. В этом случае говорят, что между ними существует обратная связь. Эта связь позволяет корректировать поведение объекта управления, то есть управлять им.

Такой процесс получил название замкнутого процесса управления и в виде схемы представлен на рисунке 1.

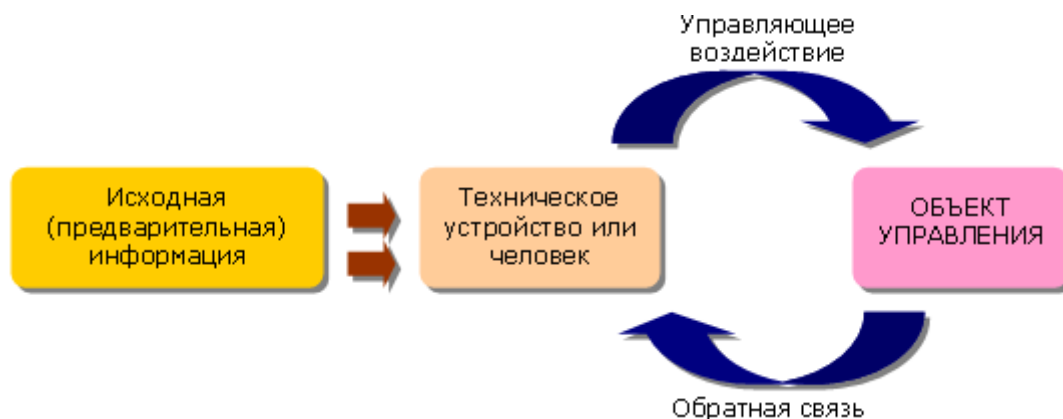


Рис. 1. Замкнутая схема управления

Рассмотрим пример. Процесс обучения в школе построен по замкнутой схеме управления. Ученики являются объектами управления. Учитель перед началом урока обладает определенной исходной информацией: знаниями по предмету, знаниями об учениках. Эти знания позволяют ему так построить урок, чтобы ученики поняли новый материал. Применяя различные методы ведения урока, учитель оказывает на учеников управляющее воздействие. В процессе опроса учеников, что равносильно обратной связи, учитель делает вывод о том, как усвоен материал, и решает, что ему дальше делать — либо провести дополнительное разъяснение, либо дать новый материал. Он должен постоянно отслеживать текущую информацию, чтобы видеть, как реагируют ученики (объект управления) на его воздействия.

Не всегда управление осуществляется по замкнутой схеме. Например, управление потоком автомобилей и пешеходов с помощью светофора является примером незамкнутой (разомкнутой) схемы управления. Светофор не может воспринять корректирующую информацию, он выступает в роли устройства, которое только выдает управляющее воздействие. Изменение цветов светофора — управляющие сигналы. Автомобили и пешеходы выступают в качестве объектов управления.

Такой процесс получил название незамкнутого процесса управления и в виде схемы представлен на рисунке 2. В отличие от схемы на рисунке 1 в этой схеме отсутствует обратная связь — данные о состоянии объекта управления.

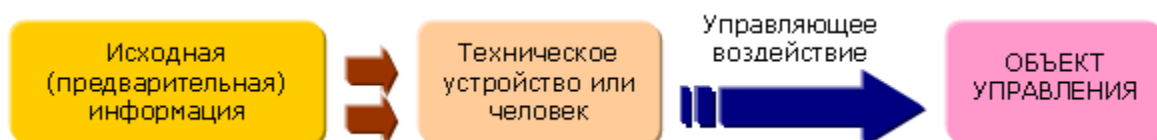


Рис. 2. Разомкнутая схема управления

В зависимости от степени участия человека в процессе управления системы управления делятся на три класса:

- автоматические,
- неавтоматические,
- автоматизированные.

В системах автоматического управления все процессы, связанные с получением информации о состоянии управляемого объекта, обработкой этой информации, формированием управляющих сигналов и пр., осуществляются автоматически в соответствии с представленной на рис. 1 замкнутой схемой управления. В подобных системах не требуется непосредственное участие человека. Системы автоматического управления используются на космических спутниках,

на опасном для здоровья человека производстве, в ткацкой и литейной промышленности, в хлебопекарнях, при поточном производстве, например при изготовлении микросхем, и пр.

В неавтоматических системах управления человек сам оценивает состояние объекта управления и на основе этой оценки воздействует на него. С такими системами вы сталкиваетесь постоянно в школе и дома. Дирижер управляет оркестром, исполняющим музыкальное произведение. Учитель на уроке управляет классом в процессе обучения.

В автоматизированных системах управления сбор и обработка информации, необходимой для выработки управляющих воздействий, осуществляется автоматически, при помощи аппаратуры и компьютерной техники, а решение по управлению принимает человек. Например, рабочий металлорежущего станка производит его установку и включение, остальные процессы выполняются автоматически. Автоматизированная система продажи железнодорожных или авиационных билетов, льготных проездных билетов в метрополитене работает под управлением человека, который запрашивает у компьютера необходимую информацию и на ее основе принимает решение о продаже.

4. Представление и кодирование информации.

Язык как знаковая система

Для обмена информацией с другими людьми человек использует естественные языки (русский, английский, китайский и др.), то есть информация представляется с помощью естественных языков. В основе языка лежит алфавит, то есть набор символов (знаков), которые человек различает по их начертанию. В основе русского языка лежит кириллица, содержащая 33 знака, английский язык использует латиницу (26 знаков), китайский язык использует алфавит из десятков тысяч знаков (иероглифов).

Последовательности символов алфавита в соответствии с правилами грамматики образуют основные объекты языка — слова. Правила, согласно которым образуются предложения из слов данного языка, называются синтаксисом. Необходимо отметить, что в естественных языках грамматика и синтаксис языка формулируются с помощью большого количества правил, из которых существуют исключения, так как такие правила складывались исторически.

Наряду с естественными языками были разработаны формальные языки (системы счисления, язык алгебры, языки программирования и др.). Основное отличие формальных языков от естественных состоит в наличии строгих правил грамматики и синтаксиса.

Например, системы счисления можно рассматривать как формальные языки, имеющие алфавит (цифры) и позволяющие не только именовать и записывать объекты (числа), но и выполнять над ними арифметические операции по строго определенным правилам.

Некоторые языки используют в качестве знаков не буквы и цифры, а другие символы, например химические формулы, ноты, изображения элементов электрических или логических схем, дорожные знаки, точки и тире (код азбуки Морзе) и др.

Знаки могут иметь различную физическую природу. Например, для представления информации с использованием языка в письменной форме используются знаки, которые являются изображениями на бумаге или других носителях, в устной речи в качестве знаков языка используются различные звуки (фонемы), а при обработке текста на компьютере знаки представляются в форме последовательностей электрических импульсов (компьютерных кодов).

Представление информации в живых организмах

Человек воспринимает информацию об окружающем мире с помощью органов чувств (зрения, слуха, обоняния, осязания и вкуса). Чувствительные нервные окончания органов чувств (рецепторы) воспринимают воздействие (например, на глазном дне колбочки и палочки реагируют на воздействие световых лучей) и передают его нейронам (нервным клеткам), цепи которых составляют нервную систему.

Нейрон может находиться в одном из двух состояний: невозбужденном и возбужденном. Возбужденный нейрон генерирует электрический импульс, который передается по нервной системе.

Состояния нейрона (нет импульса, есть импульс) можно рассматривать как знаки некоторого алфавита нервной системы, с помощью которого происходит передача информации.

Генетическая информация во многом определяет строение и развитие живых организмов и передается по наследству.

Хранится генетическая информация в клетках организмов в структуре молекул ДНК. Молекула ДНК состоит из двух скрученных друг с другом в спираль цепей, построенных из четырех нуклеотидов: А, G, Т и С, которые образуют генетический алфавит.

Молекула ДНК человека включает в себя около 3 миллиардов пар нуклеотидов и поэтому в ней закодирована вся информация об организме человека: его внешность, здоровье или предрасположенность к болезням, способности и пр.

Кодирование информации

Представление информации происходит в различных формах в процессе восприятия окружающей среды живыми организмами и человеком, в процессах обмена информацией между человеком и человеком, человеком и компьютером, компьютером и компьютером и так далее. Преобразование информации из одной формы представления (знаковой системы) в другую называется кодированием.

Средством кодирования служит таблица соответствия знаковых систем, которая устанавливает взаимно однозначное соответствие между знаками или группами знаков двух различных знаковых систем.

В процессе обмена информацией часто приходится производить операции кодирования и декодирования информации. При вводе знака алфавита в компьютер путем нажатия соответствующей клавиши на клавиатуре происходит кодирование знака, то есть преобразование его в компьютерный код. При выводе знака на экран монитора или принтер происходит обратный процесс - декодирование, когда из компьютерного кода знак преобразуется в его графическое изображение.

Двоичное кодирование информации

В компьютере для представления информации используется двоичное кодирование, так как удалось создать надежно работающие технические устройства, которые могут со стопроцентной надежностью сохранять и распознавать не более двух различных состояний (цифр):

- электромагнитные реле (замкнуто/разомкнуто), широко использовались в конструкциях первых ЭВМ;
- участок поверхности магнитного носителя информации (намагничен/размагничен);
- участок поверхности лазерного диска (отражает/не отражает);

- триггер (см. п. 3.7.3), может устойчиво находиться в одном из двух состояний, широко используется в оперативной памяти компьютера.

Все виды информации в компьютере кодируются на машинном языке, в виде логических последовательностей нулей и единиц.

Цифры двоичного кода можно рассматривать как два равновероятных состояния (события). При записи двоичной цифры реализуется выбор одного из двух возможных состояний (одной из двух цифр) и, следовательно, она несет количество информации, равное 1 биту.

Даже сама единица измерения количества информации бит (bit) получила свое название от английского словосочетания Binary digit (двоичная цифра).

Важно, что каждая цифра машинного двоичного кода несет информацию в 1 бит. Таким образом, две цифры несут информацию в 2 бита, три цифры — в 3 бита и так далее. Количество информации в битах равно количеству цифр двоичного машинного кода.

Представление числовой информации с помощью систем счисления

Для записи информации о количестве объектов используются числа. Числа записываются с использованием особых знаковых систем, которые называются системами счисления. Алфавит систем счисления состоит из символов, которые называются цифрами. Например, в десятичной системе счисления числа записываются с помощью десяти всем хорошо известных цифр: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

Все системы счисления делятся на две большие группы: позиционные и непозиционные системы счисления. В позиционных системах счисления значение цифры зависит от ее положения в числе, а в непозиционных — не зависит.

Римская непозиционная система счисления. Самой распространенной из непозиционных систем счисления является римская. В качестве цифр в ней используются: I (1), V (5), X (10), L (50), C (100), D (500), M (1000).

Значение цифры не зависит от ее положения в числе. Например, в числе XXX (30) цифра X встречается трижды и в каждом случае обозначает одну и ту же величину - число 10, три числа по 10 в сумме дают 30.

Позиционные системы счисления. Первая позиционная система счисления была придумана еще в Древнем Вавилоне, причем вавилонская нумерация была шестидесятеричной, то есть в ней использовалось шестьдесят цифр! Интересно, что до сих пор при измерении времени мы используем основание, равное 60 (в 1 минуте содержится 60 секунд, а в 1 часе - 60 минут).

В XIX веке довольно широкое распространение получила двенадцатеричная система счисления. До сих пор мы часто употребляем дюжину (число 12): в сутках две дюжины часов, круг содержит тридцать дюжин градусов и так далее.

В позиционных системах счисления количественное значение цифры зависит от ее позиции в числе. Наиболее распространенными в настоящее время позиционными системами счисления являются десятичная, двоичная, восьмеричная и шестнадцатеричная. Каждая позиционная система имеет определенный алфавит цифр и основание.

Двоичное кодирование текстовой информации

Начиная с конца 60-х годов, компьютеры все больше стали использоваться для обработки текстовой информации и в настоящее время большая часть персональных компьютеров в мире (и наибольшее время) занято обработкой именно текстовой информации.

Традиционно для кодирования одного символа используется количество информации, равное 1 байту, то есть $I = 1 \text{ байт} = 8 \text{ битов}$.

Для кодирования одного символа требуется 1 байт информации.

Если рассматривать символы как возможные события, то можно вычислить, какое количество различных символов можно закодировать: $N = 2^1 = 2^8 = 256$.

Такое количество символов вполне достаточно для представления текстовой информации, включая прописные и строчные буквы русского и латинского алфавита, цифры, знаки, графические символы и пр.

Кодирование заключается в том, что каждому символу ставится в соответствие уникальный десятичный код от 0 до 255 или соответствующий ему двоичный код от 00000000 до 11111111. Таким образом, человек различает символы по их начертаниям, а компьютер - по их кодам.

При вводе в компьютер текстовой информации происходит ее двоичное кодирование, изображение символа преобразуется в его двоичный код. Пользователь нажимает на клавиатуре клавишу с символом, и в компьютер поступает определенная последовательность из восьми электрических импульсов (двоичный код символа). Код символа хранится в оперативной памяти компьютера, где занимает один байт.

В процессе вывода символа на экран компьютера производится обратный процесс — декодирование, то есть преобразование кода символа в его изображение.

Важно, что присвоение символу конкретного кода — это вопрос соглашения, которое фиксируется в кодовой таблице. Первые 33 кода (с 0 по 32) соответствуют не символам, а операциям (перевод строки, ввод пробела и так далее). Коды с 33 по 127 являются интернациональными и соответствуют символам латинского алфавита, цифрам, знакам арифметических операций и знакам препинания. Коды с 128 по 255 являются национальными, то есть в национальных кодировках одному и тому же коду соответствуют различные символы.

К сожалению, в настоящее время существуют пять различных кодовых таблиц для русских букв (КОИ8, CP1251, CP866, Mac, ISO), поэтому тексты, созданные в одной кодировке, не будут правильно отображаться в другой.

В настоящее время широкое распространение получил новый международный стандарт Unicode, который отводит на каждый символ не один байт, а два, поэтому с его помощью можно закодировать не 256 символов, а $N = 2^{16} = 65536$ различных символов.

Аналоговый и дискретный способы представления изображений и звука

Человек способен воспринимать и хранить информацию в форме образов (зрительных, звуковых, осязательных, вкусовых и обонятельных). Зрительные образы могут быть сохранены в виде изображений (рисунков, фотографий и так далее), а звуковые - зафиксированы на пластинках, магнитных лентах, лазерных дисках и так далее.

Информация, в том числе графическая и звуковая, может быть представлена в аналоговой или дискретной форме. При аналоговом представлении физическая величина принимает бесконечное множество значений, причем ее значения изменяются непрерывно. При дискретном

представлении физическая величина принимает конечное множество значений, причем ее величина изменяется скачкообразно.

Примером аналогового представления графической информации может служить, например, живописное полотно, цвет которого изменяется непрерывно, а дискретного — изображение, напечатанное с помощью струйного принтера и состоящее из отдельных точек разного цвета. Примером аналогового хранения звуковой информации является виниловая пластинка (звуковая дорожка изменяет свою форму непрерывно), а дискретного — аудиокompакт-диск (звуковая дорожка которого содержит участки с различной отражающей способностью).

Преобразование графической и звуковой информации из аналоговой формы в дискретную производится путем дискретизации, то есть разбиения непрерывного графического изображения и непрерывного (аналогового) звукового сигнала на отдельные элементы. В процессе дискретизации производится кодирование, то есть присвоение каждому элементу конкретного значения в форме кода. Дискретизация — это преобразование непрерывных изображений и звука в набор дискретных значений в форме кодов.

Двоичное кодирование графической информации

В процессе кодирования изображения производится его пространственная дискретизация. Пространственную дискретизацию изображения можно сравнить с построением изображения из мозаики (большого количества маленьких разноцветных стекол). Изображение разбивается на отдельные маленькие фрагменты (точки), причем каждому фрагменту присваивается значение его цвета, то есть код цвета (красный, зеленый, синий и так далее).

Качество кодирования изображения зависит от двух параметров. Во-первых, качество кодирования изображения тем выше, чем меньше размер точки и соответственно большее количество точек составляет изображение. Во-вторых, чем большее количество цветов, то есть большее количество возможных состояний точки изображения, используется, тем более качественно кодируется изображение (каждая точка несет большее количество информации). Совокупность используемых в наборе цветов образует палитру цветов.

Графическая информация на экране монитора представляется в виде растрового изображения, которое формируется из определенного количества строк, которые в свою очередь содержат определенное количество точек (пикселей).

Качество изображения определяется разрешающей способностью монитора, т.е. количеством точек, из которых оно складывается. Чем больше разрешающая способность, то есть чем больше количество строк раstra и точек в строке, тем выше качество изображения. В современных персональных компьютерах обычно используются три основные разрешающие способности экрана: 800x600, 1024x768 и 1280x1024 точки.

Рассмотрим формирование на экране монитора растрового изображения, состоящего из 600 строк по 800 точек в каждой строке (всего 480 000 точек). В простейшем случае (черно-белое изображение без градаций серого цвета) каждая точка экрана может иметь одно из двух состояний - «черная» или «белая», то есть для хранения ее состояния необходим 1 бит.

Цветные изображения формируются в соответствии с двоичным кодом цвета каждой точки, хранящимся в видеопамяти.. Цветные изображения могут иметь различную глубину цвета, которая задается количеством битов, используемым для кодирования цвета точки. Наиболее распространенными значениями глубины цвета являются 8, 16, 24 или 32 бита.

Цветное изображение на экране монитора формируется за счет смешивания трех базовых цветов: красного, зеленого и синего. Такая цветовая модель называется RGB-моделью по первым буквам английских названий цветов (Red, Green, Blue).

Двоичное кодирование звуковой информации

Временная дискретизация звука. Звук представляет собой звуковую волну с непрерывно меняющейся амплитудой и частотой. Чем больше амплитуда сигнала, тем он громче для человека, чем больше частота сигнала, тем выше тон. Для того чтобы компьютер мог обрабатывать звук, непрерывный звуковой сигнал должен быть превращен в последовательность электрических импульсов (двоичных нулей и единиц).

В процессе кодирования непрерывного звукового сигнала производится его временная дискретизация. Непрерывная звуковая волна разбивается на отдельные маленькие временные участки, причем для каждого такого участка устанавливается определенная величина амплитуды.

При двоичном кодировании непрерывного звукового сигнала он заменяется последовательностью дискретных уровней сигнала. Качество кодирования зависит от количества измерений уровня сигнала в единицу времени, то есть частоты дискретизации. Чем большее количество измерений производится за 1 секунду (чем больше частота дискретизации), тем точнее процедура двоичного кодирования.

Хранение информации

Информация, закодированная с помощью естественных и формальных языков, а также информация в форме зрительных и звуковых образов хранится в памяти человека. Однако для долговременного хранения информации, ее накопления и передачи из поколения в поколение используются носители информации.

Материальная природа носителей информации может быть различной: молекулы ДНК, которые хранят генетическую информацию; бумага, на которой хранятся тексты и изображения; магнитная лента, на которой хранится звуковая информация; фото- и киноплёнки, на которых хранится графическая информация; микросхемы памяти, магнитные и лазерные диски, на которых хранятся программы и данные в компьютере, и так далее.

По оценкам специалистов, объем информации, фиксируемой на различных носителях, превышает один эксабайт в год (10^{18} байт/год). Примерно 80% всей этой информации хранится в цифровой форме на магнитных и оптических носителях и только 20% — на аналоговых носителях (бумага, магнитные ленты, фото- и киноплёнки).

Носители информации характеризуются информационной емкостью, то есть количеством информации, которое они могут хранить. Наиболее информационно емкими являются молекулы ДНК, которые имеют очень малый размер и плотно упакованы. Это позволяет хранить огромное количество информации (до 10^{21} битов в 1 см^3), что дает возможность организму развиваться из одной-единственной клетки, содержащей всю необходимую генетическую информацию.

Современные микросхемы памяти позволяют хранить в 1 см^3 до 10^{10} битов информации, однако это в 100 миллиардов раз меньше, чем в ДНК. Можно сказать, что современные технологии пока существенно проигрывают биологической эволюции.

Однако если сравнивать информационную емкость традиционных носителей информации (книг) и современных компьютерных носителей, то прогресс очевиден. На каждом гибком магнитном диске может храниться книга объемом около 600 страниц, а на жестком магнитном диске или DVD - целая библиотека, включающая десятки тысяч книг.

Большое значение имеет надежность и долговременность хранения информации. Большую устойчивость к возможным повреждениям имеют молекулы ДНК, так как существует механизм обнаружения повреждений их структуры (мутаций) и самовосстановления.

Надежность (устойчивость к повреждениям) достаточно высока у аналоговых носителей, повреждение которых приводит к потере информации только на поврежденном участке. Поврежденная часть фотографии не лишает возможности видеть оставшуюся часть, повреждение участка магнитной ленты приводит лишь к временному пропаданию звука и так далее.

Цифровые носители гораздо более чувствительны к повреждениям, даже потеря одного бита данных на магнитном или оптическом диске может привести к невозможности считать файл, то есть к потере большого объема данных. Именно поэтому необходимо соблюдать правила эксплуатации и хранения цифровых носителей информации.

Наиболее долговременным носителем информации является молекула ДНК, которая в течение десятков тысяч лет (человек) и миллионов лет (некоторые живые организмы), сохраняет генетическую информацию данного вида.

Аналоговые носители способны сохранять информацию в течение тысяч лет (египетские папирусы и шумерские глиняные таблички), сотен лет (бумага) и десятков лет (магнитные ленты, фото- и киноплёнки).

Цифровые носители появились сравнительно недавно и поэтому об их долговременности можно судить только по оценкам специалистов. По экспертным оценкам, при правильном хранении оптические носители способны хранить информацию сотни лет, а магнитные — десятки лет.

5. Функциональная схема компьютера

По своему назначению **компьютер - это универсальный прибор для работы с информацией**. По принципам своего устройства компьютер - это модель человека, работающего с информацией.

Персональный компьютер (ПК) — это компьютер, предназначенный для обслуживания одного рабочего места. По своим характеристикам он может отличаться от больших ЭВМ, но функционально способен выполнять аналогичные операции. По способу эксплуатации различают настольные (desktop), портативные (laptop и notebook) и карманные (palmtop) модели ПК.

Аппаратное обеспечение. Поскольку компьютер предоставляет все три класса информационных методов для работы с данными (аппаратные, программные и естественные), принято говорить о компьютерной системе как о состоящей из аппаратных и программных средств, работающих совместно. Узлы, составляющие аппаратные средства компьютера, называют аппаратным обеспечением. Они выполняют всю физическую работу с данными: регистрацию, хранение, транспортировку и преобразование как по форме, так и по содержанию, а также представляют их в виде, удобном для взаимодействия с естественными информационными методами человека.

Совокупность аппаратных средств компьютера называют его аппаратной конфигурацией.

Программное обеспечение. Программы могут находиться в двух состояниях: активном и пассивном. В пассивном состоянии программа не работает и выглядит как данные, содержательная часть которых - сведения. В этом состоянии содержимое программы можно «читать» с помощью других программ, как читают книги, и изменять. Из него можно узнать назначение программы и принцип ее работы. В пассивном состоянии программы создаются, редактируются, хранятся и транспортируются. Процесс создания и редактирования программ называется программированием.

Когда программа находится в активном состоянии, содержательная часть ее данных рассматривается как команды, согласно которым работают аппаратные средства компьютера.

Чтобы изменить порядок их работы, достаточно прервать исполнение одной программы и начать исполнение другой, содержащей иной набор команд.

Совокупность программ, хранящихся на компьютере, образует его программное обеспечение. Совокупность программ, подготовленных к работе, называют установленным программным обеспечением. Совокупность программ, работающих в тот или иной момент времени, называют программной конфигурацией.

Устройство компьютера. Любой компьютер (даже самый большой) состоит из четырех частей:

- устройства ввода информации
- устройства обработки информации
- устройства хранения
- устройства вывода информации.



Конструктивно эти части могут быть объединены в одном корпусе размером с книгу или же каждая часть может состоять из нескольких достаточно громоздких устройств

Базовая аппаратная конфигурация ПК. Базовой аппаратной конфигурацией персонального компьютера называют минимальный комплект аппаратных средств, достаточный для начала работы с компьютером. С течением времени понятие базовой конфигурации постепенно меняется.

Чаще всего персональный компьютер состоит из следующих устройств:

- Системный блок
- Монитор
- Клавиатура
- Мышь

Дополнительно могут подключаться другие устройства ввода и вывода информации, например звуковые колонки, принтер, сканер...

Системный блок — основной блок компьютерной системы. В нем располагаются устройства, считающиеся внутренними. Устройства, подключаемые к системному блоку снаружи, считаются внешними. Для внешних устройств используют также термин периферийное оборудование.

Монитор — устройство для визуального воспроизведения символьной и графической информации. Служит в качестве устройства вывода. Для настольных ПК в настоящее время наиболее распространены мониторы, основанные на электронно-лучевых трубках. Они отдаленно напоминают бытовые телевизоры.

Клавиатура — клавишное устройство, предназначенное для управления работой компьютера и ввода в него информации. Информация вводится в виде алфавитно-цифровых символьных данных.

Мышь — устройство «графического» управления.

Внутренние устройства персонального компьютера.

Внутренними считаются устройства, располагающиеся в системном блоке. Доступ к некоторым из них имеется на лицевой панели, что удобно для быстрой смены информационных носителей, например гибких магнитных дисков. Разъемы некоторых устройств выведены на заднюю стенку — они служат для подключения периферийного оборудования. К некоторым устройствам системного блока доступ не предусмотрен — для обычной работы он не требуется.

Процессор. Микропроцессор — основная микросхема персонального компьютера. Все вычисления выполняются в ней. Основная характеристика процессора — тактовая частота (измеряется в мегагерцах, МГц). Чем выше тактовая частота, тем выше производительность процессора. Так, например, при тактовой частоте 500 МГц процессор может за одну секунду изменить свое состояние 500 миллионов раз. Для большинства операций одного такта недостаточно, поэтому количество операций, которые процессор может выполнить в секунду, зависит не только от тактовой частоты, но и от сложности операций.

Единственное устройство, о существовании которого процессор «знает от рождения», — оперативная память — с нею он работает совместно. Оттуда поступают данные и команды. Данные копируются в ячейки процессора (они называются регистрами), а потом преобразуются в соответствии с содержанием команд. Более полную картину того, как процессор взаимодействует с оперативной памятью, вы получите в главах, посвященных основам программирования.

Оперативная память. Оперативную память можно представить как обширный массив ячеек, в которых хранятся числовые данные и команды в то время, когда компьютер включен. Объем оперативной памяти измеряется в миллионах байтов — мегабайтах (Мбайт).

Процессор может обратиться к любой ячейке оперативной памяти (байту), поскольку она имеет неповторимый числовой адрес. Обратиться к индивидуальному биту оперативной памяти процессор не может, так как у бита нет адреса. В то же время, процессор может изменить состояние любого бита, но для этого требуется несколько действий.

Материнская плата. Материнская плата — это самая большая плата персонального компьютера. На ней располагаются магистрали, связывающие процессор с оперативной памятью, — так называемые шины. Различают шину данных, по которой процессор копирует данные из ячеек памяти, адресную шину, по которой он подключается к конкретным ячейкам памяти, и шину команд, по которой в процессор поступают команды из программ. К шинам материнской платы подключаются также все прочие внутренние устройства компьютера. Управляет работой материнской платы микропроцессорный набор микросхем — так называемый чипсет.

Видеоадаптер. Видеоадаптер — внутреннее устройство, устанавливаемое в один из разъемов материнской платы. В первых персональных компьютерах видеоадаптеров не было. Вместо них в оперативной памяти отводилась небольшая область для хранения видеоданных. Специальная микросхема (видеоконтроллер) считывала данные из ячеек видеопамати и в соответствии с ними управляла монитором.

По мере улучшения графических возможностей компьютеров область видеопамати отделили от основной оперативной памяти и вместе с видеоконтроллером выделили в отдельный прибор, который назвали видеоадаптером. Современные видеоадаптеры имеют собственный вычислительный процессор (видеопроцессор), который снизил нагрузку на основной процессор при построении сложных изображений. Особенно большую роль видеопроцессор играет при

построении на плоском экране трехмерных изображений. В ходе таких операций ему приходится выполнять особенно много математических расчетов.

В некоторых моделях материнских плат функции видеоадаптера выполняют микросхемы чипсета — в этом случае говорят, что видеоадаптер интегрирован с материнской платой. Если же видеоадаптер выполнен в виде отдельного устройства, его называют видеокартой. Разъем видеокарты выведен на заднюю стенку. К нему подключается монитор.

Звуковой адаптер. Для компьютеров IBM PC работа со звуком изначально не была предусмотрена. Первые десять лет существования компьютеры этой платформы считались офисной техникой и обходились без звуковых устройств. В настоящее время средства для работы со звуком считаются стандартными. Для этого на материнской плате устанавливается звуковой адаптер. Он может быть интегрирован в чипсете материнской платы или выполнен как отдельная подключаемая плата, которая называется звуковой картой.

Разъемы звуковой карты выведены на заднюю стенку компьютера. Для воспроизведения звука к ним подключают звуковые колонки или наушники. Отдельный разъем предназначен для подключения микрофона. При наличии специальной программы это позволяет записывать звук. Имеется также разъем (линейный выход) для подключения к внешней звукозаписывающей или звуковоспроизводящей аппаратуре (магнитофонам, усилителям и т.п.).

Жесткий диск. Поскольку оперативная память компьютера очищается при отключении питания, необходимо устройство для длительного хранения данных и программ. В настоящее время для этих целей широко применяют так называемые жесткие диски.

Принцип действия жесткого диска основан на регистрации изменений магнитного поля вблизи записывающей головки.

Основным параметром жесткого диска является емкость, измеряемая в гигабайтах (миллиардах байтов), Гбайт. Средний размер современного жесткого диска составляет 80 — 160 Гбайт, причем этот параметр неуклонно растет.

Дисковод гибких дисков. Для транспортировки данных между удаленными компьютерами используют так называемые гибкие диски. Стандартный гибкий диск (дискета) имеет сравнительно небольшую емкость 1,44 Мбайт. По современным меркам этого совершенно недостаточно для большинства задач хранения и транспортировки данных, но низкая стоимость носителей и высокая степень готовности к работе сделали гибкие диски самыми распространенными носителями данных.

Для записи и чтения данных, размещенных на гибких дисках, служит специальное устройство — дисковод. Приемное отверстие дисковода выведено на лицевую панель системного блока.

Дисковод CD-ROM. Для транспортировки больших объемов данных удобно использовать компакт-диски CD-ROM. Эти диски позволяют только читать ранее записанные данные — производить запись на них нельзя. Емкость одного диска составляет порядка 650-700 Мбайт.

Для чтения компакт-дисков служат дисководы CD-ROM. Основным параметром дисковода CD-ROM — скорость чтения. Она измеряется в кратных единицах. За единицу принята скорость чтения, утвержденная в середине 80-х гг. для музыкальных компакт-дисков (аудиодисков). Современные дисководы CD-ROM обеспечивают скорость чтения 40x - 52x.

Основной недостаток дисководов CD-ROM — невозможность записи дисков — преодолен в современных устройствах однократной записи — CD-R. Существуют также устройства CD-RW, позволяющие осуществлять многократную запись.

Принцип хранения данных на компакт-дисках не магнитный, как у гибких дисков, а оптический.

Коммуникационные порты. Для связи с другими устройствами, например принтером, сканером, клавиатурой, мышью и т. п., компьютер оснащается так называемыми портами. Порт — это не просто разъем для подключения внешнего оборудования, хотя порт и заканчивается разъемом. Порт — более сложное устройство, чем просто разъем, имеющее свои микросхемы и управляемое программно.

Сетевой адаптер. Сетевые адаптеры необходимы компьютерам, чтобы они могли обмениваться данными между собой. Этот прибор следит за тем, чтобы процессор не подал новую порцию данных на внешний порт, пока сетевой адаптер соседнего компьютера не скопировал к себе предыдущую порцию. После этого процессору дается сигнал о том, что данные забраны и можно подавать новые. Так осуществляется передача.

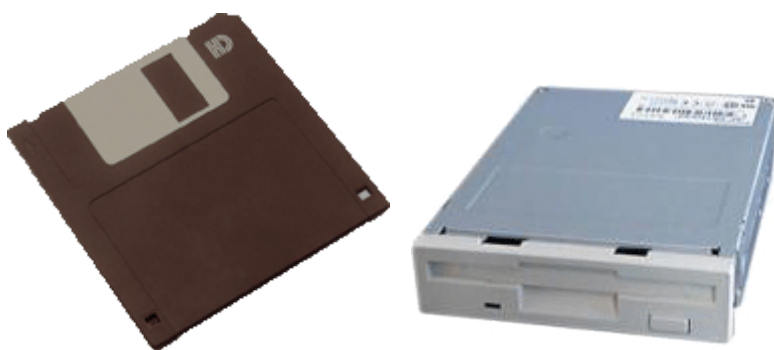
Когда сетевой адаптер «узнает» от соседнего адаптера, что у того есть порция данных, он копирует их к себе, а потом проверяет, ему ли они адресованы. Если да, он передает их процессору. Если нет, он выставляет их на выходной порт, откуда их заберет сетевой адаптер очередного соседнего компьютера. Так данные перемещаются между компьютерами до тех пор, пока не попадут к адресату. Сетевые адаптеры могут быть встроены в материнскую плату, но чаще устанавливаются отдельно, в виде дополнительных плат, называемых сетевыми картами.

6. Устройства памяти компьютера. Носители информации (гибкие диски, жесткие диски, диски CD-ROM/R/RW, DVD и др.)

Основной функцией внешней памяти компьютера является способность долговременно хранить большой объем информации (программы, документы, аудио-и видеоклипы и т. д.). Устройство, которое обеспечивает запись/считывание информации, называется накопителем или дисководом, а хранится информация на носителях (например, дискетах).

В накопителях на гибких магнитных дисках (НГМД или дискетах) и накопителях на жестких магнитных дисках (НЖМД или винчестерах), в основу записи, хранения и считывания информации положен магнитный принцип, а в лазерных дисководах — оптический принцип.

Гибкие магнитные диски.



Гибкие магнитные диски помещаются в пластмассовый корпус. Такой носитель информации называется дискетой. Дискета вставляется в дисковод, вращающий диск с постоянной угловой скоростью. Магнитная головка дисковода устанавливается на определенную концентрическую дорожку диска, на которую и записывается (или считывается) информация.

Информационная ёмкость дискеты невелика и составляет всего 1.44 Мбайт. Скорость записи и считывания информации также мала (около 50 Кбайт/с) из-за медленного вращения диска (360 об./мин).

В целях сохранения информации гибкие магнитные диски следует предохранять от воздействия сильных магнитных полей и нагревания, так как это может привести к размагничиванию носителя и потере информации.

Жесткие магнитные диски.

Жесткий диск (HDD — Hard Disk Drive) относится к несменным дисковым магнитным накопителям. Первый жесткий диск был разработан фирмой IBM в 1973 г. и имел емкость 16 Кбайт.



Жесткие магнитные диски представляют собой несколько десятков дисков, размещенных на одной оси, заключенных в металлический корпус и вращающихся с высокой угловой скоростью. За счет множества дорожек на каждой стороне дисков и большого количества дисков информационная емкость жестких дисков может в десятки тысяч раз превышать информационную емкость дискет и достигать сотен Гбайт. Скорость записи и считывания информации с жестких дисков достаточно велика (около 133 Мбайт/с) за счет быстрого вращения дисков (7200 об./мин).

Часто жесткий диск называют винчестер. Бытует легенда, объясняющая, почему за жесткими дисками повелось такое причудливое название. Первый жесткий диск, выпущенный в Америке в начале 70-х годов, имел емкость по 30 Мб информации на каждой рабочей поверхности. В то же время, широко известная в той же Америке магазинная винтовка О. Ф. Винчестера имела калибр - 0.30; может грохотал при своей работе первый винчестер как автомат или порохом от него пахло - не ясно, но с той поры стали называть жесткие диски винчестерами.



В процессе работы компьютера случаются сбои. Вирусы, перебои энергоснабжения, программные ошибки - все это может послужить причиной повреждения информации, хранящейся на Вашем жестком диске. Повреждение информации далеко не всегда означает ее потерю, так что полезно знать о том, как она хранится на жестком диске, ибо тогда ее можно восстановить. Тогда, например, в случае повреждения вирусом загрузочной области, вовсе не обязательно форматировать весь диск (!), а, восстановив поврежденное место, продолжить нормальную работу с сохранением всех своих бесценных данных.

В жестких дисках используются достаточно хрупкие и миниатюрные элементы. Чтобы сохранить информацию и работоспособность жестких дисков, необходимо оберегать их от ударов и резких изменений пространственной ориентации в процессе работы.

Лазерные дисководы и диски.

В начале 80-х годов голландская фирма «Philips» объявила о совершенной ею революцией в области звуковоспроизведения. Ее инженеры придумали то, что сейчас пользуется огромной популярностью - Это лазерные диски и проигрыватели.

За последние несколько лет компьютерные устройства для чтения компакт-дисков (CD), называемые CD-ROM, стали практически необходимой частью любого компьютера. Это произошло потому, что разнообразные программные продукты стали занимать значительное количество места, и поставка их на дискетах оказалась чрезмерно дорогостоящей и ненадёжной. Поэтому их стали поставлять на CD (таких же, как и обычные музыкальные).

Лазерные дисководы используют оптический принцип чтения информации. На лазерных дисках CD (CD — Compact Disk, компакт диск) и DVD (DVD — Digital Video Disk, цифровой видеодиск) информация записана на одну спиралевидную дорожку (как на грампластинке), содержащую чередующиеся участки с различной отражающей способностью. Лазерный луч падает на поверхность вращающегося диска, а интенсивность отраженного луча зависит от отражающей способности участка дорожки и приобретает значения 0 или 1. Для сохранности информации лазерные диски надо предохранять от механических повреждений (царапин), а также от загрязнения.

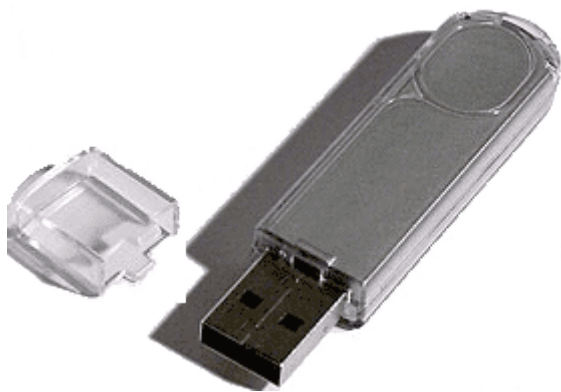
На лазерных дисках хранится информация, которая была записана на них в процессе изготовления. Запись на них новой информации невозможна. Производятся такие диски путем штамповки. Существуют CD-R и DVD-R диски информация на которые может быть записана только один раз. На дисках CD-RW и DVD-RW информация может быть записана/перезаписана многократно. Диски разных видов можно отличить не только по маркировке, но и по цвету отражающей поверхности.

Запись на CD и DVD при помощи обычных CD-ROM и DVD-ROM невозможна. Для этого необходимы устройства CD-RW и DVD-RW с помощью которых возможны чтение-однократная запись и чтение-запись-перезапись. Эти устройства обладают достаточно мощным лазером, позволяющем менять отражающую способность участков поверхности в процессе записи диска.

Информационная ёмкость CD-ROM достигает 700 Мбайт, а скорость считывания информации (до 7.8 Мбайт/с) зависит от скорости вращения диска. DVD-диски имеют гораздо большую информационную ёмкость (однослойный односторонний диск - 4.7 Гбайт) по сравнению с CD-дисками, т.к. используются лазеры с меньшей длиной волны, что позволяет размещать оптические дорожки более плотно. Так же существуют двухслойные DVD-диски и двухсторонние DVD-диски. В настоящее время скорости считывания 16-скоростных DVD-дисководов достигает 21 Мбайт/с.

Устройства на основе flash-памяти.





Flash-память - это энергонезависимый тип памяти, позволяющий записывать и хранить данные в микросхемах. Устройства на основе flash-памяти не имеют в своём составе движущихся частей, что обеспечивает высокую сохранность данных при их использовании в мобильных устройствах.

Flash-память представляет собой микросхему, помещенную в миниатюрный корпус. Для записи или считывания информации накопители подключаются к компьютеру через USB-порт. Информационная емкость карт памяти достигает 1024 Мбайт.

Тип носителя	Емкость носителя	Скорость обмена данными (Мбайт/с)	Опасные воздействия
НГМД 3,5"	1,44Мб	0,05	Магнитные поля, нагревание, физическое воздействие
НЖМД	сотни Гбайт	около 133	Удары, изменение пространственной ориентации в процессе работы
CD-ROM	650-800Мбайт	до 7,8	Царапины, загрязнение
DVD-ROM	до 17Гбайт	до 21	
Устройства на основе flash-памяти	до 1024 Мбайт	USB 1.0 - 1,5 USB 1.1 - 12 USB 2.0 - 480	Перенапряжение питания

7. Программное обеспечение компьютера (системное и прикладное).

Совокупность программ, хранящихся на компьютере, образует его программное обеспечение.

Совокупность используемых в компьютере программ принято называть программным обеспечением. Программное обеспечение создает на компьютере определенную среду для работы

и включает в себя инструментарий, с помощью которого вы имеете возможность создавать любые компьютерные объекты. Разнообразие сред определяете составом программного обеспечения компьютера, так как любая, даже самая небольшая программа после ее запуска создает свою рабочую среду.

Программное обеспечение компьютера — совокупность все используемых в компьютере программ.

В жизни все объекты можно сгруппировать по определенным знакам и составить для себя представление о том, где можно использовать того или иного представителя данной группы (класса). То же самое можно сделать и по отношению к компьютерным программам.

Для того чтобы ясно понимать, где и какую программу вам лучше использовать для преобразования информации и получения желаемого результата, необходимо иметь представление об имеющихся разновидностях программ. Все программное обеспечение принято разделять на три класса:

- системное,
- прикладное,
- инструментарий программирования (системы программирования).

Системное программное обеспечение

Этот класс программного обеспечения является необходимой принадлежностью компьютера, так как обеспечивает взаимодействие человека, всех устройств и программ компьютера.

Этот комплекс программ определяет на компьютере системную среду и правила работы в ней. Чем более совершенно системное программное обеспечение, тем комфортнее мы чувствуем себя в системной среде.

Самой важной системной программой является операционная система, которая обычно хранится жестком диске. При включении компьютера ее основная часть переписывается с жесткого диска во внутреннюю память и там находится на протяжении всего сеанса работы компьютера.

Операционная система - это набор программ, управляющих оперативной памятью, процессором, внешними устройствами и файлами; ведущих диалог с пользователем.

Важной частью операционной системой является файловая система ОС. В файлах хранится все: и программное обеспечение, и информация, необходимая для пользователя. С файлами постоянно приходится что-то делать: создавать, удалять, копировать, перемещать, искать и переименовывать. За все эти действия и отвечает файловая система.

Если вы включили компьютер и при этом на экране не происходит никаких изменений, хотя все устройства находятся рабочем состоянии, то это говорит об отсутствии в нем операционной системы.

Операционная система обеспечивает:

- выполнение прикладных программ;
- управление ресурсами компьютера — памятью, процессором и всеми внешними устройствами;
- контакт человека с компьютером.

К наиболее известным операционным системам относятся: MS-DOS, Windows, Unix, OS/2.

К системному ПО можно кроме ОС отнести и множество программ обслуживающего, сервисного характера.

Прикладное программное обеспечение

Все имеющиеся на компьютер прикладные программы составляют прикладное программное обеспечение. Оно определяет на компьютере прикладную среду правила работы в ней. Прикладная среда всегда является «дружественной» по отношению любому человеку, овладевшим несложными приемами работы в ней. Прикладные программы могут работать на компьютере только при условии, что на компьютере уже установлена операционная система.

Каждая прикладная среда предназначена для создания и исследования определенного вида компьютерного объекта. Например, для создания графического объекта предназначена среда графического редактора, для работы с текстом — среда текстового процессора и т. д.

Комплекс прикладных программ в среде операционной системы Windows называют приложением. Нередко его называют также пакетом прикладных программ (ППП).

Наибольшей популярностью пользуются следующие группы прикладного программного обеспечения:

- текстовые процессоры — для создания текстовых документов;
- табличные процессоры (электронные таблицы) — для вычислений и анализа информации, представленной в табличной форме;
- базы данных — для организации и управления данными;
- графические пакеты — для представления информации в виде рисунков и графиков; » коммуникационные программы — для обмена информацией между компьютерами;
- интегрированные пакеты, включающие несколько прикладных программ разного назначения;
- обучающие программы, электронные учебники, словари, энциклопедии, системы проектирования и дизайна;
- игры.

Инструментарий программирования.

Этот класс программ предназначен для создания системного и прикладного программного обеспечения. Методы работы с инструментарием программирования определяются той средой, в которой осуществляется преобразование алгоритма в программу для компьютера.

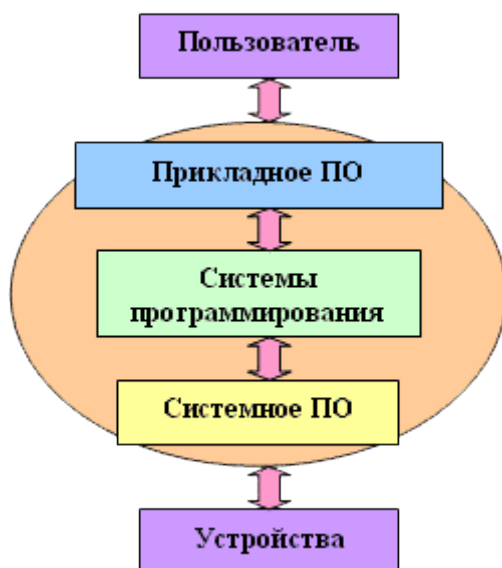
Базовые инструменты любой среды программирования совершенно одинаковы по своей сути, а отличаются только формой представления.

Представьте себе набор типовых инструментов любого специалиста, будь то слесарь, столяр, портниха, электромонтер, мастер по ремонту автомашин и пр. Их инструменты существенно отличаются друг от друга, так как служат для решения различных задач. У людей одной и той же профессии базовые наборы инструментов очень похожи и отличаются только формой, качеством, маркой.

Аналогичная картина складывается и с инструментарием программирования. Он может быть разнообразным, но всегда будет существовать некий базовый набор инструментов, для использования которого нужно овладеть специальным языком программирования.

Для создания прикладного ПО широко используются такие языки, как Basic, Pascal, C++, Delphi и др. Во многих учебных заведениях используется язык ЛОГО.

Взаимосвязь программного обеспечения (уровни программной конфигурации)



8. Назначение и состав операционной системы компьютера. Загрузка компьютера.

Для того чтобы мы могли не думать о том, как в компьютере происходит работа процессора с программами, данными и с аппаратными устройствами, существует специальный комплекс программ, называемых операционной системой.

Операционные системы разные, но их назначение и функции одинаковые. Операционная система является базовой и необходимой составляющей ПО компьютера, без нее компьютер не может работать в принципе.

Операционная система – комплекс программ, обеспечивающих взаимодействие всех аппаратных и программных частей компьютера между собой и взаимодействие пользователя и компьютера.

Операционная система обеспечивает связь между пользователем, программами и аппаратными устройствами.

Структура операционной системы:

1. Ядро – переводит команды с языка программ на язык «машинных кодов», понятный компьютеру.
2. Драйверы – программы, управляющие устройствами.
3. Интерфейс – оболочка, с помощью которой пользователь общается с компьютером.

Операционная система обеспечивает совместное функционирование всех устройств компьютера и предоставляет пользователю доступ к его ресурсам.

Процесс работы компьютера в определенном смысле сводится к обмену файлами между устройствами. В операционной системе имеются программные модули, управляющие файловой системой.

В состав операционной системы входит специальная программа — командный процессор, которая запрашивает у пользователя команды и выполняет их. Пользователь может дать, например, команду выполнения какой-либо операции над файлами (копирование, удаление, переименование), команду вывода документа на печать и т. д. Операционная система должна эти команды выполнить.

К магистрали компьютера подключаются различные устройства (дискетоды, монитор, клавиатура, мышь, принтер и др.). В состав операционной системы входят драйверы устройств — специальные программы, которые обеспечивают управление работой устройств и согласование информационного обмена с другими устройствами. Любому устройству соответствует свой драйвер.

Для упрощения работы пользователя в состав современных операционных систем, и в частности в состав Windows, входят программные модули, создающие графический пользовательский интерфейс. В операционных системах с графическим интерфейсом пользователь может вводить команды посредством мыши, тогда как в режиме командной строки необходимо вводить команды с помощью клавиатуры.

Операционная система содержит также сервисные программы, или утилиты. Такие программы позволяют обслуживать диски (проверять, сжимать, дефрагментировать и т. д.), выполнять операции с файлами (архивировать и т. д.), работать в компьютерных сетях и т. д.

Для удобства пользователя в операционной системе обычно имеется и справочная система. Она предназначена для оперативного получения необходимой информации о функционировании как операционной системы в целом, так и о работе ее отдельных модулей.

Запуск компьютера

При поступлении сигнала о запуске процессор обращается к специально выделенной ячейке памяти. В ОЗУ в этот момент ничего нет, если бы там была какая-либо программа, то она начала бы выполняться.

Для того чтобы компьютер мог начать работу необходимо наличие специальной микросхемы – ПЗУ. Программы ПЗУ записываются на заводе и называются BIOS.

После включения компьютера процессор начинает считывать и выполнять микрокоманды, которые хранятся в микросхеме BIOS. Прежде всего начинает выполняться программа тестирования POST, которая проверяет работоспособность основных устройств компьютера. В случае неисправности выдаются определенные звуковые сигналы, а после инициализации видеоадаптера процесс тестирования отображается на экране монитора.

Затем BIOS начитает поиск программы-загрузчика операционной системы. Программа-загрузчик помещается в ОЗУ и начинается процесс загрузки файлов операционной системы.

Загрузка операционной системы

Файлы операционной системы хранятся во внешней, долговременной памяти (на жестком диске, на CD ...). Однако программы могут выполняться, только если они находятся в ОЗУ, поэтому файлы ОС необходимо загрузить в оперативную память.

Диск, на котором находятся файлы операционной системы и с которого происходит загрузка, называют системным. Если системные диски в компьютере отсутствуют, на экране монитора появляется сообщение "Non system disk" и компьютер «зависает», т. е. загрузка операционной системы прекращается и компьютер остается неработоспособным.

После окончания загрузки операционной системы управление передается командному процессору. В случае использования интерфейса командной строки на экране появляется приглашение системы для ввода команд, в противном случае загружается графический интерфейс операционной системы. В случае загрузки графического интерфейса операционной системы команды могут вводиться с помощью мыши.

Компьютеры без операционных систем

Первые персональные компьютеры не имели операционных систем и были похожи на современные игровые приставки. При включении компьютера в сеть процессор обращался к постоянной памяти (ПЗУ), в котором была записана программа поддержки несложного языка программирования, например языка БЕЙСИК или похожего. Первичное изучение команд этого языка обычно занимало не более нескольких часов, и вскоре на компьютере можно было набирать и запускать несложные программы. Подключив к компьютеру магнитофон, можно было загрузить постороннюю программу. Загруженная программа отключала ПЗУ и далее работа с компьютером происходила под управлением загруженной программы (как в игровых приставках).

Первые бытовые персональные компьютеры 70—80-х годов не имели операционных систем, но некоторые пользователи рассматривали содержащийся в ПЗУ язык программирования как самостоятельную операционную систему, хотя и упрощенную. Она позволяла принимать и понимать команды от клавиатуры и загружать посторонние программы.

Первые дисковые операционные системы

Серьезная необходимость в операционных системах возникла, когда к персональным компьютерам стали подключать дисководы. Дисковод отличается от магнитофона тем, что это устройство свободного доступа, а магнитофон — устройство последовательного доступа. Для загрузки программы с ленты надо было перемотать кассету, после чего компьютер загружал первую встретившуюся программу.

На диске есть свобода выбора. На музыкальном диске можно включить воспроизведение любой песни. С магнитного диска можно загрузить любую программу. Поэтому команды загрузки стали очень сложными. Надо было указывать номер дорожки и номер сектора, в котором находится то, что надо загрузить. Например, для загрузки игры Посадка на Луну приходилось давать команду типа: `LOAD *d* 29:37, 31:14`

Помнить, в каких секторах что хранится, было мучительно трудно. И выход был найден. Была написана программа, которая переводит названия программ и файлов в номера дорожек и секторов. Человек мог загружать то, что ему нужно, пользуясь только названиями. Эта программа и стала дисковой операционной системой.

Дисковой операционной системе поручили и другие задачи. Она могла не только загружать файлы с диска в компьютер, но и записывать файлы на диск, следить за тем, чтобы два разных файла не попадали в один сектор, при необходимости удалять файлы, копировать их с диска на диск. В общем, она избавила человека от необходимости хранить множество записей на отдельных бумажках, упростила работу с дисководом и значительно уменьшила количество ошибок.

Неграфические операционные системы

В дальнейшем операционные системы развивались параллельно с аппаратным обеспечением. Появлялись новые дисководы гибких дисков, менялись и операционные системы. С появлением жестких дисков открылась возможность хранить на них не десятки, а сотни и тысячи файлов. В именах файлов стало так же легко запутаться, как в номерах дорожек и секторов. Тогда дисковые операционные системы стали сложнее. В них ввели средства для разбиения дисков на каталоги и средства для обслуживания каталогов (перенос и копирование файлов между

каталогами, сортировка файлов и прочее). Так на дисках появилась файловая структура, а операционная система взяла на себя ее создание и обслуживание. Когда же жесткие диски приобрели еще большие размеры, операционная система «научилась» делить их на несколько логических дисков.

Вместе с развитием жестких и гибких дисков происходило увеличение оперативной памяти компьютера, менялись также процессоры. Каждая новая операционная система все лучше использовала оперативную память и могла работать со все более мощными процессорами. Для компьютеров IBM PC основной операционной системой с 1981 г. по 1995 г. была так называемая система MS-DOS. За эти годы она прошла развитие от версии MS-DOS 1.0 до MS-DOS 6.22.

Программы-оболочки

Операционная система MS-DOS позволила успешно работать с персональными компьютерами на протяжении почти 15 лет. Тем не менее, эту работу нельзя назвать удобной. Во-первых, MS-DOS — неграфическая операционная система, которая использует интерфейс командной строки. Это значит, что все команды надо набирать по буквам в специальной строке. Требовалось хорошо знать эти команды, помнить, как они записываются. Изучение операционной системы стало самостоятельной задачей, достаточно сложной для простого пользователя.

Когда-то MS-DOS выступила «посредником» между человеком и компьютером и помогла превратить сложные команды обращения к дискам в более простые и понятные, но по мере развития сама «обросла» избытком команд и стала сдерживать работу с компьютером. Так возникла необходимость в новом посреднике — тогда появились так называемые программы-оболочки. Оболочка — это программа, которая запускается под управлением операционной системы и помогает человеку работать с этой операционной системой. Одна из самых известных и распространенных во всем мире программ-оболочек называется Norton Commander. Ее разработал известнейший американский программист Питер Нортона, получивший всемирное признание за то, что упростил работу с компьютером для миллионов людей. Программа-оболочка наглядно показывает на экране всю файловую структуру компьютера: диски, каталоги и файлы. С такой программой не надо набирать сложные команды MS-DOS в командной строке. Файлы можно разыскивать, копировать, перемещать, удалять, сортировать, изменять (редактировать, править) запускать, пользуясь всего лишь несколькими клавишами. Просто, понятно и удобно. Сегодня Norton Commander все еще используют на многих компьютерах, особенно на тех, которые работают в системе MS-DOS. Правда, она все-таки устарела. Сейчас для работы с принято использовать более современные средства

Графические оболочки

Несмотря на то, что появление программ-оболочек заметно упростило работу с компьютером и его операционной системой, оболочки все-таки долгое время оставались неграфическими.

Одна из особенностей компьютеров IBM PC состоит в том, что в них текстовый и графический режим работы с экраном существуют отдельно. Компьютер переключается либо в тот режим, либо в другой. Нельзя, например, сделать так, чтобы часть экрана была в текстовом режиме, а часть — в графическом. Эти режимы несовместимы.

Работа с текстовым экраном долгое время была вполне приемлема для служебных целей. На многих предприятиях и в организациях не было необходимости в работе с графикой, а если такая потребность возникала, для этого было принято использовать компьютеры Macintosh. Однако когда встал вопрос об использовании IBM PC в качестве домашнего компьютера, возникла острая необходимость в графической операционной системе, которая наглядно выводит информацию на экран и которой можно управлять с помощью мыши.

Работы над графической операционной системой для IBM PC в компании Microsoft начались еще в 1981 г., но впервые такая система вышла в свет только в 1995 г. под названием Microsoft Windows 95. До появления Microsoft Windows 95 компьютеры IBM PC работали с неграфической системой MS-DOS, но для нее были сделаны несколько графических оболочек Windows 1.0, Windows 2.0, Windows 3.0, Windows 3.1, , Windows 3.11.

Оболочки Windows запускались под управлением MS-DOS, то есть не были самостоятельными операционными системами. Но поскольку с появлением Windows открылись некоторые новые возможности, все-таки принято называть Windows не оболочкой, а средой. Вот некоторые особенности Windows, отличающие эту среду от прочих оболочек:

- Многозадачность.
- Единый программный интерфейс.
- Единый интерфейс пользователя.
- Графический интерфейс пользователя.
- Единый аппаратно-программный интерфейс.

Графические операционные системы

Выпущенная в сентябре 1995 г. система Windows 95 стала первой графической операционной системой для компьютеров IBM PC.

Все следующие версии операционных систем Windows (98, NT, ME, 2000, XP) являются графическими.

Производители аппаратного обеспечения изготавливают узлы и приборы так, чтобы они были совместимы с Windows . Мы можем достаточно смело приобретать новые устройства и устанавливать их в компьютер, рассчитывая на то, что все прочие устройства и программы будут работать нормально. Система Windows ввела новый стандарт самоустанавливающихся устройств (plug-and-play). Подключение таких устройств происходит автоматически. Операционная система сама «узнает», что установлено в компьютере, и настраивается на работу с новым оборудованием.

На сегодняшний день на рынке программного обеспечения для IBM PC-совместимых компьютеров сосуществуют несколько семейств операционных систем, но операционные системы Windows являются наиболее распространенными среды пользователей.

Операционные системы компании Microsoft:

- DOS (Disk Operating System). Интерфейс – командная строка. Все команды приходилось набирать вручную, в командной строке ОС.
- Windows 3.1 и 3.11. – первый графический интерфейс. Хотя многие не считали эту систему операционной, а лишь системой, расширяющей возможности DOS. У всех дальнейших операционных систем интерфейс графический.
- Windows-95.
- Семейство Windows-98 / NT / ME / 2000 / XP.

Элементы интерфейса Windows:

- Рабочий стол. Название «Рабочий стол» подобрано удачно. На нем, как и на обычном рабочем столе расположены различные программы и инструменты, представленные в виде значков, или иконки.
- Значки.

Значками в Windows обозначаются программы, документы. Запуск производится двойным щелчком кнопки мыши по значку. Программа может быть расположена непосредственно на Рабочем столе, а может быть скрыта глубоко на диске, но и в этом случае представлена на Рабочем столе своим образом – ярлыком.

- Ярлыки.

Ярлык программы – это не сама программа, а только ее образ, указание на то место на диске, где она находится. Двойной щелчок по ярлыку также вызывает запуск программы. Ярлыки от значков отличаются наличием небольшой стрелочки внизу слева.

- Панель задач. Располагается в нижней части экрана. На ней находятся: кнопка Пуск, кнопки открытых окон, индикаторы и часы.
- Окно.

Окно – один из главных элементов интерфейса Windows

9. Файловая система. Папки и файлы. Имя, тип, путь доступа к файлу.

Файл.

Все программы и данные хранятся в долговременной (внешней) памяти компьютера в виде файлов.

Файл — это определенное количество информации (программа или данные), имеющее имя и хранящееся в долговременной (внешней) памяти.

Имя файла состоит из двух частей, разделенных точкой: собственно имя файла и расширение, определяющее его тип (программа, данные и т. д.). Собственно имя файлу дает пользователь, а тип файла обычно задается программой автоматически при его создании.

Тип файла	Расширение
Исполняемые программы	exe, com
Текстовые файлы	txt, rtf,
Графические файлы	bmp, gif, jpg, png, pds
Web-страницы	htm, html
Звуковые файлы	wav, mp3, midi, kar, ogg
Видеофайлы	avi, mpeg
Код (текст) программы на языках программирования	bas, pas, cpp

В различных операционных системах существуют различные форматы имен файлов. В операционной системе MS-DOS собственно имя файла должно содержать не более восьми букв латинского алфавита и цифр, а расширение состоит из трех латинских букв, например: *proba.txt*

В операционной системе Windows имя файла может иметь до 255 символов, причем допускается использование русского алфавита, например:

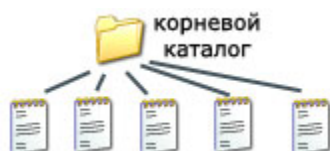
Единицы измерения информации.doc

Файловая система.

На каждом носителе информации (гибком, жестком или лазерном диске) может храниться большое количество файлов. Порядок хранения файлов на диске определяется установленной файловой системой.

Файловая система - это система хранения файлов и организации каталогов.

Для дисков с небольшим количеством файлов (до нескольких десятков) удобно применять одноуровневую файловую систему, когда каталог (оглавление диска) представляет собой линейную последовательность имен файлов.



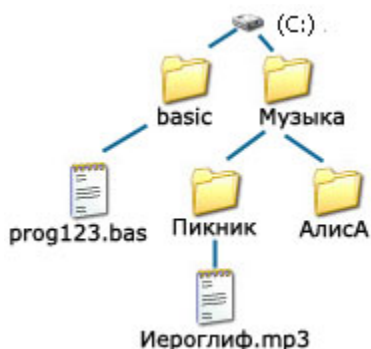
Если на диске хранятся сотни и тысячи файлов, то для удобства поиска файлы организуются в много уровневую иерархическую файловую систему, которая имеет «древовидную» структуру.



Начальный, корневой, каталог содержит вложенные каталоги 1-го уровня, в свою очередь, в каждом из них бывают вложенные каталоги 2-го уровня и т. д. Необходимо отметить, что в каталогах всех уровней могут храниться и файлы.

Путь к файлу.

Для того чтобы найти файл в иерархической файловой структуре необходимо указать путь к файлу. В путь к файлу входят записываемые через разделитель "\" логическое имя диска и последовательность имен вложенных друг в друга каталогов, в последнем из которых находится данный нужный файл.



Например, путь к файлам на рисунке можно записать так:

C:\basic\

C:\Музыка\Пикник\

Полное имя файла.

Путь к файлу вместе с именем файла называют полным именем файла.

Пример полного имени файлов:

C:\basic\prog123.bas

C:\Музыка\Пикник\Иероглиф.mp3

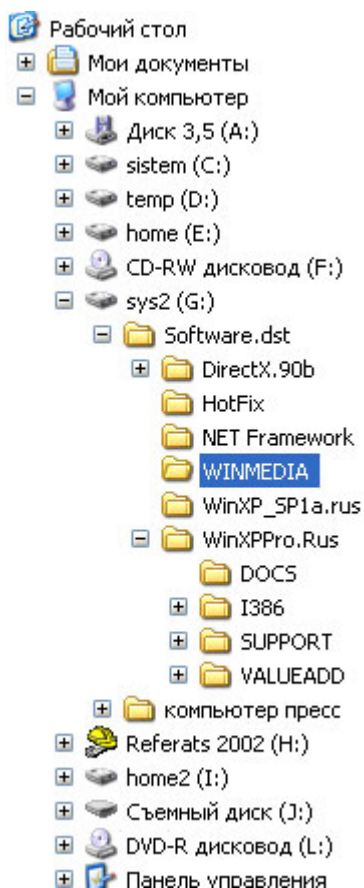
Операции над файлами.

В процессе работы на компьютере над файлами чаще всего производятся следующие операции: копирование (копия файла помещается в другой каталог); перемещение (сам файл перемещается в другой каталог); удаление (запись о файле удаляется из каталога); переименование (изменяется имя файла).

Графическое представление файловой системы.

Иерархическая файловая система MS-DOS, содержащая каталоги и файлы, представлена в операционной системе Windows с помощью графического интерфейса в форме иерархической системы папок и документов. Папка в Windows является аналогом каталога MS-DOS. Однако иерархические структуры этих систем несколько различаются. В иерархической файловой системе MS-DOS вершиной иерархии объектов является корневой каталог диска, который можно сравнить со стволом дерева — на нем растут ветки (подкаталоги), а на ветках располагаются листья (файлы).

В Windows на вершине иерархии папок находится папка Рабочий стол. (Следующий уровень представлен папками Мой компьютер, Корзина и Сетевое окружение (если компьютер подключен к локальной сети)).



10. Представление данных в памяти персонального компьютера (числа, символы, графика, звук).

Форма и язык представления информации

Воспринимая информацию с помощью органов чувств, человек стремится зафиксировать ее так, чтобы она стала понятной и другим, представляя ее в той или иной форме.

Музыкальную тему композитор может наиграть на пианино, а затем записать с помощью нот. Образы, навеянные все той же мелодией, поэт может воплотить в виде стихотворения, хореограф выразить танцем, а художник — в картине.

Человек выражает свои мысли в виде предложений, составленных из слов. Слова, в свою очередь, состоят из букв. Это — алфавитное представление информации.

Форма представления одной и той же информации может быть различной. Это зависит от цели, которую вы перед собой поставили. С подобными операциями вы сталкиваетесь на уроках математики и физики, когда представляете решение в разной форме. Например, решение задачи: «Найти значение математического выражения ...» можно представить в табличной или графической форме. Для этого вы пользуетесь визуальными средствами представления информации: числами, таблицей, рисунком.

Таким образом, информацию можно представить в различной форме:

- знаковой письменной, состоящей из различных знаков, среди которых принято выделять
 - символную в виде текста, чисел, специальных символов (например, текст учебника);
 - графическую (например, географическая карта);
 - табличную (например, таблица записи хода физического эксперимента);
- в виде жестов или сигналов (например, сигналы регулировщика дорожного движения);

- устной словесной (например, разговор).

Форма представления информации очень важна при ее передаче: если человек плохо слышит, то передавать ему информацию в звуковой форме нельзя; если у собаки слабо развито обоняние, то она не может работать в розыскной службе. В разные времена люди передавали информацию в различной форме с помощью: речи, дыма, барабанного боя, звона колоколов, письма, телеграфа, радио, телефона, факса.

Независимо от формы представления и способа передачи информации, она всегда передается с помощью какого-либо языка.

На уроках математики вы используете специальный язык, в основе которого — цифры, знаки арифметических действий и отношений. Они составляют алфавит языка математики.

На уроках физики при рассмотрении какого-либо физического явления вы используете характерные для данного языка специальные символы, из которых составляете формулы. Формула — это слово на языке физики.

На уроках химии вы также используете определенные символы, знаки, объединяя их в «слова» данного языка.

Существует язык глухонемых, где символы языка — определенные знаки, выражаемые мимикой лица и движениями рук.

Основу любого языка составляет алфавит — набор однозначно определенных знаков (символов), из которых формируется сообщение.

Языки делятся на естественные (разговорные) и формальные. Алфавит естественных языков зависит от национальных традиций. Формальные языки встречаются в специальных областях человеческой деятельности (математике, физике, химии и т. д.). В мире насчитывается около 10000 разных языков, диалектов, наречий. Многие разговорные языки произошли от одного и того же языка. Например, от латинского языка образовались французский, испанский, итальянский и другие языки.

Кодирование информации

С появлением языка, а затем и знаковых систем расширились возможности общения между людьми. Это позволило хранить идеи, полученные знания и любые данные, передавать их различными способами на расстояние и в другие времена — не только своим современникам, но и будущим поколениям. До наших дней дошли творения предков, которые с помощью различных символов увековечили себя и свои деяния в памятниках и надписях. Наскальные рисунки (петроглифы) до сих пор служат загадкой для ученых. Возможно, таким способом древние люди хотели вступить в контакт с нами, будущими жителями планеты и сообщить о событиях их жизни.

Каждый народ имеет свой язык, состоящий из набора символов (букв): русский, английский, японский и многие другие. Вы уже познакомились с языком математики, физики, химии.

Представление информации с помощью какого-либо языка часто называют кодированием.

Код — набор символов (условных обозначений) для представления информации. Кодирование — процесс представления информации в виде кода.

Водитель передает сигнал с помощью гудка или миганием фар. Кодом является наличие или отсутствие гудка, а в случае световой сигнализации — мигание фар или его отсутствие.

Вы встречаетесь с кодированием информации при переходе дороги по сигналам светофора. Код определяют цвета светофора — красный, желтый, зеленый.

В основу естественного языка, на котором общаются люди, тоже положен код. Только в этом случае он называется алфавитом. При разговоре этот код передается звуками, при письме — буквами. Одну и ту же информацию можно представить с помощью различных кодов. Например, запись разговора можно зафиксировать посредством русских букв или специальных стенографических значков.

По мере развития техники появлялись разные способы кодирования информации. Во второй половине XIX века американский изобретатель Сэмюэль Морзе изобрел удивительный код, который служит человечеству до сих пор. Информация кодируется тремя «буквами»: длинный сигнал (тире), короткий сигнал (точка) и отсутствие сигнала (пауза) для разделения букв. Таким образом, кодирование сводится к использованию набора символов, расположенных в строго определенном порядке.

Люди всегда искали способы быстрого обмена сообщениями. Для этого посылали гонцов, использовали почтовых голубей. У народов существовали различные способы оповещения о надвигающейся опасности: барабанный бой, дым костров, флаги и т. д. Однако использование такого представления информации требует предварительной договоренности о понимании принимаемого сообщения.

Знаменитый немецкий ученый Готфрид Вильгельм Лейбниц предложил еще в XVII веке уникальную и простую систему представления чисел. «Вычисление с помощью двоек... является для науки основным и порождает новые открытия... при сведении чисел к простейшим началам, каковы 0 и 1, везде появляется чудесный порядок».

Сегодня такой способ представления информации с помощью языка, содержащего всего два символа алфавита — 0 и 1, широко используется в технических устройствах, в том числе и в компьютере. Эти два символа 0 и 1 принято называть двоичными цифрами или битами (от англ. bit — Binary Digit - двоичный знак).

Инженеров такой способ кодирования привлек простотой технической реализации — есть сигнал или нет сигнала. С помощью этих двух цифр можно закодировать любое сообщение.

Более крупной единицей измерения объема информации принято считать 1 байт, который состоит из 8 бит.

Принято также использовать и более крупные единицы измерения объема информации. Число $1024 (2^{10})$ является множителем при переходе к более высокой единице измерения.

Килобит	Кбит	Кбит = 1024 бит \approx 1000 бит
Мегабит	Мбит	1 Мбит = 1024 Кбит \approx 1 000 000 бит
Гигабит	Гбит	Гбит = 1024 Мбит \approx 1 000 000 000 бит
Килобайт	Кбайт (Кб)	1 Кбайт = 1024 байт \approx 1000 байт
Мегабайт	Мбайт (Мб)	1 Мбайт = 1024 Кбайт \approx 1 000 000 байт
Гигабайт	Гбайт (Гб)	1 Гбайт = 1024 Мбайт \approx 1 000 000 000 байт

Кодирование информации в компьютере

Вся информация, которую обрабатывает компьютер, должна быть представлена двоичным кодом с помощью двух цифр — 0 и 1. Эти два символа принято называть двоичными цифрами, или битами. С помощью двух цифр 1 и 0 можно закодировать любое сообщение. Это явилось причиной того, что в компьютере обязательно должно быть организовано два важных процесса:

- кодирование, которое обеспечивается устройствами ввода при преобразовании входной информации в форму, воспринимаемую компьютером, то есть в двоичный код;
- декодирование, которое обеспечивается устройствами вывода при преобразовании данных из двоичного кода в форму, понятную человеку.

С точки зрения технической реализации использование двоичной системы счисления для кодирования информации оказалось намного более простым, чем применение других способов. Действительно, удобно кодировать информацию в виде последовательности нулей и единиц, если представить эти значения как два возможных устойчивых состояния электронного элемента:

- 0 — отсутствие электрического сигнала или сигнал имеет низкий уровень;
- 1 — наличие сигнала или сигнал имеет высокий уровень.

Эти состояния легко различать. Недостаток двоичного кодирования — длинные коды. Но в технике легче иметь дело с большим числом простых элементов, чем с небольшим количеством сложных.

Вам и в быту ежедневно приходится сталкиваться с устройством, которое может находиться только в двух устойчивых состояниях: включено/выключено. Конечно же, это хорошо знакомый всем выключатель. А вот придумать выключатель, который мог бы устойчиво и быстро переключаться в любое из 10 состояний, оказалось невозможным. В результате после ряда неудачных попыток разработчики пришли к выводу о невозможности построения компьютера на основе десятичной системы счисления. И в основу представления чисел в компьютере была положена именно двоичная система счисления.

В настоящее время существуют разные способы двоичного кодирования и декодирования информации в компьютере. В первую очередь это зависит от вида информации, а именно, что должно кодироваться: текст, числа, графические изображения или звук. Кроме того, при кодировании чисел важную роль играет то, как они будут использоваться: в тексте, в расчетах или в процессе ввода-вывода. Накладываются также и особенности технической реализации.

Кодирование чисел

Система счисления — совокупность приемов и правил записи чисел с помощью определенного набора символов.

Для записи чисел могут использоваться не только цифры, но и буквы (например, запись римских цифр — XXI). Одно и то же число может быть по-разному представлено в различных системах счисления.

В зависимости от способа изображения чисел системы счисления делятся на позиционные и непозиционные.

В позиционной системе счисления количественное значение каждой цифры числа зависит от того, в каком месте (позиции или разряде) записана та или иная цифра этого числа. Например,

меняя позицию цифры 2 в десятичной системе счисления, можно записать разные по величине десятичные числа, например 2; 20; 2000; 0,02 и т. д.

В непозиционной системе счисления цифры не изменяют своего количественного значения при изменении их расположения (позиции) в числе. Примером непозиционной системы может служить римская система, в которой независимо от местоположения одинаковый символ имеет неизменное значение (например, символ X в числе XXV).

Количество различных символов, используемых для изображения числа в позиционной системе счисления, называется основанием системы счисления.

В компьютере наиболее подходящей и надежной оказалась двоичная система счисления, в которой для представления чисел используются последовательности цифр 0 и 1.

Кроме того, для работы с памятью компьютера оказалось удобным использовать представление информации с помощью еще двух систем счисления:

- восьмеричной (любое число представляется с помощью восьми цифр — 0, 1, 2... 7);
- шестнадцатеричной (используемые символы-цифры — 0, 1, 2... 9 и буквы — A, B, C, D, E, F, заменяющие числа 10, 11, 12, 13, 14, 15 соответственно).

Кодирование символьной информации

Нажатие алфавитно-цифровой клавиши на клавиатуре приводит к тому, что в компьютер посылается сигнал в виде двоичного числа, представляющего собой одно из значений кодовой таблицы. Кодовая таблица - это внутреннее представление символов в компьютере. Во всем мире в качестве стандарта принята таблица ASCII (American Standart Code for Informational Interchange - американский стандартный код информационного обмена).

Для хранения двоичного кода одного символа выделен 1 байт = 8 бит. Учитывая, что каждый бит принимает значение 1 или 0, количество возможных сочетаний единиц и нулей равно $2^8 = 256$.

Значит, с помощью 1 байта можно получить 256 разных двоичных кодовых комбинаций и отобразить с их помощью 256 различных символов. Эти коды и составляют таблицу ASCII.

Пример, при нажатии клавиши с буквой S в память компьютера записывается код 01010011. При выводе буквы S на экран компьютер выполняет декодирование — на основании этого двоичного кода строится изображение символа.

SUN (СОЛНЦЕ) - 01010011 010101101 01001110

Стандарт ASCII кодирует первые 128 символов от 0 до 127: цифры, буквы латинского алфавита, управляющие символы. Первые 32 символа являются управляющими и предназначены в основном для передачи команд управления. Их назначение может варьироваться в зависимости от программных и аппаратных средств. Вторая половина кодовой таблицы (от 128 до 255) американским стандартом не определена и предназначена для символов национальных алфавитов, псевдографических и некоторых математических символов. В разных странах могут использоваться различные варианты второй половины кодовой таблицы.

Обратите внимание! Цифры кодируются по стандарту ASCII записываются в двух случаях - при вводе-выводе и когда они встречаются в тексте. Если цифры участвуют в вычислениях, то осуществляется их преобразование в другой двоичный код.

Для сравнения рассмотрим число 45 для двух вариантов кодирования.

При использовании в тексте это число потребует для своего представления 2 байта, поскольку каждая цифра будет представлена своим кодом в соответствии с таблицей ASCII. В двоичной системе - 00110100 00110101.

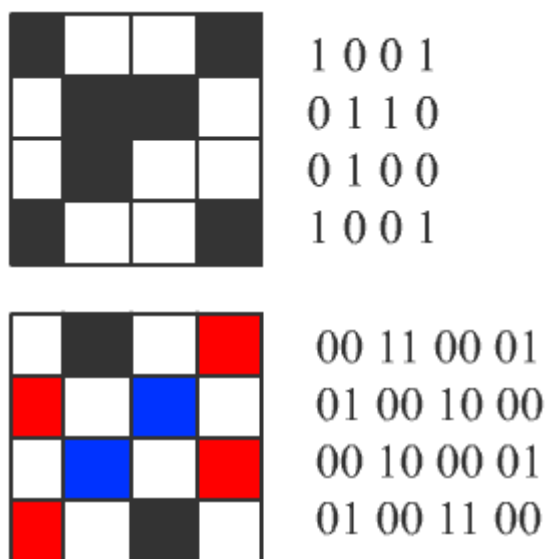
При использовании в вычислениях код этого числа будет получен по специальным правилам перевода и представлен в виде 8-разрядного двоичного числа 00101101, на что потребуется 1 байт.

Кодирование графической информации

Создавать и хранить графические объекты в компьютере можно двумя способами — как растровое или как векторное изображение. Для каждого типа изображения используется свой способ кодирования.

Растровое изображение представляет собой совокупность точек, используемых для его отображения на экране монитора. Объем растрового изображения определяется как произведение количества точек и информационного объема одной точки, который зависит от количества возможных цветов. Для черно-белого изображения информационный объем одной точки равен 1 биту, так как точка может быть либо черной, либо белой, что можно закодировать двумя цифрами — 0 или 1.

Для кодирования 8 цветов необходимо 3 бита; для 16 цветов — 4 бита; для 32 цветов — 8 битов (1 байт) и т.д.



Векторное изображение представляет собой совокупность графических примитивов. Каждый примитив состоит из элементарных отрезков кривых, параметры которых (координаты узловых точек, радиус кривизны и пр.) описываются математическими формулами. Для каждой линии указывается ее тип (сплошная, пунктирная, штрих-пунктирная), толщина и цвет, а замкнутые фигуры дополнительно характеризуются типом заливки. Кодирование векторных изображений выполняется различными способами в зависимости от прикладной среды. В частности, формулы, описывающие отрезки кривых, могут кодироваться как обычная буквенно-цифровая информация для дальнейшей обработки специальными программами.

Кодирование звуковой информации

Звук представляет собой звуковую волну с непрерывно меняющейся амплитудой и частотой. Чем больше амплитуда сигнала, тем он громче для человека, чем больше частота сигнала, тем выше тон. Для того чтобы компьютер мог обрабатывать звук, непрерывный звуковой сигнал должен быть превращен в последовательность электрических импульсов (двоичных нулей и единиц).

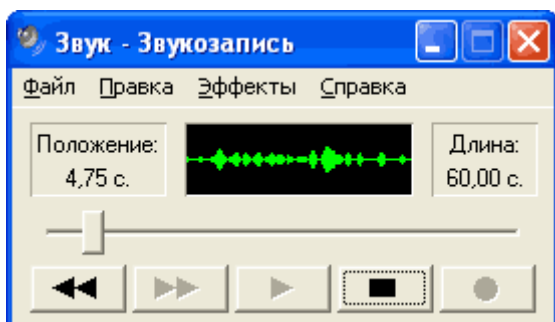
В процессе кодирования непрерывного звукового сигнала производится его временная дискретизация. Непрерывная звуковая волна разбивается на отдельные маленькие участки, причем для каждого такого участка устанавливается определенная величина амплитуды. Таким образом, непрерывная зависимость амплитуды сигнала от времени заменяется на дискретную последовательность уровней громкости.

Современные звуковые карты обеспечивают 16-битную глубину кодирования звука. В таком случае количество уровней сигнала будет равно 65536.

При двоичном кодировании непрерывного звукового сигнала он заменяется последовательностью дискретных уровней сигнала. Качество кодирования зависит от количества измерений уровня сигнала в единицу времени, т.е. от частоты дискретизации. Чем больше количество измерений производится за 1 секунду (чем больше частота дискретизации), тем точнее процедура двоичного кодирования.

Количество измерений в секунду может лежать в диапазоне от 8000 до 48000, т.е. частота дискретизации аналогового звукового сигнала может принимать значения от 8 до 48 кГц - качество звучания аудио-CD. Следует также учитывать, что возможны как моно-, так и стерео-режимы.

Стандартная программа Windows Звукозапись играет роль цифрового магнитофона и позволяет записывать звук, т.е. дискретизировать звуковые сигналы, и сохранять их в звуковых файлах в формате wav. Также эта программа позволяет производить простейшее редактирование звуковых файлов.



11. Понятие модели. Материальные и информационные модели. Формализация как замена реального объекта его информационной моделью.

Моделирование.

Человечество в своей деятельности (научной, образовательной, технологической, художественной) постоянно создает и использует модели окружающего мира. Строгие правила построения моделей сформулировать невозможно, однако человечество накопило богатый опыт моделирования различных объектов и процессов.

Модели позволяют представить в наглядной форме объекты и процессы, недоступные для непосредственного восприятия (очень большие или очень маленькие объекты, очень быстрые или очень медленные процессы и др.). Наглядные модели часто используются в процессе обучения. В курсе географии первые представления о нашей планете Земля мы получаем, изучая ее модель — глобус, в курсе физики изучаем работу двигателя внутреннего сгорания по его модели, в химии при изучении строения вещества используем модели молекул и кристаллических решеток, в биологии изучаем строение человека по анатомическим муляжам и др.

Модели играют чрезвычайно важную роль в проектировании и создании различных технических устройств, машин и механизмов, зданий, электрических цепей и т. д. Без предварительного создания чертежа невозможно изготовить даже простую деталь, не говоря уже о сложном механизме.

В процессе проектирования зданий и сооружений кроме чертежей часто изготавливают макеты. В процессе разработки летательных аппаратов поведение их моделей в воздушных потоках исследуют в аэродинамической трубе. Разработка электрической схемы обязательно предшествует созданию электрических цепей и так далее.

Развитие науки невозможно без создания теоретических моделей (теорий, законов, гипотез и пр.), отражающих строение, свойства и поведение реальных объектов. Создание новых теоретических моделей иногда коренным образом меняет представление человечества об окружающем мире (гелиоцентрическая система мира Коперника, модель атома Резерфорда-Бора, модель расширяющейся Вселенной, модель генома человека и пр.). Адекватность теоретических моделей законам реального мира проверяется с помощью опытов и экспериментов.

Все художественное творчество фактически является процессом создания моделей. Например, такой литературный жанр, как басня, переносит реальные отношения между людьми на отношения между животными и фактически создает модели человеческих отношений. Более того, практически любое литературное произведение может рассматриваться как модель реальной человеческой жизни. Моделями, в художественной форме отражающими реальную действительность, являются также живописные полотна, скульптуры, театральные постановки и пр.

Моделирование — это метод познания, состоящий в создании и исследовании моделей.

Модель.

Каждый объект имеет большое количество различных свойств. В процессе построения модели выделяются главные, наиболее существенные для проводимого исследования свойства. В процессе исследования аэродинамических качеств модели самолета в аэродинамической трубе важно, чтобы модель имела геометрическое подобие оригинала, но не важен, например, ее цвет. При построении электрических схем — моделей электрических цепей — необходимо учитывать порядок подключения элементов цепи друг к другу, но не важно их геометрическое расположение друг относительно друга и так далее.

Разные науки исследуют объекты и процессы под разными углами зрения и строят различные типы моделей.

В физике изучаются процессы взаимодействия и изменения объектов, в химии — их химический состав, в биологии — строение и поведение живых организмов и так далее.

Возьмем в качестве примера человека: в разных науках он исследуется в рамках различных моделей. В рамках механики его можно рассматривать как материальную точку, в химии — как объект, состоящий из различных химических веществ, в биологии — как систему, стремящуюся к самосохранению, и так далее.

Модель — это некий новый объект, который отражает существенные особенности изучаемого объекта, явления или процесса. С другой стороны, разные объекты могут описываться одной моделью. Так, в механике различные материальные тела (от планеты до песчинки) могут рассматриваться как материальные точки. Один и тот же объект может иметь множество моделей, а разные объекты могут описываться одной моделью.

Никакая модель не может заменить сам объект. Но при решении конкретной задачи, когда нас интересуют определенные свойства изучаемого объекта, модель оказывается полезным, а подчас и единственным инструментом исследования.

Модели материальные и модели информационные.

Все модели можно разбить на два больших класса: модели предметные (материальные) и модели информационные. Предметные модели воспроизводят геометрические, физические и другие свойства объектов в материальной форме (глобус, анатомические муляжи, модели кристаллических решеток, макеты зданий и сооружений и др.).

Информационные модели представляют объекты и процессы в образной или знаковой форме.

Образные модели (рисунки, фотографии и др.) представляют собой зрительные образы объектов, зафиксированные на каком-либо носителе информации (бумаге, фото- и киноплёнке и др.). Широко используются образные информационные модели в образовании (учебные плакаты по различным предметам) и науках, где требуется классификация объектов по их внешним признакам (в ботанике, биологии, палеонтологии и др.).

Знаковые информационные модели строятся с использованием различных языков (знаковых систем). Знаковая информационная модель может быть представлена в форме текста (например, программы на языке программирования), формулы (например, второго закона Ньютона $F=ma$), таблицы (например, периодической таблицы элементов Д. И. Менделеева) и так далее.

Иногда при построении знаковых информационных моделей используются одновременно несколько различных языков. Примерами таких моделей могут служить географические карты, графики, диаграммы и пр. Во всех этих моделях используются одновременно как язык графических элементов, так и символичный язык.

Моделирование и формализация

На протяжении своей истории человечество использовало различные способы и инструменты для создания информационных моделей. Эти способы постоянно совершенствовались. Так, первые информационные модели создавались в форме наскальных рисунков, в настоящее же время информационные модели обычно строятся и исследуются с использованием современных компьютерных технологий.

Формализация.

Естественные языки используются для создания описательных информационных моделей. В истории науки известны многочисленные описательные информационные модели; например, гелиоцентрическая модель мира, которую предложил Коперник, формулировалась следующим образом:

- Земля вращается вокруг своей оси и вокруг Солнца;
- орбиты всех планет проходят вокруг Солнца.

С помощью формальных языков строятся формальные информационные модели (математические, логические и др.). Одним из наиболее широко используемых формальных языков является математика. Модели, построенные с использованием математических понятий и формул, называются математическими моделями. Язык математики является совокупностью формальных языков.

Язык алгебры позволяет формализовать функциональные зависимости между величинами. Так, Ньютон формализовал гелиоцентрическую систему мира, открыв законы механики и закон всемирного тяготения и записав их в виде алгебраических функциональных зависимостей. В школьном курсе физики рассматривается много разнообразных функциональных зависимостей, выраженных на языке алгебры, которые представляют собой математические модели изучаемых явлений или процессов.

Язык алгебры логики (алгебры высказываний) позволяет строить формальные логические модели. С помощью алгебры высказываний можно формализовать (записать в виде логических выражений) простые и сложные высказывания, выраженные на естественном языке. Построение логических моделей позволяет решать логические задачи, строить логические модели устройств компьютера (сумматора, триггера) и так далее.

В процессе познания окружающего мира человечество постоянно использует моделирование и формализацию. При изучении нового объекта сначала обычно строится его описательная информационная модель на естественном языке, затем она формализуется, то есть выражается с использованием формальных языков (математики, логики и др.).

Визуализация формальных моделей.

В процессе исследования формальных моделей часто производится их визуализация. Для визуализации алгоритмов используются блок-схемы: пространственных соотношений между объектами — чертежи, моделей электрических цепей — электрические схемы, логических моделей устройств — логические схемы и так далее.

Так при визуализации формальных физических моделей с помощью анимации может отображаться динамика процесса, производится построение графиков изменения физических величин и так далее. Визуальные модели обычно являются интерактивными, то есть исследователь может менять начальные условия и параметры протекания процессов и наблюдать изменения в поведении модели.

12. Модели объектов и процессов (графические, вербальные, табличные, математические и др.).

Все модели можно разбить на два больших класса: модели предметные (материальные) и модели информационные. Предметные модели воспроизводят геометрические, физические и другие свойства объектов в материальной форме (глобус, анатомические муляжи, модели кристаллических решеток, макеты зданий и сооружений и др.).

Типы информационных моделей

Информационные модели представляют объекты и процессы в образной или знаковой форме. Образные модели (рисунки, фотографии и др.) представляют собой зрительные образы объектов, зафиксированные на каком-либо носителе информации (бумаге, фото- и киноплёнке и др.). Широко используются образные информационные модели в образовании (учебные плакаты по различным предметам) и науках, где требуется классификация объектов по их внешним признакам (в ботанике, биологии, палеонтологии и др.).

Знаковые информационные модели строятся с использованием различных языков (знаковых систем). Знаковая информационная модель может быть представлена в форме текста (например, программы на языке программирования), формулы (например, второго закона Ньютона $F=ma$), таблицы (например, периодической таблицы элементов Д. И. Менделеева) и так далее.

Иногда при построении знаковых информационных моделей используются одновременно несколько различных языков. Примерами таких моделей могут служить географические карты, графики, диаграммы и пр. Во всех этих моделях используются одновременно как язык графических элементов, так и символичный язык.

Табличные информационные модели

Одним из наиболее часто используемых типов информационных моделей является прямоугольная таблица, которая состоит из столбцов и строк. Такой тип моделей применяется для описания ряда объектов, обладающих одинаковыми наборами свойств. С помощью таблиц могут быть построены как статические, так и динамические информационные модели в различных предметных областях. Широко известно табличное представление математических функций, статистических данных, расписаний поездов и самолетов, уроков и так далее.

Представление объектов и их свойств в форме таблицы часто используется в научных исследованиях. Так, на развитие химии и физики решающее влияние оказало создание Д. И. Менделеевым в конце XIX века периодической системы элементов, которая представляет собой табличную информационную модель. В этой модели химические элементы располагаются в ячейках таблицы по возрастанию атомных весов, а в столбцах — по количеству валентных электронов, причем по положению в таблице можно определить некоторые физические и химические свойства элементов.

Табличные информационные модели проще всего строить и исследовать на компьютере с помощью электронных таблиц и систем управления базами данных. Визуализируем полученную табличную модель путем построения диаграммы в электронных таблицах.

Иерархические и сетевые модели

Множество окружающих нас объектов обладает одинаковыми свойствами, которые отличают их от других групп объектов. Группа объектов, обладающих одинаковыми общими свойствами, называется классом объектов. Внутри класса объектов могут быть выделены подклассы, объекты которых обладают какими-то особыми свойствами. В свою очередь подклассы могут делиться на более мелкие группы и т.д.

В процессе классификации объектов часто строят информационные модели, которые имеют иерархическую структуру.

В иерархической информационной модели объекты распределены по уровням. Каждый элемент более высокого уровня может состоять из элементов нижнего уровня, а элемент нижнего уровня может входить в состав только одного элемента более высокого уровня.

Иерархические информационные модели для наглядного представления удобно изображать в форме графа.



Сетевые информационные модели применяются для отражения систем со сложной структурой, в которых связи между элементами имеют произвольный характер.

Пример, структура глобальной сети Интернет, в которой различные региональные части связаны между собой высокоскоростными линиями связи. Причем, одни части имеют прямые связи со всеми региональными частями Интернета, а другие могут обмениваться информацией между собой только через американскую часть.

Классификация моделей по временному фактору

Статическая модель — это как бы одномоментный срез информации по объекту. Например, обследование учащихся в стоматологической поликлинике дает картину состояния их ротовой полости на данный момент времени: число молочных и постоянных зубов, пломб, дефектов и т.п.

Динамическая модель позволяет увидеть изменения объекта во времени. В примере с поликлиникой карточку школьника, отражающую изменения, происходящие с его зубами за многие годы, можно считать динамической моделью.

При строительстве дома рассчитывают прочность и устойчивость к постоянной нагрузке его фундамента, стен, балок — это статическая модель здания. Но еще надо обеспечить противодействие ветрам, движению грунтовых вод, сейсмическим колебаниям и другим изменяющимся во времени факторам. Это можно решить с помощью динамических моделей

Информационные модели отражают различные типы систем объектов, в которых реализуются различные структуры взаимодействия и взаимосвязи между элементами системы. Для отражения систем с различными структурами используются различные типы информационных моделей: табличные, иерархические и сетевые.

Знаковые и вербальные информационные модели

К информационным моделям можно отнести вербальные (от лат. «verbalize» — устный) модели, полученные в результате раздумий, умозаключений. Они могут так и остаться мысленными или быть выражены словесно. Примером такой модели может стать наше поведение при переходе улицы. Человек анализирует ситуацию на дороге (что показывает светофор, как далеко находятся машины, с какой скоростью они движутся и т. п.) и вырабатывает свою модель поведения. Если ситуация смоделирована правильно, то переход будет безопасным, если нет, то может произойти авария. К таким моделям можно отнести и идею, возникшую у изобретателя, и музыкальную тему, промелькнувшую в голове композитора, и рифму, прозвучавшую пока еще в сознании поэта.

Вербальная модель – информационная модель в мысленной или разговорной форме.
Знаковая модель – информационная модель, выраженная специальными знаками, т. е. средствами любого формального языка.
Знаковые модели окружают нас повсюду. Это рисунки, тексты, графики и схемы... Вербальные и знаковые модели, как правило, взаимосвязаны. Мысленный образ, родившийся в мозгу человека, может быть облечен в знаковую форму. И наоборот, знаковая модель помогает сформировать в сознании верный мысленный образ.

Согласно легенде, яблоко, упавшее на голову Ньютону, вызвало в его сознании мысль о земном притяжении. И только впоследствии эта мысль оформилась в закон, т. е. обрела знаковую форму.

Человек прочитал текст, объясняющий некоторое физическое явление, и у него сформировался мысленный образ. В дальнейшем такой образ поможет распознать реальное явление.

13. Понятие алгоритма. Свойства алгоритма. Исполнители алгоритмов (назначение, среда, режим работы, система команд). Компьютер как формальный исполнитель алгоритмов (программ).

Появление алгоритмов связывают с зарождением математики. Более 1000 лет назад (в 825 году) ученый из города Хорезма Абдулла (или Абу Джафар) Мухаммед бен Муса аль-Хорезми создал книгу по математике, в которой описал способы выполнения арифметических действий над многозначными числами. Само слово алгоритм возникло в Европе после перевода на латынь книги этого математика.

Алгоритм – описание последовательности действий (план), строгое исполнение которых приводит к решению поставленной задачи за конечное число шагов.

Вы постоянно сталкиваетесь с этим понятием в различных сферах деятельности человека (кулинарные книги, инструкции по использованию различных приборов, правила решения математических задач...). Обычно мы выполняем привычные действия не задумываясь, механически. Например, вы хорошо знаете, как открывать ключом дверь. Однако, чтобы научить этому малыша, придется четко разъяснить и сами эти действия и порядок их выполнения:

1. Достать ключ из кармана.

2. Вставить ключ в замочную скважину.

3. Повернуть ключ два раза против часовой стрелки.

4. Вынуть ключ.

Если вы внимательно оглянитесь вокруг, то обнаружите множество алгоритмов которые мы с вами постоянно выполняем. Мир алгоритмов очень разнообразен. Несмотря на это, удастся выделить общие свойства, которыми обладает любой алгоритм.

Свойства алгоритмов:

1. Дискретность (алгоритм должен состоять из конкретных действий, следующих в определенном порядке);

2. Детерминированность (любое действие должно быть строго и недвусмысленно определено в каждом случае);

3. Конечность (каждое действие и алгоритм в целом должны иметь возможность завершения);

4. Массовость (один и тот же алгоритм можно использовать с разными исходными данными);

5. Результативность (отсутствие ошибок, алгоритм должен приводить к правильному результату для всех допустимых входных значениях).

Виды

алгоритмов:




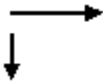
1. Линейный алгоритм (описание действий, которые выполняются однократно в заданном порядке);

2. Циклический алгоритм (описание действий, которые должны повторяться указанное число раз или пока не выполнено задание);

3. Разветвляющий алгоритм (алгоритм, в котором в зависимости от условия выполняется либо одна, либо другая последовательность действий)

4. Вспомогательный алгоритм (алгоритм, который можно использовать в других алгоритмах, указав только его имя).

Для более наглядного представления алгоритма широко используется *графическая форма - блок-схема*, которая составляется из стандартных графических объектов.

Вид стандартного графического объекта	Назначение
	Начало алгоритма
	Конец алгоритма
	Выполняемое действие записывается внутри прямоугольника
	Условие выполнения действий записывается внутри ромба
	Счетчик кол-во повторов
	Последовательность выполнения действий.

Стадии создания алгоритма:

1. Алгоритм должен быть представлен в форме, понятной человеку, который его разрабатывает.

2. Алгоритм должен быть представлен в форме, понятной тому объекту (в том числе и человеку), который будет выполнять описанные в алгоритме действия.

Объект, который будет выполнять алгоритм, обычно называют исполнителем.

Исполнитель - объект, который выполняет алгоритм.

Идеальными исполнителями являются машины, роботы, компьютеры...

Исполнитель способен выполнить только ограниченное количество команд. Поэтому алгоритм разрабатывается и детализируется так, чтобы в нем присутствовали только те команды и конструкции, которые может выполнить исполнитель.

Исполнитель, как и любой объект, находится в определенной среде и может выполнять только допустимые в нем действия. Если исполнитель встретит в алгоритме неизвестную ему команду, то выполнение алгоритма прекратится.

Компьютер – автоматический исполнитель алгоритмов.

Алгоритм, записанный на «понятном» компьютеру языке программирования, называется программой.

Программирование - процесс составления программы для компьютера. Для первых ЭВМ программы записывались в виде последовательности элементарных операций. Это была очень трудоемкая и неэффективная работа. Поэтому в последствии были разработанные специальные языки программирования. В настоящее время существует множество искусственных языков для составления программ. Однако, так и не удалось создать идеальный язык, который бы устроил бы всех.

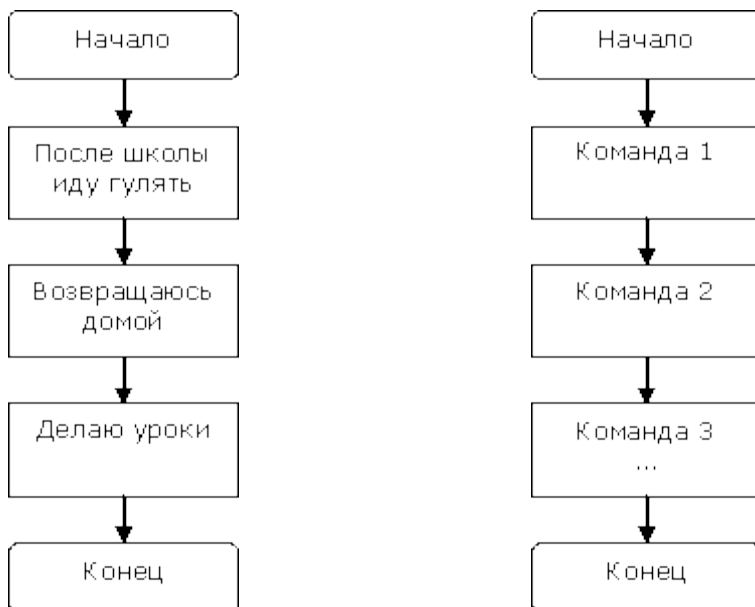
14. Линейная алгоритмическая конструкция. Команда присваивания. Примеры.

Для представления алгоритма в виде, понятном компьютеру, служат языки программирования. Сначала разрабатывается алгоритм действий, а потом он записывается на одном из таких языков. В итоге получается текст программы - полное, законченное и детальное описание алгоритма на языке программирования. Затем этот текст программы специальными служебными приложениями, которые называются трансляторами, либо переводится в машинный код (язык нулей и единиц), либо исполняется.

Языки программирования - искусственные языки. От естественных они отличаются ограниченным числом "слов", значение которых понятно транслятору, и очень строгими правилами записи команд (операторов).

Существует большое количество алгоритмов, в которых команды должны быть выполнены одна за другой. Такие алгоритмы называются линейными.

Программа имеет линейную структуру, если все операторы (команды) выполняются последовательно друг за другом.



Пример: программа, складывающая два числа

REM Сумма двух числа = 5

b = 6

c = a + b

```
PRINT "Результат: ", c
END
```

или так:

```
REM Сумма двух чисел
DIM a, b, c AS INTEGER
a = 5
b = 6
c = a + b
PRINT "Результат: ", c
END
```

Пример: Вычислите площадь прямоугольника по его сторонам.

```
REM Площадь прямоугольника
INPUT "Введите сторону a", a
INPUT "Введите сторону b", b
s = a * b
PRINT "Площадь равна: ", s
END
```

Разберем эти примеры.

Некоторые операторы языка Basic.

REM – оператор комментария. Все что следует после этого оператора до конца строки игнорируется компилятором и предназначено исключительно для человека. Т.е. здесь можно писать что угодно. Удобно использовать комментарий в начале программы для указания её названия и назначения.

PRINT (вывод, печать) – оператор вывода.

INPUT (ввод) – оператор ввода. Используется для передачи в программу каких-либо значений.

DIM – оператор описания типа переменной последовательного изменения значений переменных. В итоге - в определенных переменных получается результат. Переменная получает определенное значение в результате присваивания. Присваивание - это занесение в ячейку, отведенную под переменную, определенного значения в результате. Под переменной языках программирования понимают программный объект (число, слово, часть слова, несколько слов, символы), имеющий имя и значение, которое может быть получено и изменено программой. Если "заглянуть" в компьютер, то переменную можно определить так:

Переменная - это имя физического участка в памяти, в котором в каждый момент времени может быть только одно значение. Переменная - это ячейка в оперативной памяти компьютера для хранения какой-либо информации. Само название "переменная" подразумевает, что содержимое этого участка может изменяться. В качестве имен переменных могут быть латинские буквы с индексами. Причем может быть не одна буква, а несколько.

Пример:

```
DIM a, b, chislo1 AS INTEGER
```

Integer – целые числа от -32768 до 32768

Если в программе используются переменные не описанные с помощью оператора DIM, то компьютер будет рассматривать их как универсальные переменные. Это может привести к неэффективному использованию оперативной памяти. К тому же, такие программы не всегда легки для восприятия - плохо читаемы. Процесс решения вычислительной задачи - это процесс выполнения команды.

Для задания значения переменной служит **оператор присваивания**. Он записывается так:
LET переменная = значение (или просто: переменная = значение)
Ключевое слово LET в большинстве случаев не используется.

Пример:

```
LET a = 3  
chislo1 = 15
```

При выполнении оператора присваивания переменная, имя которой указано слева от знака равенства, получает значение, равное значению выражения (арифметического, строкового или логического), которое находится справа от знака равенства.

В результате операции **a=5** переменная **a** получает значение **5**.
В результате операции **c=a+b** переменная **c** получает значение равное сумме значений переменной **a** и **b**.

Программистам иногда бывает нужно поменять значения, хранящиеся в разных переменных. Например, в переменной **a** храниться число **3**, а в переменной **b** - число **5**. Как сделать так, чтобы переменные обменялись своими значениями?

Просто присвоить переменной **a** или **b** значение другой переменной нельзя, ведь тогда ее исходное значение будет перезаписано и утрачено. В таких случаях вводят дополнительную переменную (например, **temp**) и присваивают значения через нее.

Пример:

```
a=3  
b=5  
temp=a  
a=b  
b=temp
```

Иногда в программах (особенно с циклами) можно встретить такую запись: **s=s+i**. С точки зрения математики это совершенно бессмысленная запись, но рассмотрим её внимательней.

Оператор = это не равно, а оператор присваивания. **s=s+i** - звучит не "*переменная s равно переменная s плюс переменная i*", а так: "*переменной s присвоить значение равное сумме значений переменной s до присваивания и переменной i*"

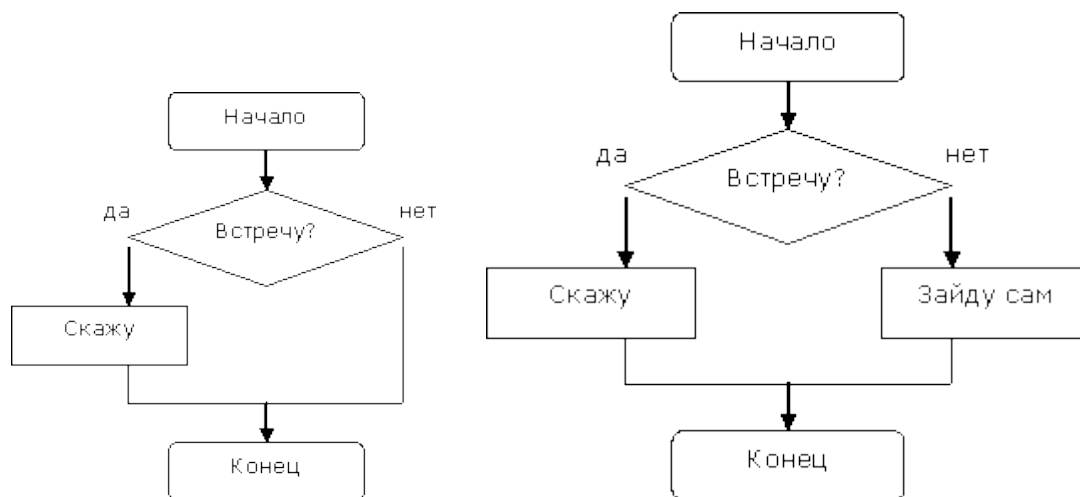
В результате операции **s=s+i** переменная **s** получает значение равное сумме предыдущего значения переменной **s** и значения переменной **i**. Т.е., если до операции присваивания значение **s** было равно **5**, а переменной **i** равно **3**, то после операции значение переменной **s** будет равно **8** (**5+3**, старое значение **s** + значение **i**).

END – оператор конца программы.

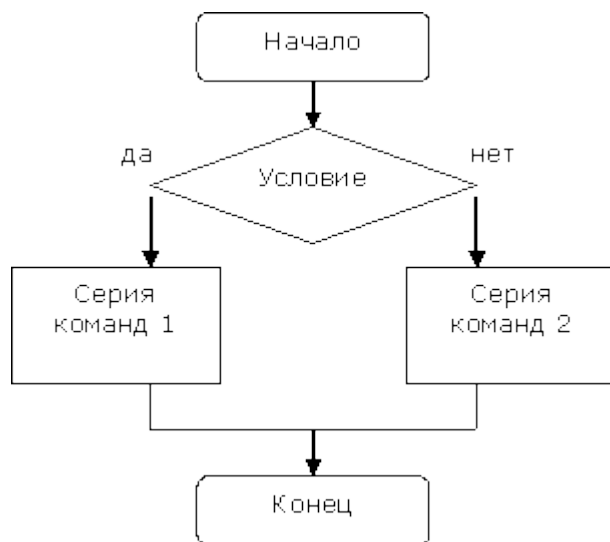
15. Алгоритмическая структура «ветвление». Команда ветвления. Примеры полного и неполного ветвления.

Разветвляющий алгоритм – это алгоритм, в котором в зависимости от условия выполняется либо одна, либо другая последовательность действий.

Во многих случаях требуется, чтобы при одних условиях выполнялась одна последовательность действий, а при других - другая.



Вся программа состоит из команд (операторов). Команды бывают простые и составные (команды, внутри которых встречаются другие команды). Составные команды часто называют управляющими конструкциями. Этим подчеркивается то, что эти операторы управляют дальнейшим ходом программы.



Рассмотрим запись условного оператора на языке Basic.

Простая форма оператора выглядит следующим образом:

IF <УСЛОВИЕ> THEN <ОПЕРАТОР>

или

```

IF                                <УСЛОВИЕ>
<ОПЕРАТОР                          1>
<ОПЕРАТОР                          2>
                                     ...
<ОПЕРАТОР                          N>
END IF

```

Если *условие справедливо*, то программа *выполняет* тот оператор, который стоит после ключевого слова **THEN** (или серию операторов от ключевого слова **THEN** до **END IF**), и дальше руководствуется обычным порядком действий. Если *условие не справедливо*, то оператор, стоящий после **THEN** (или серия операторов от **THEN** до **END IF**) *не выполняется*, и программа сразу переходит к обычному порядку действий. Конструкция **IF...THEN** позволяет в зависимости от справедливости условия либо выполнить оператор, либо пропустить этот оператор. Конструкция **IF...THEN...END IF** позволяет в зависимости от справедливости условия либо выполнить группу операторов, либо пропустить эту группу операторов.

Условия - еще один тип логических выражений. В них используются следующие *операторы сравнения*:

	равно
>	не равно
	больше
	меньше
=	больше или равно
=	меньше или равно

Справа и слева от знака сравнения должны стоять величины, относящиеся к одному типу. В результате сравнения получается логическая величина, имеющее значение ИСТИНА (TRUE) или ЛОЖЬ (FALSE).

Пример:
 $5 < 7$ - ИСТИНА; $8 = 12$ - ЛОЖЬ (проверяем равно ли 8 12, именно проверяем, а не утверждаем, что $8 = 12$);

Предыдущие конструкции позволяли обойти или выполнить серию оператор в зависимости от справедливости условия. Это еще не было ветвлением. Чтобы вычисления могли разветвляться по нескольким направлениям, служит конструкция **IF...THEN...ELSE...END IF**.

```

IF <УСЛОВИЕ> THEN
<ОПЕРАТОРЫ 1>
ELSE
<ОПЕРАТОРЫ 2>
END IF

```

Если условие справедливо (ИСТИНА), то выполняются <операторы 1> (стоящие между **THEN** и **ELSE**), а <операторы 2> (стоящие между **ELSE** и **END IF**) будут пропущены. Если условие не справедливо (ЛОЖЬ), то <операторы 1> игнорируются и выполняются <операторы 2>.

IF - если, THEN - тогда, ELSE - иначе.

Если в комнате темно, тогда надо включить свет.

Если пойдет дождь, **тогда** надо взять зонтик, **иначе**, зонтик не брать.

Пример: Проверить, равно ли введенное число некоторому значению, и в случае равенства выдать на экран сообщение о равенстве чисел.

```
REM сравнить число со каким-то значением
INPUT "Введите a", a
IF a=7 THEN PRINT "Числа равны"
END
```

После запуска программы проверяется равно ли введенное значение семи или нет. Если равно, то на экран выводится сообщение 'Числа равны'.

Пример: Определить большее из двух чисел, вывести его на экран, затем - увеличить его в двое и вывести результат на экран.

```
REM определить большее из двух чисел...
INPUT "Введите a", a
INPUT "Введите b", b
IF a>b THEN
PRINT "Большее число: ", a
c=2*a
ELSE
PRINT "Большее число: ", b
c=2*b
END IF
PRINT "результат: ", c
END
```

Сначала программа запрашивает оба числа, затем проверяет условие $a > b$. Если условие верно, то на экран выводится число a , затем это число удваивается. Иначе на экран выводится число b , затем число b удваивается. В завершении на экран выводится удвоенное значение большего числа.

Обратите внимание: программа имеет один недостаток - не учитывается тот случай, когда введенные числа равны. Исправим это, использовав вложение одного условия в другое.

```
REM определить большее из двух чисел...
INPUT "Введите a", a
INPUT "Введите b", b
IF a=b THEN
PRINT "Числа равны"
c=2*a
ELSE
IF a>b THEN
PRINT "Большее число: ", a
c=2*a
ELSE
PRINT "Большее число: ", b
c=2*b
END IF
END IF
PRINT "результат: ", c
END
```

В этой программе два условных оператора, первым проверяется условие равенства чисел и, в случае его выполнения, будет выдано сообщение о равенстве чисел, если числа не равны, то проверяется второе условие...

Пример: Решение квадратного уравнения.
Решение квадратного уравнения зависит от значения дискриминанта.

```
REM Решение квадратного уравнения
INPUT "Введите коэффициент a: ", a
INPUT "Введите коэффициент b: ", b
INPUT "Введите коэффициент c: ", c
d=b*b-4*a*c
IF d<0 THEN
PRINT "Корней нет"
ELSE
  IF d=0 THEN
    x=-b/(2*a)
    PRINT "корень уравнения: ", x
  ELSE
    x1=(-b-SQR(d))/(2*a)
    x2=(-b+SQR(d))/(2*a)
    PRINT "корни уравнения: ", x1, x2
  END IF
END IF
END
```

Структура "Выбор".

Структура **IF...** позволяет выбрать между двумя вариантами. Если требуется осуществить выбор между большим числом вариантов, то это можно организовать используя лишь структуру **IF...** Но можно (что чаще проще) и с помощью структуры "Выбор". Эта структура имеет вид:

```
SELECT CASE <Выражение>
CASE <условие 1>
<серия 1>
CASE <условие 2>
<серия 2>
...
CASE ELSE
<серия иначе>
END SELECT
```

Выражение, заданное после ключевых слов **SELECT CASE**, сравнивается с определенными значениями - условиями и если они истинны, то выполняется соответствующая серия команд. Если не одно условие не истинно, то выполняется серия команд между **CASE ELSE** и **END SELECT**.

Пример: Выдать словесное значение числа

```
REM Преобразование чисел в слова
INPUT "Введите число", a
SELECT CASE a
CASE 1
PRINT "один"
CASE 2
PRINT "два"
```

```

CASE 3
PRINT "три"
...
CASE 10
PRINT "десять"
CASE ELSE
PRINT "это число не могу перевести"
END SELECT
END

```

В данном примере введенное число сравнивается с числами от 1 до 10 и если наше число равно одному из этих чисел, то на экран выводится словесное значение числа. Если это не так на экран выводится сообщение: "это число не могу перевести".

16. Алгоритмическая структура «цикл». Циклы со счетчиком и циклы по условию.

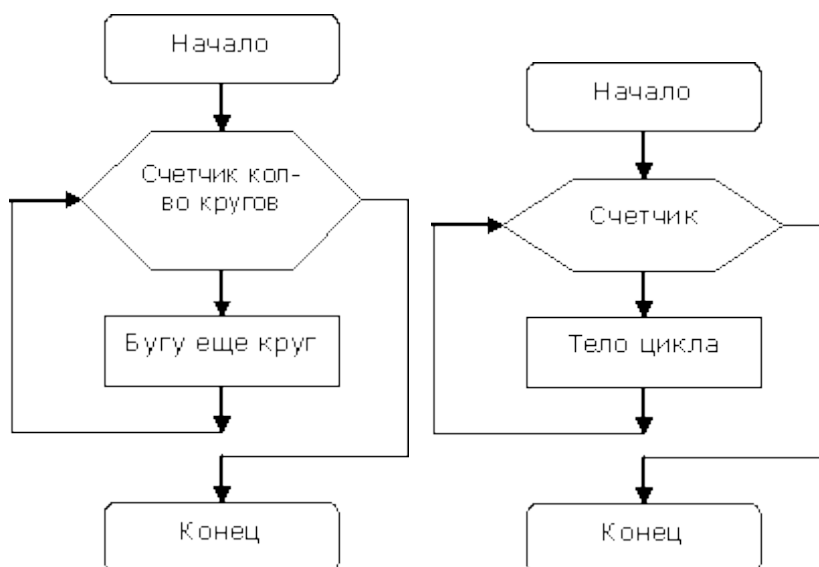
Лучшее качества компьютеров проявляются не тогда, когда они рассчитывают значения сложных выражений, а когда многократно, с незначительными изменениями, повторяют сравнительно простые операции. Даже очень простые расчеты могут поставить человека в тупик, если их надо повторить тысячи раз, а повторять операции миллионы раз человек совершенно не способен.

С необходимостью повторяющихся вычислений программисты сталкиваются постоянно. Например, если надо подсчитать, сколько раз буква "о" встречается в тексте необходимо перебрать все буквы. При всей простоте этой программы исполнить ее человеку очень трудно, а для компьютера это задача на несколько секунд.

Циклический алгоритм - описание действий, которые должны повторяться указанное число раз или пока не выполнено заданное условие.

Перечень повторяющихся действий называют телом цикла.

Например, на уроке физкультуры вы должны пробежать некоторое количество кругов вокруг стадиона.



Такие циклы называются - **циклы со счетчиком**.

На языке Basic они записываются следующим образом:

```
FOR Счетчик=НачЗнач TO КонЗнач [STEP шаг]  
тело цикла  
NEXT [Счетчик]
```

Параметры указанные в квадратных скобках являются не обязательными (их можно не записывать). По умолчанию шаг цикла равен одному, т.е. каждый раз после прохождения тела цикла счетчик увеличивается на единицу.

Пример: Вывести на экран все числа от 1 до 100. Для этого можно было бы написать следующую программу:

```
REM Вывод чисел от 1 до 100  
PRINT 1  
PRINT 2  
PRINT 3  
PRINT 4  
PRINT 5  
PRINT 6  
PRINT 7  
...  
PRINT 98  
PRINT 99  
PRINT 100  
END
```

Всего каких-то 102 строчки ;-). Хотя эту же программу можно написать намного короче:

```
REM Вывод чисел от 1 до 100  
FOR I=1 TO 100  
PRINT I  
NEXT  
END
```

Немного исправив программу можно сделать, чтобы она выводила все числа от а до b.

```
REM Вывод чисел от а до b  
a=55  
b=107  
FOR I=a TO b  
PRINT I  
NEXT  
END
```

В этом случае счетчик при первом прохождении цикла принимает значение переменной а, после чего выполняются операторы до ключевого слова NEXT. После этого счетчик увеличивается на единицу и сравнивается со значение переменной b, если счетчик меньше, то цикл выполняется еще.

Легко сделать чтобы программа выводила числа в обратном порядке. Для этого шаг цикла должен быть равен -1 (минус один). В этом случае значение счетчика каждый раз после прохождения цикла будет уменьшено на единицу.

```
REM Вывод чисел от b до a
a=55
b=107
FOR I=b TO a STEP -1
PRINT I
NEXT
END
```

Пример: Вычислить сумму двузначных натуральных чисел.

```
REM Вычислить сумму двузначных натуральных чисел
FOR I=10 TO 99
s=s+I
NEXT
PRINT "Результат = ",s
END
```

Программа перебирает числа от 10 до 99 каждый раз выполняя действия $s=s+I$. С точки зрения математики это совершенно бессмысленная запись, но рассмотрим её внимательней. Процесс решения вычислительной задачи - это процесс последовательного изменения значений переменных. В итоге - в определенных переменных получается результат. Переменная получает определенное значение в результате *присваивания*. Вы помните, что присваивание - это занесение в ячейку, отведенную под переменную, определенного значения в результате выполнения команды.

В результате операции $a=5$ переменная **a** получает значение 5.

В результате операции $c=a+b$ переменная **c** получает значение равное сумме значений переменной **a** и **b**.

В результате операции $s=s+I$ переменная **s** получает значение равное сумме предыдущего значения переменной **s** и значения переменной **I**. Т.е., если до операции присваивания значение **s** было равно 5, а переменной **I** равно 3, то после операции значение переменной **s** будет равно 8 ($5+3$, старое значение **s** + значение **I**).

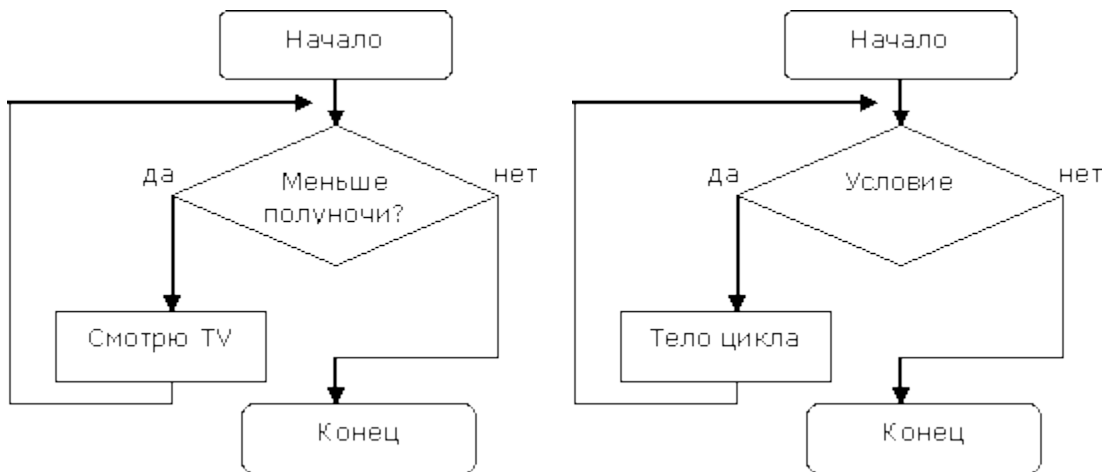
Значит после выполнения нашей программы в переменной **s** будет храниться сумма всех двузначных чисел от 10 до 99.

Пример: вычислить факториал числа **a** (записывается так: $a!$). Факториал - это произведение чисел от 1 до **a**. Например, $5!$ (факториал пяти) - это $5!=1*2*3*4*5$.

```
REM Вычислить факториал числа
a=5
f=1
FOR I=1 TO a
f=f*I
NEXT
PRINT f
END
```

Вы, конечно, заметили, что до начала цикла мы присвоили переменной **f** значение равное единице. Иначе бы мы получили в результате ноль.

В субботу вечером вы смотрите телевизор. Время от времени поглядываете на часы и если время меньше полуночи, то продолжаете смотреть телевизор, если это не так, то вы прекращаете просмотр телепередач.



Циклы такого вида называют - **циклы с предусловием**.

На языке Basic они записываются следующим образом:

DO WHILE условие

Тело цикла

LOOP

В этом цикле проверяется условие и если оно выполняется (ИСТИНА), то выполняется тело цикла до ключевого слова LOOP, затем условие проверяется снова ... и так до тех пор пока условие истинно.

DO UNTIL условие

Тело цикла

LOOP

Этот цикл отличается от предыдущего только тем, что он выполняется до тех пор пока условие не истинно (т.е. совсем наоборот).

Пример: Вывести все натуральные числа меньше данного.

```

REM Вывод всех чисел меньше данного
a=0
chislo=10
DO WHILE a<chislo
PRINT a
a=a+1
LOOP
END

```

Стоит обратить внимание на то, что цикл *может быть не выполнен ни разу* (если условие первоначально не истинно, например, **a=5**, а **chislo=4**). И наоборот, если условие будет истинно при любых значениях переменных, то цикл будет *выполняться бесконечное число раз* (произойдет *заикливание*).

Пример цикла, который будет выполняться бесконечное число раз:

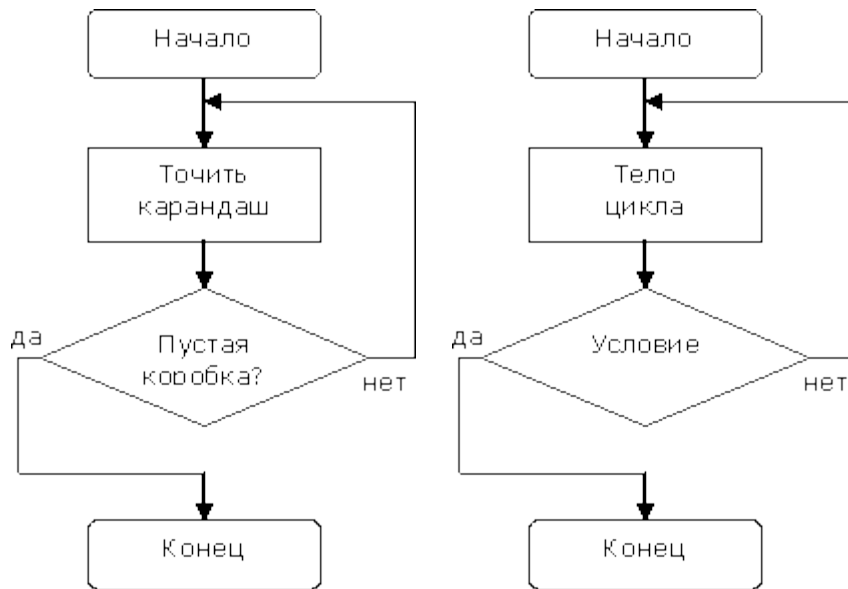
```

REM заикливание
DO WHILE a=a
PRINT "Это сообщение будет выводиться на экран постоянно"
LOOP

```

```
PRINT "А это сообщение вы не увидите никогда"  
END
```

Вам надо поточить все карандаши в коробке. Вы точите один карандаш и откладываете его в сторону. Затем проверяете, остались ли карандаши в коробке. Если условие ложно, то снова выполняется действие 'заточить карандаш'. Как только условие становится истинным, то цикл прекращается.



Циклы такого вида называют - **циклы с постусловием**.

На языке Basic они записываются следующим образом:

```
DO  
Тело цикла  
LOOP WHILE условие
```

```
DO  
Тело цикла  
LOOP UNTIL условие
```

Циклы такого рода отличаются тем, что хоть один раз, но тело цикла будет выполнено вне зависимости от условия. Условие проверяется после первого выполнения тела цикла.

Пример: Вычислите сумму цифр в числе.

```
REM Сумма цифр числа  
DIM a, chislo, s AS INTEGER  
INPUT "Введите число: ", chislo  
a=chislo  
DO  
s=s+a MOD 10  
a=a/10  
a=INT(a)  
LOOP UNTIL a=0  
PRINT "Сумма цифр числа ",chislo ," равна: ", s  
END
```

Переменную s используем для хранения суммы цифр. До начала цикла в переменную a заносим значение переменной $chislo$. Все дальнейшие преобразования осуществляем с переменной a . В цикле берем остаток от деления на 10 (последняя цифра числа) и прибавляем к тому, что уже есть в переменной s ; делим значение переменной a на 10, берем целую часть (т.е. отбрасываем последнюю цифру числа) и заносим в a . Цикл продолжается до тех пор пока значение переменной a не окажется равным нулю (перебрали все цифры числа). Результат выводим на экран.

17. Технология решения задач с помощью компьютера (моделирование, формализация, алгоритмизация, программирование). Показать на примере задачи (математической, физической или другой).

Рассмотрим процесс решения задачи на конкретном примере:

Тело брошено вертикально вверх с начальной скоростью с некоторой высоты. Определить его местоположение и скорость в заданный момент времени.

На первом этапе обычно строится описательная информационная модель объекта или процесса. В нашем случае с использованием физических понятий создается идеализированная модель движения объекта. Из условия задачи можно сформулировать следующие основные предположения:

- 1) тело мало по сравнению с Землей, поэтому его можно считать материальной точкой;
- 2) скорость бросания тела мала, поэтому:
— ускорение свободного падения считать постоянной величиной;
— сопротивлением воздуха можно пренебречь.

На втором этапе создается формализованная модель, т. е. описательная информационная модель записывается с помощью какого-либо формального языка.

Из курса физики известно, что описанное выше движение является равноускоренным. При заданных начальной скорости (V_0), начальной высоте (H_0) и ускорении свободного падения ($g = 9,8 \text{ м/с}^2$) зависимость скорости (V) и высоты (H) от времени (t) можно описать следующими математическими формулами:
 $V = V_0 - gt$, $Y = H_0 + Vt - gt^2/2$.

На третьем этапе необходимо формализованную информационную модель преобразовать в компьютерную на понятном для компьютера языке. Существуют два принципиально различных пути построения компьютерной модели:
— создание алгоритма решения задачи и его кодирование на одном из языков программирования;
— формирование компьютерной модели с использованием одного из приложений (электронных таблиц, СУБД и т. д.).

Для реализации первого пути надо построить алгоритм определения координаты тела в определенный момент времени и закодировать его на одном из языков программирования, например на языке Visual Basic.

Второй путь требует создания компьютерной модели, которую можно исследовать в электронных таблицах. Для этого следует представить математическую модель в форме таблицы функции зависимости координаты от времени (таблицы функции $H = H_0 + Vt - gt^2/2$) и таблицы зависимости скорости тела от времени ($V = V_0 - g \cdot t$).

Четвертый этап исследования информационной модели состоит в проведении компьютерного эксперимента. Если компьютерная модель существует в виде программы на одном из языков программирования, ее нужно запустить на выполнение и получить результаты. Если

компьютерная модель исследуется в приложении, например в электронных таблицах, можно провести сортировку или поиск данных, построить диаграмму или график и т. д.

На пятом этапе выполняется анализ полученных результатов и при необходимости корректировка исследуемой модели. Например, в нашей модели необходимо учесть, что не имеет физического смысла вычисление координаты тела после его падения на поверхность Земли.

Таким образом, технология решения задач с помощью компьютера состоит из следующих этапов: построение описательной модели — формализация — построение компьютерной модели — компьютерный эксперимент — анализ результатов и корректировка модели.

18. Программные средства и технологии обработки текстовой информации (текстовый редактор, текстовый процессор, редакционно-издательские системы).

Текстовые редакторы

Для обработки текстовой информации на компьютере используются приложения общего назначения - текстовые редакторы.

Текстовые редакторы — это программы для создания, редактирования, форматирования, сохранения и печати документов. Современный документ может содержать, кроме текста, и другие- объекты (таблицы, диаграммы, рисунки и т. д.).

Простые текстовые редакторы (например, Блокнот) позволяют редактировать текст и осуществлять простейшее форматирование шрифта.

Более совершенные текстовые редакторы, имеющие целый спектр возможностей по созданию документов (например, поиск и замена символов, средства проверки орфографии, вставка таблиц и др.), называют иногда текстовыми процессорами. Примером такой программы является Word из офисного пакета MicrosoftOffice, или Writer из пакета StarOffice.

Мощные программы обработки текста — настольные издательские системы — предназначены для подготовки документов к публикации. Пример подобной системы — Adobe PageMaker.

Для подготовки к публикации в Интернете web-страниц используют специализированные приложения (например, Microsoft FrontPage или Macromedia Dreamweaver).

Редактирование и форматирование

Редактирование — преобразование, обеспечивающее добавление, удаление, перемещение или исправление содержания документа.

Редактирование документа обычно производится путем добавления, удаления или перемещения символов или фрагментов текста. Объектно-ориентированный подход дает возможность реализовать механизм встраивания и внедрения объектов (OLE — Object Linking Embedding). Этот механизм позволяет копировать и вставлять объекты из одного приложения в другое. Например, работая с документом в текстовом редакторе Word, в него можно встроить изображения, анимацию, звук и даже видеофрагменты и таким образом из обычного текстового документа получить мультимедиа-документ.

Форматирование — преобразование, изменяющее форму представления документа.

Любой документ состоит из страниц, поэтому в начале работы над документом необходимо задать значения параметров страницы: формат, ориентацию, поля и др. Стандартным является формат страницы А4 (21х29,7 см). Существуют две возможные ориентации страницы – книжная и альбомная. Для обычных текстов чаще используется книжная ориентация, а для таблиц с большим количеством столбцов – альбомная.

Форматирование абзацев.

Абзац с литературной точки зрения – это часть текста, представляющая собой законченный по смыслу фрагмент произведения, окончание которого служит естественной паузой для перехода к новой мысли.

В компьютерных документах абзацем считается любой текст, заканчивающийся управляющим символом конца абзаца. Ввод конца абзаца обеспечивается нажатием клавиши ВВОД (ENTER).

Форматирование абзацев позволяет подготовить правильно и красиво оформленный документ.

В процессе форматирования абзаца задаются параметры его выравнивания (выравнивание отражает расположение текста относительно границ полей страницы), отступы (абзац целиком может иметь отступы слева и справа) и интервалы (расстояние между строками абзаца), отступ красной строки и др.

Форматирование шрифта (символов).

Символы – это буквы, цифры, пробелы, знаки пунктуации, специальные символы. Символы можно форматировать (изменять их внешний вид). Среди основных свойств символов можно выделить следующие: шрифт, размер, начертание и цвет.

Шрифт – это полный набор символов определенного начертания. Каждый шрифт имеет своё название, например Times New Roman, Arial, Comic Sans MS. Единицей измерения шрифта является пункт (1 пт = 0,367 мм). Размеры шрифтов можно изменять в больших пределах. Кроме нормального (обычного) начертания символов обычно применяют полужирное, курсивное, полужирное курсивное.

По способу представления в компьютере различаются шрифты растровые и векторные. Для представления растровых шрифтов служат методы растровой графики, символы шрифта — это группы пикселей. Растровые шрифты допускают масштабирование только с определенными коэффициентами.

В векторных шрифтах символы описываются математическими формулами и возможно произвольное их масштабирование. Среди векторных шрифтов наибольшее распространение получили шрифты типа TrueType.

Можно также установить дополнительные параметры форматирования символов: подчеркивание символов различными типами линий, изменение вида символов (верхний и нижний индекс, зачеркнутый), изменение расстояний между символами.

Если планируется цветная печать документа, то можно задать различные цвета для различных групп символов.

Проверка орфографии и синтаксиса

Для проверки орфографии и синтаксиса используются специальные программные модули, которые обычно включаются в состав текстовых процессоров и издательских систем. Такие

системы содержат словари и грамматические правила для нескольких языков, что позволяет исправлять ошибки в многоязычных документах.

Формат файла

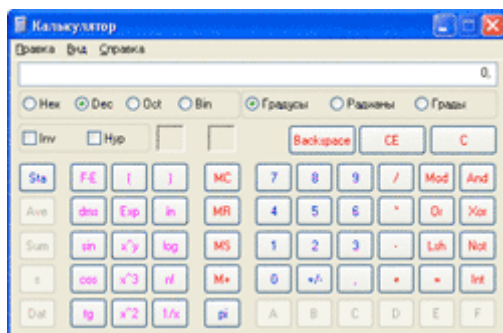
Формат файла определяет способ хранения текста в файле. Простейший формат текстового файла (TXT) содержит только символы (числовые коды символов), другие же форматы (DOC, RTF) содержат дополнительные управляющие числовые коды, которые обеспечивают форматирование текста.

19. Программные средства и технологии обработки числовой информации (электронные калькуляторы и электронные таблицы).

Электронные калькуляторы

Электронные калькуляторы являются специализированными программными приложениями, предназначенными для произведения вычислений. Электронные калькуляторы по своим функциональным возможностям соответствуют аппаратным микрокалькуляторам.

Аппаратные микрокалькуляторы могут существенно различаться по своим возможностям и областям применения. Простые микрокалькуляторы позволяют осуществлять только арифметические операции над числами и используются в быту. Инженерные микрокалькуляторы позволяют также вычислять значения различных функций (\sin , \cos и др.) и используются в процессе обучения и для инженерных расчетов; программистские микрокалькуляторы позволяют проводить вычисления в различных системах счисления и другие операции.



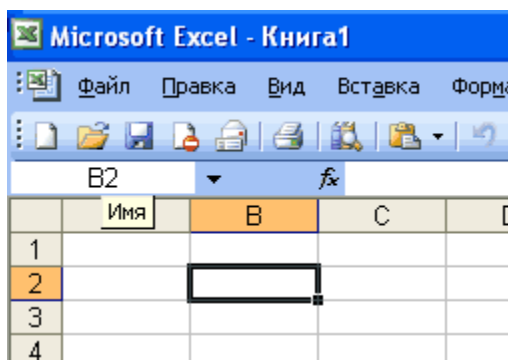
Электронные калькуляторы гораздо удобнее, так как могут обладать возможностями всех вышеперечисленных типов аппаратных микрокалькуляторов. Электронный Калькулятор является стандартным приложением операционной системы Windows. Также кроме стандартной программы Калькулятор существует не мало других удобных калькуляторов.

Электронные таблицы

Электронная таблица — это программа обработки числовых данных, хранящая и обрабатывающая данные в прямоугольных таблицах.

Электронная таблица состоит из столбцов и строк. Заголовки столбцов обозначаются буквами или сочетаниями букв (A, G, AB и т. п.), заголовки строк — числами (1, 16, 278 и т. п.). Ячейка — место пересечения столбца и строки.

Каждая ячейка таблицы имеет свой собственный адрес. Адрес ячейки электронной таблицы составляется из заголовка столбца и заголовка строки, например: A1, B5, E7. Ячейка, с которой производятся какие-то действия, выделяется рамкой и называется активной.



Электронные таблицы, с которыми работает пользователь в приложении, называются рабочими листами. Можно вводить и изменять данные одновременно на нескольких рабочих листах, а также выполнять вычисления на основе данных из нескольких листов. Документы электронных таблиц могут включать несколько рабочих листов и называются рабочими книгами.

Типы данных.

Электронные таблицы позволяют работать с тремя основными типами данных: число, текст и формула.

Числа в электронных таблицах Excel могут быть записаны в обычном числовом или экспоненциальном формате, например: 195,2 или 1.952E+02. По умолчанию числа выравниваются в ячейке по правому краю. Это объясняется тем, что при размещении чисел друг под другом (в столбце таблицы) удобно иметь выравнивание по разрядам (единицы под единицами, десятки под десятками и т. д.).

Текстом в электронных таблицах Excel является последовательность символов, состоящая из букв, цифр и пробелов, например запись «32 Мбайт» является текстовой. По умолчанию текст выравнивается в ячейке по левому краю. Это объясняется традиционным способом письма (слева направо).

Формула должна начинаться со знака равенства и может включать в себя числа, Имена ячеек, функции (Математические, Статистические, Финансовые, Дата и время и т.д.) и знаки математических: операций. Например, формула «=A1+B2» обеспечивает сложение чисел, хранящихся в ячейках A1 и B2, а формула «=A1*5» — умножение числа, хранящегося в ячейке A1, на 5. При вводе формулы в ячейке отображается не сама формула, а результат вычислений по этой формуле. При изменении исходных значений, входящих в формулу, результат пересчитывается немедленно.

	C1			fx
	A	B	C	
1	3	5		=A1*B1
			15	

Абсолютные и относительные ссылки.

В формулах используются ссылки на адреса ячеек. Существуют два основных типа ссылок: относительные и абсолютные. Различия между ними проявляются при копировании формулы из активной ячейки в другую ячейку.

Относительная ссылка в формуле используется для указания адреса ячейки, вычисляемого относительно ячейки, в которой находится формула. При перемещении или копировании формулы из активной ячейки относительные ссылки автоматически обновляются в зависимости от нового положения формулы. Относительные ссылки имеют следующий вид: A1, B3.

Абсолютная ссылка в формуле используется для указания фиксированного адреса ячейки. При перемещении или копировании формулы абсолютные ссылки не изменяются. В абсолютных ссылках перед неизменяемым значением адреса ячейки ставится знак доллара (например, \$A\$1).

Если символ доллара стоит перед буквой (например: \$A1), то координата столбца абсолютная, а строки — относительная. Если символ доллара стоит перед числом (например, A\$1), то, наоборот, координата столбца относительная, а строки — абсолютная. Такие ссылки называются смешанными.

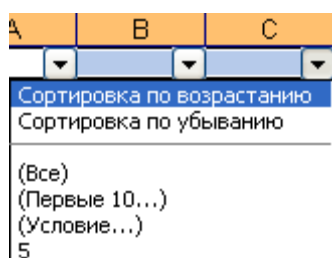
Пусть, например, в ячейке C1 записана формула =A\$1+\$B1, которая при копировании в ячейку D2 приобретает вид =B\$1+\$B2.

Относительные ссылки при копировании изменились, а абсолютные — нет.

Сортировка и поиск данных.

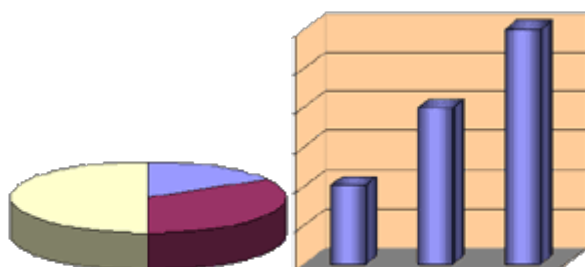
Электронные таблицы позволяют осуществлять сортировку данных. Данные в электронных таблицах сортируются по возрастанию или убыванию. При сортировке данные выстраиваются в определенном порядке. Можно проводить вложенные сортировки, т. е. сортировать данные по нескольким столбцам, при этом назначается последовательность сортировки столбцов.

В электронных таблицах возможен поиск данных в соответствии с указанными условиями — фильтрами. Фильтры определяются с помощью условий поиска (больше, меньше, равно и т. д.) и значений (100, 10 и т. д.). Например, больше 100. В результате поиска будут найдены те ячейки, в которых содержатся данные, удовлетворяющие заданному фильтру.



Построение диаграмм и графиков.

Электронные таблицы позволяют представлять числовые данные в виде диаграмм или графиков. Диаграммы бывают различных типов (столбчатые, круговые и т. д.); выбор типа диаграммы зависит от характера данных.



20. Компьютерная графика. Аппаратные средства (монитор, видеокарта, видеоадаптер, сканер и др.). Программные средства (растровые и векторные графические редакторы, средства деловой графики, программы анимации и др.).

Аппаратные средства



Устройства вывода информации

Монитор.

Монитор является универсальным устройством вывода информации и подключается к видеокарте, установленной в компьютере. Изображение в компьютерном формате (в виде последовательностей нулей и единиц) хранится в видеопамяти, размещенной на видеокарте. Изображение на экране монитора формируется путем считывания содержимого видеопамяти и отображения его на экран. Частота считывания изображения влияет на стабильность изображения на экране. В современных мониторах обновление изображения происходит обычно с частотой 75 и более раз в секунду, что обеспечивает комфортность восприятия изображения пользователем компьютера (человек не замечает мерцание изображения). Для сравнения можно напомнить, что частота смены кадров в кино составляет 24 кадра в секунду.

В настольных компьютерах обычно используются мониторы на электронно-лучевой трубке (ЭЛТ). Изображение на экране монитора создается пучком электронов, испускаемых электронной пушкой. Этот пучок электронов разгоняется высоким электрическим напряжением (десятки киловольт) и падает на внутреннюю поверхность экрана, покрытую люминофором (веществом, светящимся под воздействием пучка электронов).

Система управления пучком заставляет пробегать его построчно весь экран (создает растр), а также регулирует его интенсивность (соответственно яркость свечения точки люминофора). Пользователь видит изображение на экране монитора, так как люминофор излучает световые лучи в видимой части спектра. Качество изображения тем выше, чем меньше размер точки изображения (точки люминофора), в высококачественных мониторах размер точки составляет 0,22 мм.

Однако монитор является также источником высокого статического электрического потенциала, электромагнитного и рентгеновского излучений, которые могут оказывать неблагоприятное воздействие на здоровье человека. Современные мониторы практически безопасны, так как соответствуют жестким санитарно-гигиеническим требованиям, зафиксированным в международном стандарте безопасности ТСО'99.

В портативных и карманных компьютерах применяют плоские мониторы на жидких кристаллах (ЖК). В последнее время такие мониторы стали использоваться и в настольных компьютерах.



ЖК-мониторы сделаны из вещества, которое находится в жидком состоянии, но при этом обладает некоторыми свойствами, присущими кристаллическим телам. Фактически это жидкости, обладающие анизотропией свойств (в частности, оптических), связанных с упорядоченностью в ориентации молекул. Молекулы жидких кристаллов под воздействием электрического напряжения могут изменять свою ориентацию и вследствие этого изменять свойства светового луча, проходящего сквозь них.

Преимущество ЖК-мониторов перед мониторами на ЭЛТ состоит в отсутствии вредных для человека электромагнитных излучений и компактности.

Мониторы могут иметь различный размер экрана. Размер диагонали экрана измеряется в дюймах (1 дюйм = 2,54 см) и обычно составляет 15, 17, 19 и более дюймов.



Принтеры.

Принтеры предназначены для вывода на бумагу (создания «твердой копии») числовой, текстовой и графической информации. По своему принципу действия принтеры делятся на матричные, струйные и лазерные.

Матричные принтеры — это принтеры ударного действия. Печатающая головка матричного принтера состоит из вертикального столбца маленьких стержней (обычно 9 или 24), которые под воздействием магнитного поля «выталкиваются» из головки и ударяют по бумаге (через красящую ленту). Перемещаясь, печатающая головка оставляет на бумаге строку символов.

Недостатки матричных принтеров состоят в том, что они печатают медленно, производят много шума и качество печати оставляет желать лучшего (соответствует примерно качеству пишущей машинки).

Струйные принтеры

В последние годы широкое распространение получили черно-белые и цветные струйные принтеры. В них используется чернильная печатающая головка, которая под давлением выбрасывает чернила из ряда мельчайших отверстий на бумагу. Перемещаясь вдоль бумаги, печатающая головка оставляет строку символов или полосу изображения.

Струйные принтеры могут печатать достаточно быстро (до нескольких страниц в минуту) и производят мало шума. Качество печати (в том числе и цветной) определяется разрешающей способностью струйных принтеров, которая может достигать фотографического качества 2400 dpi. Это означает, что полоска изображения по горизонтали длиной в 1 дюйм формируется из 2400 точек (чернильных капель).

Лазерные принтеры обеспечивают практически бесшумную печать. Высокую скорость печати (до 30 страниц в минуту) лазерные принтеры достигают за счет постраничной печати, при которой страница печатается сразу целиком.

Высокое типографское качество печати лазерных принтеров обеспечивается за счет высокой разрешающей способности, которая может достигать 1200 dpi и более.

Плоттер.

Для вывода сложных и широкоформатных графических объектов (плакатов, чертежей, электрических и электронных схем и пр.) используются специальные устройства вывода — плоттеры.

Принцип действия плоттера такой же, как и струйного принтера.

Устройства ввода информации

Сенсорный экран

Сенсорный, или тактильный, экран представляет собой поверхность, которая покрыта специальным слоем. Прикосновение к определенному месту экрана обеспечивает выбор задания, которое должно быть выполнено компьютером, или команды в экранном меню.



Сенсорный экран позволяет также перемещать объекты. Он удобен в использовании, особенно когда необходим быстрый доступ к информации. Такие устройства ввода можно увидеть в банковских компьютерах, аэропортах, а также в военной сфере и промышленности.

Световое перо

Световое перо похоже на обычный карандаш, на кончике которого имеется специальное устройство — светочувствительный элемент.

Соприкосновение пера с экраном замыкает фотоэлектрическую цепь и определяет место ввода или коррекции данных. Если перемещать по экрану такое перо, можно рисовать или писать на экране, как на листе бумаги.

Световое перо используется для ввода информации в самых маленьких персональных компьютерах — в карманных микрокомпьютерах. Оно также применяется в различных системах проектирования и дизайна.

Графический планшет, или дигитайзер

Графический планшет, или дигитайзер, используется для создания либо копирования рисунков или фотографий. Он позволяет создавать рисунки так же, как на листе бумаги. Изображение преобразуется в цифровую форму, отсюда название устройства (от англ. digit — цифра).

С помощью специальной ручки можно чертить, рисовать схемы, добавлять заметки и подписи к электронным документам. Качество графических планшетов характеризуется разрешающей способностью, которая измеряется в lpi (линиях на дюйм) и способностью реагировать на силу нажатия пера.

В хороших планшетах разрешающая способность достигает 2048 lpi (перемещение пера по поверхности планшета на 1 дюйм соответствует перемещению на 2048 точек на экране монитора), а количество воспринимаемых градаций нажатий на перо составляет 1024.

Условия создания изображения приближены к реальным, достаточно специальным пером или пальцем сделать рисунок на специальной поверхности. Результат работы дигитайзера воспроизводится на экране монитора и в случае необходимости может быть распечатан на принтере. Дигитайзерами обычно пользуются архитекторы, дизайнеры.

Сканер

Большое распространение в наше время приобрели устройства сканирования изображений, таких как тексты или рисунки. Термин «сканирование» происходит от английского глагола to scan, что означает «пристально всматриваться».

Сканер предназначен для ввода в компьютер графической или текстовой информации с листа бумаги, со страницы журнала или книги. Для работы сканера необходимо программное обеспечение, которое создает и сохраняет в памяти электронную копию изображения. Все разнообразие подобных программ можно подразделить на два класса — для работы с графическим изображением и для распознавания текста.

Сканируемое изображение освещается белым светом (черно-белые сканеры) или тремя цветами (красным, зеленым и синим). Отраженный свет проецируется на линейку фотоэлементов,



которая движется, последовательно считывает изображение и преобразует его в компьютерный формат. В отсканированном изображении количество различаемых цветов может достигать десятков миллиардов.

Сканеры различаются по следующим параметрам:

глубина распознавания цвета: черно-белые, с градацией серого, цветные;

оптическое разрешение, или точность сканирования, измеряется в точках на дюйм и определяет количество точек, которые сканер различает на каждом дюйме;

К важным характеристикам сканера также относятся время сканирования и максимальный размер сканируемого документа. Сканеры находят широкое применение в издательской деятельности, системах проектирования, анимации. Эти устройства незаменимы при создании презентаций, докладов, рекламных материалов высокого качества.

Разрешающая способность сканеров составляет 600 dpi и выше, то есть на полоске изображения длиной 1 дюйм сканер может распознать 600 и более точек.

Цифровые камеры и ТВ-тюнеры

Последние годы все большее распространение получают цифровые камеры (видеокамеры и фотоаппараты). Цифровые камеры позволяют получать видеоизображение и фотоснимки непосредственно в цифровом (компьютерном) формате.

Цифровые видеокамеры могут быть подключены к компьютеру, что позволяет сохранять видеозаписи в компьютерном формате.

Для передачи «живого» видео по компьютерным сетям используются недорогие web-камеры, разрешающая способность которых обычно не превышает 640x480 точек. Цифровые фотоаппараты позволяют получать высококачественные фотографии с разрешением до 2272x1704 точек (всего до 3,9 млн пикселей). Для хранения фотографий используются модули flash-памяти или жесткие диски очень маленького размера. Запись изображений на жесткий диск компьютера может осуществляться путем подключения камеры к компьютеру.

Если установить в компьютер специальную плату (ТВ-тюнер) и подключить к ее входу телевизионную антенну, то появляется возможность просматривать телевизионные передачи непосредственно на компьютере.

Программные средства

Для обработки изображений на компьютере используются специальные программы — графические редакторы. **Графический редактор** — это программа создания, редактирования и просмотра графических изображений. Графические редакторы можно разделить на две категории: растровые и векторные.

Растровые графические редакторы. Растровые графические редакторы являются наилучшим средством обработки фотографий и рисунков, поскольку растровые изображения обеспечивают высокую точность передачи градаций цветов и полутонов. Среди растровых графических редакторов есть простые, например стандартное приложение Paint, и мощные профессиональные графические системы, например Adobe Photoshop.

Растровое изображение хранится с помощью точек различного цвета (пикселей), которые образуют строки и столбцы. Любой пиксель имеет фиксированное положение и цвет. Хранение

каждого пикселя требует некоторого количества бит информации, которое зависит от количества цветов в изображении.

Качество растрового изображения определяется размером изображения (числом пикселей по горизонтали и вертикали) и количества цветов, которые могут принимать пиксели.

Растровые изображения очень чувствительны к масштабированию (увеличению или уменьшению). Когда растровое изображение уменьшается, несколько соседних точек превращаются в одну, поэтому теряется разборчивость мелких деталей изображения. При укрупнении изображения увеличивается размер каждой точки и появляется ступенчатый эффект, который виден невооруженным глазом.



Векторные графические редакторы. Векторные графические изображения являются оптимальным средством для хранения высокоточных графических объектов (чертежи, схемы и т. д.) для которых имеет значение наличие четких и ясных контуров. С векторной графикой вы сталкиваетесь, когда работаете с системами компьютерного черчения и автоматизированного проектирования, с программами обработки трехмерной графики.

К векторным графическим редакторам относятся графический редактор, встроенный в текстовый редактор Word. Среди профессиональных векторных графических систем наиболее распространены CorelDRAW и Adobe Illustrator.

Векторные изображения формируются из объектов (точка, линия, окружность и т. д.), которые хранятся в памяти компьютера в виде графических примитивов и описывающих их математических формул.

Например, графический примитив точка задается своими координатами (X, Y) , линия — координатами начала $(X1, Y1)$ и конца $(X2, Y2)$, окружность — координатами центра (X, Y) и радиусом (K) , прямоугольник — величиной сторон и координатами левого верхнего угла $(X1, Y1)$ и правого нижнего угла $(X2, Y2)$ и т. д. Для каждого примитива назначается также цвет.

Достоинством векторной графики является то, что файлы, хранящие векторные графические изображения, имеют сравнительно небольшой объем. Важно также, что векторные графические изображения могут быть увеличены или уменьшены без потери качества.

Панели инструментов графических редакторов. Графические редакторы имеют набор инструментов для создания или рисования простейших графических объектов: прямой линии, кривой, прямоугольника, эллипса, многоугольника и т. д. После выбора объекта на панели инструментов его можно нарисовать в любом месте окна редактора.

Выделяющие инструменты. В графических редакторах над элементами изображения возможны различные операции: копирование, перемещение, удаление, поворот, изменение размеров и т. д. Чтобы выполнить какую-либо операцию над объектом, его сначала необходимо выделить.

Для выделения объектов в растровом графическом редакторе обычно имеются два инструмента: выделение прямоугольной области и выделение произвольной области. Процедура выделения аналогична процедуре рисования.

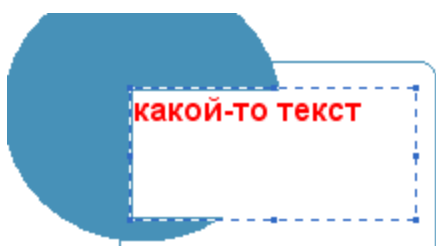
Выделение объектов в векторном редакторе осуществляется с помощью инструмента выделение объекта (на панели инструментов изображается стрелкой). Для выделения объекта достаточно выбрать инструмент выделения и щелкнуть по любому объекту на рисунке.

Инструменты редактирования рисунка позволяют вносить в рисунок изменения: стирать его части, изменять цвета и т. д. Для стирания изображения в растровых графических редакторах используется инструмент Ластик, который убирает фрагменты изображения (пиксели), при этом размер Ластика можно менять.

В векторных редакторах редактирование изображения возможно только путем удаления объектов, входящих в изображение, целиком. Для этого сначала необходимо выделить объект, а затем выполнить операцию Вырезать.

Операцию изменения цвета можно осуществить с помощью меню Палитра, содержащего набор цветов, используемых при создании или рисовании объектов.

Текстовые инструменты позволяют добавлять в рисунок текст и форматировать его.



В растровых редакторах инструментом Надпись (буква А на панели инструментов) создаются текстовые области на рисунках. Установив курсор в любом месте текстовой области, можно ввести текст. Форматирование текста производится с помощью панели Атрибуты текста.

В векторных редакторах тоже можно создавать текстовые области для ввода и форматирования текста. Кроме того, надписи к рисункам вводятся посредством так называемых выносок различных форм.

Масштабирующие инструменты в растровых графических редакторах дают возможность увеличивать или уменьшать масштаб представления объекта на экране, не влияя при этом на его реальные размеры. Обычно такой инструмент называется Лупа. В векторных графических редакторах легко изменять реальные размеры объекта с помощью мыши.

Форматы графических файлов

Форматы графических файлов определяют способ хранения информации в файле (растровый или векторный), а также форму хранения информации (используемый алгоритм сжатия).

Сжатие применяется для растровых графических файлов, так как они имеют обычно достаточно большой объем. Сжатие графических файлов отличается от их архивации с помощью программ архиваторов. тем, что алгоритм сжатия включается в формат графического файла.

Некоторые форматы графических файлов являются универсальными, так как могут быть обработаны большинством графических редакторов. Некоторые программы обработки изображений используют оригинальные форматы, которые распознают только самой создающей программой.

Рассмотрим некоторые форматы графических файлов:

BMP – универсальный формат растровой графики в windows.

GIF – формат растровых графических файлов для различных ОС. Используется для размещения графических изображений в Интернете.

JPEG - формат растровых графических файлов, который использует эффективных алгоритм сжатия (с потерями). Используется для размещения графических изображений в Интернете.

WMF – универсальный формат векторных графических файлов для windows-приложений.

CDR – оригинальный формат векторных графических файлов, используется в системе обработки изображений CorelDraw.

21. Технология хранения, поиска и сортировки данных (базы данных, информационные системы). Табличные, иерархические и сетевые базы данных.

Любой из нас, начиная с раннего детства, многократно сталкивался с "базами данных". Это - всевозможные справочники, энциклопедии ... Записная книжка - это тоже "база данных", которая есть у каждого из нас.

Базы данных представляют собой информационные модели, содержащие данные об объектах и их свойствах. Базы данных хранят информацию о группах объектов с одинаковыми свойствами. Информация в базах данных хранится в упорядоченном виде (например, в записной книжке все записи упорядочены по алфавиту, в библиотечном каталоге - либо по алфавиту, либо по области знания).

База данных — это информационная модель, позволяющая упорядоченно хранить данные о группе объектов, обладающих одинаковым набором свойств.

Существует несколько различных типов баз данных: табличные, иерархические и сетевые.

Табличные базы данных.

Табличная база данных содержит перечень объектов одного типа, т. е. объектов с одинаковым набором свойств. Такую базу данных удобно представлять в виде двумерной таблицы.

Рассмотрим, например, базу данных «Компьютер» (табл.), представляющую собой перечень объектов (компьютеров), каждый из которых имеет имя (название). В качестве характеристик (свойств) могут выступать тип процессора и объем оперативной памяти.

Таблица1 : таблица				
	№ п/п	Название	Тип процессор	Оперативная п
	1	Compaq	Celeron	64
	2	Dell	Pentium III	128
	3	IBM	Pentium IV	256
*	0			

Столбцы такой таблицы называют полями; каждое поле характеризуется своим именем (названием соответствующего свойства) и типом данных, отражающих значения данного свойства. Поля Название и Тип процессора — текстовые, а Оперативная память — числовое. При этом каждое поле обладает определенным набором свойств (размер, формат и др.). Так, для поля Оперативная память задан формат данных целое число.

Поле базы данных — это столбец таблицы, включающий в себя значения определенного свойства.

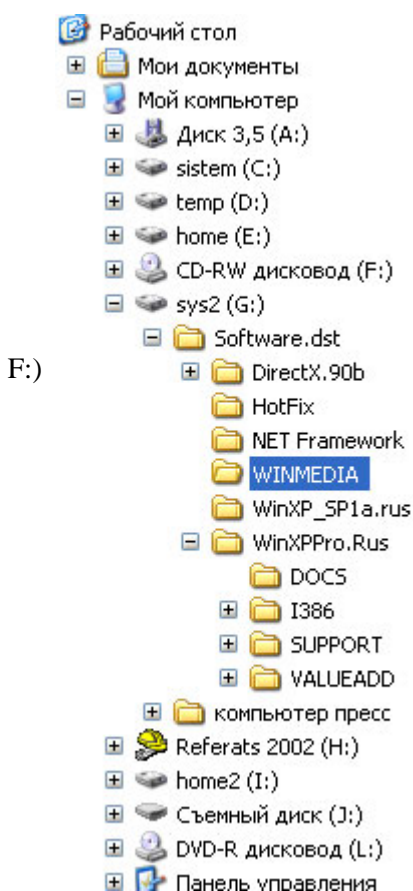
Строки таблицы являются записями об объекте; эти записи разбиты на поля столбцами таблицы. Запись базы данных — это строка таблицы, которая содержит набор значений различных свойств объекта.

В каждой таблице должно быть, по крайней мере, одно ключевое поле, содержимое которого уникально для любой записи в этой таблице. Значения ключевого поля однозначно определяют каждую запись в таблице.

Иерархические базы данных.

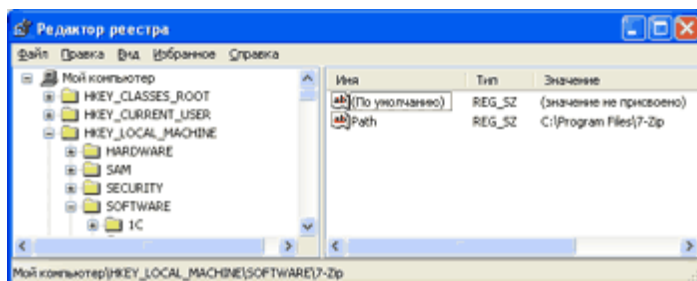
Иерархические базы данных графически могут быть представлены как дерево, состоящее из объектов различных уровней. Верхний уровень занимает один объект, второй — объекты второго уровня и т. д.

Между объектами существуют связи, каждый объект может включать в себя несколько объектов более низкого уровня. Такие объекты находятся в отношении предка (объект более близкий к корню) к потомку (объект более низкого уровня), при этом возможно, чтобы объект-предок не имел потомков или имел их несколько, тогда как у объекта-потомка обязательно только один предок. Объекты, имеющие общего предка, называются близнецами.



Иерархической базой данных является Каталог папок Windows, с которым можно работать, запустив Проводник. Верхний уровень занимает папка Рабочий стол (На втором уровне находятся папки Мой компьютер, Мои документы. Сетевое окружение и Корзина, которые представляют собой потомков папки Рабочий стол, будучи между собой близнецами. В свою очередь, папка Мой компьютер — предок по отношению к папкам третьего уровня, папкам дисков (Диск 3,5(A:), C:, D:, E:, и системным папкам (Принтеры, Панель управления и др.).

Иерархической базой данных является реестр Windows, в котором хранится вся информация, необходимая для нормального функционирования компьютерной системы (данные о конфигурации компьютера и установленных драйверах, сведения об установленных программах, настройки графического интерфейса ...).



Содержание реестра автоматически обновляется при установке нового оборудования, инсталляции программ и т.д. Для просмотра и редактирования реестра Windows в ручном режиме можно использовать специальную программу regedit.exe. Однако редактирование реестра нужно проводить крайне осторожно при условии понимания выполняемых действий. Неквалифицированное редактирование реестра может привести компьютер в неработоспособное состояние.

Еще одним примером иерархической базы данных является база данных Доменная система имен подключенных к Интернету компьютеров. На верхнем уровне находится табличная база данных, содержащая перечень доменов верхнего уровня (всего 264). На втором уровне - табличные базы данных, содержащие перечень доменов второго уровня для каждого домена первого уровня. На третьем уровне могут находиться табличные базы, содержащие перечень доменов третьего уровня для каждого домена второго уровня, и таблицы, содержащие IP-адреса компьютеров, находящихся в домене второго уровня.

База данных Доменная система имен должна содержать записи обо всех компьютерах, подключенных к Интернету (более 150 мил. записей). Размещение такой огромной базы на одном компьютере сделало бы поиск информации очень медленным и неэффективным. Решение этой проблемы было найдено путем размещения отдельных составных частей базы на различных DNS-серверах. Таким образом, иерархическая база данных Доменная система имен является распределенной базой данных.

Сетевые базы данных.

Сетевая база данных образуется обобщением иерархической за счет допущения объектов, имеющих более одного предка, т. е. каждый элемент вышестоящего уровня может быть связан одновременно с любыми элементами следующего уровня. Вообще, на связи между объектами в сетевых моделях не накладывается никаких ограничений.

Сетевой базой данных фактически является Всемирная паутина глобальной компьютерной сети Интернет. Гиперссылки связывают между собой сотни миллионов документов в единую распределенную сетевую базу данных.

Системы управления базами данных (СУБД).

Для создания баз данных, а также выполнения операции поиска и сортировки данных предназначены специальные программы — системы управления базами данных (СУБД).

Таким образом, необходимо различать собственно базы данных (БД) — упорядоченные наборы данных, и системы управления базами данных (СУБД) — программы, управляющие хранением и обработкой данных. Например, приложение Access, входящее в офисный пакет программ Microsoft Office, является СУБД, позволяющей пользователю создавать и обрабатывать табличные базы данных.

22. Локальные и глобальные компьютерные сети. Адресация в сетях.

Компьютерная сеть – это совокупность компьютеров и различных устройств, обеспечивающих информационный обмен между компьютерами в сети без использования каких-либо промежуточных носителей информации.

Создание компьютерных сетей вызвано практической потребностью пользователей удаленных друг от друга компьютеров в одной и той же информации. Сети предоставляют пользователям возможность не только быстрого обмена информацией, но и совместной работы на принтерах и других периферийных устройствах, и даже одновременной обработки документов.

Все многообразие компьютерных сетей можно классифицировать по группе признаков:

- Территориальная распространенность;
- Ведомственная принадлежность;
- Скорость передачи информации;
- Тип среды передачи;

По территориальной распространенности сети могут быть локальными, глобальными, и региональными.

По принадлежности различают ведомственные и государственные сети. Ведомственные принадлежат одной организации и располагаются на ее территории.

По скорости передачи информации компьютерные сети делятся на низко-, средне- и высокоскоростные.

По типу среды передачи разделяются на сети коаксиальные, на витой паре, оптоволоконные, с передачей информации по радиоканалам, в инфракрасном диапазоне.

Локальные компьютерные сети.

Локальная сеть объединяет компьютеры, установленные в одном помещении (например, школьный компьютерный класс, состоящий из 8—12 компьютеров) или в одном здании (например, в здании школы могут быть объединены в локальную сеть несколько десятков компьютеров, установленных в различных предметных кабинетах).

В небольших локальных сетях все компьютеры обычно равноправны, т. е. пользователи самостоятельно решают, какие ресурсы своего компьютера (диски, каталоги, файлы) сделать общедоступными по сети. Такие сети называются одноранговыми.


Если к локальной сети подключено более десяти компьютеров, то одноранговая сеть может оказаться недостаточно производительной. Для увеличения производительности, а также в целях обеспечения большей надежности при хранении информации в сети некоторые компьютеры специально выделяются для хранения файлов или программ-приложений. Такие компьютеры называются серверами, а локальная сеть — сетью на основе серверов. Каждый компьютер, подключенный к локальной сети, должен иметь специальную плату (сетевой адаптер). Между собой компьютеры (сетевые адаптеры) соединяются с помощью кабелей.

Топология сети.


Общая схема соединения компьютеров в локальные сети называется топологией сети. Топологии сети могут быть различными.

Сети Ethernet могут иметь топологию «шина» и «звезда». В первом случае все компьютеры подключены к одному общему кабелю (шине), во втором - имеется специальное центральное устройство (хаб), от которого идут «лучи» к каждому компьютеру, т.е. каждый компьютер подключен к своему кабелю.


Структура типа «шина» проще и экономичнее, так как для нее не требуется дополнительное устройство и расходуется меньше кабеля. Но она очень чувствительна к неисправностям кабельной системы. Если кабель поврежден хотя бы в одном месте, то возникают проблемы для всей сети. Место неисправности трудно обнаружить.



В этом смысле «звезда» более устойчива. Поврежденный кабель – проблема для одного конкретного компьютера, на работе сети в целом это не сказывается. Не требуется усилий по локализации неисправности.



В сети, имеющей структуру типа «кольцо» информация передается между станциями по кольцу с переключением в каждом сетевом контроллере. Переключение производится через буферные накопители, выполненные на базе оперативных запоминающих устройств, поэтому при выходе из строя одного сетевого



контроллера может нарушиться работа всего кольца. Достоинство кольцевой структуры – простота реализации устройств, а недостаток – низкая надежность.

Региональные компьютерные сети.

Локальные сети не позволяют обеспечить совместный доступ к информации пользователям, находящимся, например, в различных частях города. На помощь приходят региональные сети, объединяющие компьютеры в пределах одного региона (города, страны, континента).

Корпоративные компьютерные сети.

Многие организации, заинтересованные в защите информации от несанкционированного доступа (например, военные, банковские и пр.), создают собственные, так называемые корпоративные сети. Корпоративная сеть может объединять тысячи и десятки тысяч компьютеров, размещенных в различных странах и городах (в качестве примера можно привести сеть корпорации Microsoft, MSN).

Глобальная компьютерная сеть Интернет.

В 1969 году в США была создана компьютерная сеть ARPAnet, объединяющая компьютерные центры министерства обороны и ряда академических организаций. Эта сеть была предназначена для узкой цели: главным образом для изучения того, как поддерживать связь в случае ядерного нападения и для помощи исследователям в обмене информацией. По мере роста этой сети создавались и развивались многие другие сети. Еще до наступления эры персональных компьютеров создатели ARPAnet приступили к разработке программы Internetworking Project ("Проект объединения сетей"). Успех этого проекта привел к следующим результатам. Во-первых, была создана крупнейшая в США сеть internet (со строчной буквы i). Во-вторых, были опробованы различные варианты взаимодействия этой сети с рядом других сетей США. Это создало предпосылки для успешной интеграции многих сетей в единую мировую сеть. Такую "сеть сетей" теперь всюду называют Internet (в отечественных публикациях широко применяется и русскоязычное написание - Интернет).

В настоящее время на десятках миллионов компьютеров, подключенных к Интернету, хранится громадный объем информации (сотни миллионов файлов, документов и т. д.) и сотни миллионов людей пользуются информационными услугами глобальной сети.

Интернет — это глобальная компьютерная сеть, объединяющая многие локальные, региональные и корпоративные сети и включающая в себя десятки миллионов компьютеров.

В каждой локальной или корпоративной сети обычно имеется, по крайней мере, один компьютер, который имеет постоянное подключение к Интернету с помощью линии связи с высокой пропускной способностью (сервер Интернета).

Надежность функционирования глобальной сети обеспечивается избыточностью линий связи: как правило, серверы имеют более двух линий связи, соединяющих их с Интернетом.

Основу, «каркас» Интернета составляют более ста миллионов серверов, постоянно подключенных к сети.

К серверам Интернета могут подключаться с помощью локальных сетей или коммутируемых телефонных линий сотни миллионов пользователей сети.

Адресация в Интернет

Для того, чтобы связаться с некоторым компьютером в сети Интернет, Вам надо знать его уникальный Интернет - адрес. Существуют два равноценных формата адресов, которые различаются лишь по своей форме: IP - адрес и DNS - адрес.

IP - адрес

IP - адрес состоит из четырех блоков цифр, разделенных точками. Он может иметь такой вид:
84.42.63.1

Каждый блок может содержать число от 0 до 255. Благодаря такой организации можно получить свыше четырех миллиардов возможных адресов. Но так как некоторые адреса зарезервированы для специальных целей, а блоки конфигурируются в зависимости от типа сети, то фактическое количество возможных адресов немного меньше. И тем ни менее, его более чем достаточно для будущего расширения Интернет.

С понятием IP - адреса тесно связано понятие "хост". Под хостом понимается любое устройство, использующее протокол TCP/IP для общения с другим оборудованием. Это может быть не только компьютер, но и маршрутизатор, концентратор и т.п. Все эти устройства, подключенные в сеть, обязаны иметь свой уникальный IP - адрес.

DNS - адрес

IP - адрес имеет числовой вид, так как его используют в своей работе компьютеры. Но он весьма сложен для запоминания, поэтому была разработана доменная система имен: DNS. DNS - адрес включает более удобные для пользователя буквенные сокращения, которые также разделяются точками на отдельные информационные блоки (домены). Например: www.klyaksa.net

Если Вы вводите DNS - адрес, то он сначала направляется в так называемый сервер имен, который преобразует его в 32 - битный IP - адрес для машинного считывания.

Доменные имена

DNS - адрес обычно имеет три составляющие (хотя их может быть сколько угодно).

Доменная система имен имеет иерархическую структуру: домены верхнего уровня - домены второго уровня и так далее. Домены верхнего уровня бывают двух типов: географические (двухбуквенные - каждой стране свой код) и административные (трехбуквенные).

России принадлежит географический домен **ru**.

Портал Клякс@.net зарегистрировал домен второго уровня klyaksa в административном домене верхнего уровня net.

Имена компьютеров, которые являются серверами Интернета, включают в себя полное доменное имя и собственно имя компьютера. Так полный адрес портала Клякс@.net имеет вид www.klyaksa.net

gov - правительственное учреждение или организация
mil - военное учреждение
com - коммерческая организация
net - сетевая организация
org - организация, которая не относится не к одной из выше перечисленных

Среди часто используемых доменов - идентификаторов стран можно выделить следующие:

at - Австрия
au - Австралия
ca - Канада
ch - Швейцария
de - Германия
dk - Дания
es - Испания
fi - Финляндия
fr - Франция
it - Италия
jp - Япония
nl - Нидерланды
no - Норвегия
nz - Новая Зеландия
ru - Россия
se - Швеция
uk - Украина
za - Южная Африка

Адрес E-mail

С помощью IP - адреса или DNS - адреса в Интернет можно обратиться к любому нужному компьютеру. Если же Вы захотите послать сообщение по электронной почте, то указания только этих адресов будет недостаточно, поскольку сообщение должно попасть не только в нужный компьютер, но и к определенному пользователю системы.

Для доставки и прима сообщений электронной почты предназначен специальный протокол SMTP (Simple Mail Transport Protocol). Компьютер, через который в Интернет осуществляется передача сообщений электронной почты, называют SMTP - сервером. По электронной почте сообщения доставляются до указанного в адресе компьютера, который и отвечает за дальнейшую доставку. Поэтому такие данные, как имя пользователя и имя соответствующего SMTP - сервера разделяют знаком "@". Этот знак называется "at коммерческое" (на жаргоне - собачка, собака). Таким образом, Вы адресуете свое сообщение конкретному пользователю конкретного компьютера.

Например:
ivanov@klyaksa.net Здесь ivanov - пользователь, которому предназначено послание, а klyaksa.net - SMTP - сервер, на котором находится его электронный почтовый ящик (mailbox). В почтовом ящике хранятся сообщения, пришедшие по конкретному адресу.

URL

URL (Uniform Resource Locator, унифицированный определитель ресурсов) - это адрес некоторой информации в Интернет. Он имеет следующий формат:

тип ресурса://адрес узла/прочая информация

Наиболее распространенными считаются следующие типы ресурсов:

ftp:// ftp - сервер

gopher:// меню gopher

http:// адрес в WWW

mailto:// адрес электронной почты

news:// группа новостей UseNet

telnet:// компьютер, в котором можно зарегистрироваться, используя telnet

Ресурсная часть URL всегда заканчивается двоеточием и двумя или тремя наклонными чертами. Далее следует конкретный адрес узла, который Вы хотите посетить. За ним в качестве

ограничителя может стоять наклонная черта. В принципе, этого вполне достаточно. Но если Вы хотите просмотреть конкретный документ на данном узле и знаете точно его место расположения, то можете включить его адрес в URL. Ниже приведены несколько URL и расшифровка их значений:

<http://www.klyaksa.net/index.php> главная страница информационно-образовательного портала Клякс@.net

<ftp://ftp.microsoft.com/dirmap.txt> файл с именем dirmap.txt на ftp - сервере компании Microsoft

Итак, в Интернет возможны следующие виды адресов:

Адрес	формат
IP	12.105.58.9
DNS	компьютер.сеть. домен
E - mail	пользователь@е mail-сервер
URL	тип ресурса://DNS - адрес

23. Глобальная сеть Интернет и ее информационные сервисы (электронная почта, Всемирная паутина, файловые архивы и пр.). Поиск

Интернет - это мировая компьютерная сеть. В ней множество компьютеров по всему свету соединены проводами, телефонными линиями, радио и спутниковой связью. Со своего персонального компьютера Вы можете связаться с любой точкой земного шара и получить доступ к информации, которая содержится на любом компьютере, подключенном в сеть Интернет. А так как количество пользователей всемирной сетью растет, то и растут Ваши возможности в ней. Вы можете вступать в дискуссии по тем или иным темам, посещать виртуальные выставки, вести электронный бизнес, общаться с помощью почты и многое, многое другое. Достоинствами работы в сети Интернет являются быстрота, дешевизна, многоаспектность и перспективность.

Прародителем Интернет была сеть ARPANet. Она возникла в 1969 году, в Америке, для того, чтобы облегчить сотрудничество между организациями оборонной промышленности, разбросанными по разным штатам. Сначала она соединяла компьютерные системы одного типа, но по мере развития возникла необходимость в обмене данными между "разнородными" сетями. Так возник проект Interneting Project. В результате был создан стандарт передачи данных - протокол TCP/IP.

Протоколом передачи данных называется соглашение, устанавливающее, каким образом должна осуществляться передача данных из компьютера в компьютер и как можно распознавать и устранять ошибки, которые могут при этом возникать. И для того, чтобы осуществилась идея неограниченной коммуникации между компьютерами Интернет, используется один и тот же протокол TCP/IP. Он состоит из набора протоколов, каждый из которых выполняет различные задачи.

TCP, UDP транспортные протоколы, управляющие передачей данных между машинами

IP, ICMP, RIP протоколы маршрутизации. Они обрабатывают адресацию данных, обеспечивают фактическую передачу данных

DNS, ARP протоколы поддержки сетевого адреса обеспечивают идентификацию машины с уникальным номером и именем

FTP, TELNET протоколы прикладных сервисов. Это программы, которые пользователь использует для получения доступа к различным услугам

и др.

Протоколы семейства TCP/IP реализуют всевозможные сервисы (услуги) Интернет.

WWW

Популярнейший из них - World Wide Web (сокращенно WWW или Web), его еще называют Всемирной паутиной. Представление информации в WWW основано на возможностях гипертекстовых ссылок. Гипертекст - это текст, в котором содержатся ссылки на другие документы. Это дает возможность при просмотре некоторого документа легко и быстро переходить к другой связанной с ним по смыслу информации, которая может быть текстом, изображением, звуковым файлом или иметь любой другой вид, принятый в WWW. При этом связанные ссылками документы могут быть разбросаны по всему земному шару.

Многочисленные пересекающиеся связи между документами WWW компьютерной паутиной охватывают планету - отсюда и название. Таким образом, пропадает зависимость от местонахождения конкретного документа.

Gopher-система

Эта система является предшественником WWW и сейчас утрачивает свое значение, хотя пока и поддерживается в Интернет. Это информационные серверы, на которых содержатся документы академической направленности и большие текстовые файлы. Просмотр информации на Gopher-сервере организуется с помощью древовидного меню, аналогичного меню в приложениях Windows или аналогично дереву каталогов (папок) файловой системы. Меню верхнего уровня состоит из перечня крупных тем, например, экономика, культура, медицина и др. Меню следующих уровней детализируют выбранный элемент меню предыдущего уровня. Конечным пунктом движения вниз по дереву (листом дерева) служит документ аналогично тому, как конечным элементом в дереве каталогов является файл.

Электронная почта

Следующий вид сервиса Интернет - электронная почта, или E - mail. Она предназначена для передачи в сети файлов любого типа. Одни из главных ее преимуществ - дешевизна и быстрота.

Электронная почта является исторически первой информационной услугой компьютерных сетей и не требует обязательного наличия высокоскоростных и качественных линий связи.

Любой пользователь Интернета может получить свой «почтовый ящик» на одном из почтовых серверов Интернета (обычно на почтовом сервере провайдера), в котором будут храниться передаваемые и получаемые электронные письма.

У электронной почты есть преимущества перед телефонной связью. Телефонный этикет очень строг. Есть множество случаев, когда нельзя позвонить человеку по соображениям этикета. У электронной почты требования намного мягче. По электронной почте можно обратиться к малознакомому человеку или очень занятому человеку. Если он сможет, то ответит.

Чтобы электронное письмо дошло до адресата, оно, кроме текста послания, обязательно должно содержать электронный адрес получателя письма.

Адрес электронной почты записывается по определенной форме и состоит из двух частей: имя_пользователя@имя_сервера

Имя_пользователя имеет произвольный характер и задается самим пользователем; имя_сервера жестко связано с выбором пользователем сервера, на котором он разместил свой почтовый ящик.

Пример, ivanov@kyaksa.net

В нашем классе имя пользователя – это имя компьютера, например, pc01, pc02 и т.д. имя сервера: server, поэтому электронный адрес компьютера в локальной сети класса: pc01@server

Чтобы отправить электронное письмо, отправитель должен подключиться к Интернету и передать на свой почтовый сервер сообщение. Почтовый сервер сразу же отправит это письмо через систему почтовых серверов Интернет на почтовый сервер получателя, и оно попадет в его почтовый ящик.



Однако получатель получит письмо только после того, как соединится с Интернетом и «скачает» почту из своего почтового ящика на собственный локальный компьютер.

Телеконференции UseNet

Телеконференции UseNet представляют собой электронные форумы. Пользователи Интернет посылают туда свои сообщения, в которых высказываются по определенной теме. Сообщения поступают в специальные дискуссионные группы - телеконференции, при этом каждое мнение становится доступным для всех участников конкретной группы. Уже сегодня UseNet имеет более 20 000 телеконференций, посвященных различным темам: компьютерам, рецептам, вопросам генной инженерии и многому другому.

Протокол передачи файлов FTP

Протокол передачи файлов FTP используется для переписывания файлов с дистрибутивными копиями программ с удаленных серверов на Ваш компьютер. В зависимости от своих прав (обычный пользователь или др.) Вы можете производить те или иные действия по отношению к удаленному серверу (в большинстве случаев это копия имеющейся на нем информации).

Telnet

Программа Telnet была разработана для обеспечения дистанционного доступа к удаленному компьютеру в Интернет. При этом компьютер пользователя выступает в качестве терминала, подключенного к большому компьютеру. В отличие от компьютеров, терминалы не обладают собственными вычислительными возможностями. Они только обеспечивают доступ к какому-то компьютеру благодаря имеющимся у них монитору и клавиатуре. В качестве примера можно привести системы в аэропортах, на вокзалах, где Вы можете получить информацию о билетах, рейсах и т.п.

Поиск информации в Интернет

Пользуясь гипертекстовыми ссылками, можно бесконечно долго путешествовать в информационном пространстве Сети, переходя от одной web-страницы к другой, но если учесть, что в мире созданы многие миллионы web-страниц, то найти на них нужную информацию таким способом вряд ли удастся.

На помощь приходят специальные поисковые серверы (их еще называют поисковыми машинами). Поисковых серверов не очень много, и их адреса хорошо известны всем, кто работает в Интернете. В ответ на запрос, где найти нужную информацию, поисковый сервер возвращает список гиперссылок, ведущих web-страницам, на которых нужная информация имеется или упоминается. Обширность списка может быть любой, в зависимости от содержания запроса.

По принципу действия поисковые серверы делятся на два типа: поисковые каталоги и поисковые индексы.

Поисковые каталоги служат для тематического поиска. Информация на этих серверах структурирована по темам и подтемам. Имея намерение осветить какую-то узкую тему, нетрудно найти список web-страниц, ей посвященных. Поисковые индексы работают как алфавитные указатели. Клиент задает слово или группу слов, характеризующих его область поиска, например: Бразилия футбол чемпионат, — и получает список ссылок на web-страницы, содержащие указанные термины.

Интересную возможность сбора информации предоставляют и специальные классификационные серверы. На них постоянно представлены тысячи ссылок на ресурсы Интернета, классифицированные не только по теме, но и по популярности. При каждом использовании ссылки срабатывает счетчик обращений, в результате чего новые посетители сервера видят, какие адреса используются чаще и могут выбрать для просмотра наиболее популярные ресурсы по интересующей их теме. Это удобно для проведения первичного сбора информации.

24. Информационное общество

Человеческое общество по мере своего развития прошло этапы овладения веществом, затем энергией и, наконец, информацией.

В первобытно-общинном, рабовладельческом и феодальном обществах (в основе существования которых лежало ремесло) деятельность общества в целом и каждого человека в отдельности была направлена, в первую очередь, на овладение веществом. На заре цивилизации (десятки тысяч лет до н. э.) люди научились изготавливать простые орудия труда и охоты (каменный топор, стрелы и так далее), в античности появились первые механизмы (рычаг и др.) и средства передвижения (колесницы, корабли), в средние века были изобретены первые сложные орудия труда и механизмы (ткацкий станок, часы). Овладение энергией находилось в этот период на начальной ступени, в качестве источников энергии использовались Солнце, вода, огонь, ветер и мускульная сила человека.

С самого начала человеческой истории возникла потребность передачи и хранения информации. Для передачи информации сначала использовался язык жестов, а затем человеческая речь. Для хранения информации стали использоваться наскальные рисунки, а в IV тысячелетии до нашей эры появилась письменность и первые носители информации (шумерские глиняные таблички и египетские папирусы). История создания устройств для обработки числовой информации начинается также еще с древности — с абака (счетной доски, являющейся прообразом счетов).

Индустриальное общество.

Начиная примерно с XVII века в процессе становления машинного производства на Первый план выходит проблема овладения энергией (машины и станки необходимо было приводить в движение). Сначала совершенствовались способы овладения энергией ветра и воды (ветряные мельницы и водяные колеса), а затем человечество овладело тепловой энергией (в середине XVIII века была изобретена, паровая машина, а в конце XIX века — двигатель внутреннего сгорания).

В конце XIX века началось овладение электрической энергией, были изобретены электрогенератор и электродвигатель. И наконец, в середине XX века человечество овладело атомной энергией, в 1954 году в СССР была пущена в эксплуатацию первая атомная электростанция.

Овладение энергией позволило перейти к массовому машинному производству потребительских товаров, было создано индустриальное общество. Основными показателями развитости индустриального общества являлись количественные показатели, то есть сколько было добыто угля и нефти, сколько произведено станков и так далее.

В этот период происходили также существенные изменения в способах хранения и передачи информации. В середине XV века было изобретено книгопечатание, что позволило сделать информацию доступной для гораздо большего количества людей. С конца XIX века для передачи информации на дальние расстояния по проводам стали широко использоваться телеграф и телефон, а в XX веке — электромагнитные волны (радио, телевидение).

Информационное общество.

Первой попыткой автоматизированной обработки информации стало создание Чарльзом Бэббиджем в середине XIX века механической цифровой аналитической машины. Однако лишь с середины XX века, с момента появления электронных устройств обработки и хранения информации (ЭВМ, а затем персонального компьютера), начался постепенный переход от индустриального общества к информационному.

В информационном обществе главным ресурсом является информация, именно на основе владения информацией о самых различных процессах и явлениях можно эффективно и оптимально строить любую деятельность.

Важно не только произвести большое количество продукции, но произвести нужную продукцию в определенное время, с определенными затратами и так далее. Поэтому в информационном обществе повышается не только качество потребления, но и качество производства; человек, использующий информационные технологии, имеет лучшие условия труда, труд становится творческим, интеллектуальным и так далее.

В настоящее время развитые страны мира (США, Япония, страны Западной Европы) фактически уже вступили в информационное общество, другие же, в том числе и Россия, находятся на ближних подступах к нему.

В качестве критериев развитости информационного общества можно выбрать три: наличие компьютеров, уровень развития компьютерных сетей и количество населения, занятого в информационной сфере, а также использующего информационные и коммуникационные технологии в своей повседневной деятельности.

Производство компьютеров.

Первые электронно-вычислительные машины (ЭВМ), которые могли автоматически по заданной программе обрабатывать большие объемы информации, были созданы в 1946 году в США (ЭНИАК) и в 1950 году в СССР (МЭСМ). В 40-60-х годах производство ЭВМ измерялась единицами, десятками и, в лучшем случае, сотнями штук. ЭВМ были очень дорогими и очень большими (занимали громадные залы) и поэтому оставались недоступными для массового потребителя.

Массовое производство сравнительно недорогих персональных компьютеров началось с середины 70-х годов XX века с компьютера Apple II (с этого компьютера начала свое существование фирма Apple). Количество производимых персональных компьютеров начало составлять десятки тысяч в год, что по тем временам было колоссальным достижением.

В начале 80-х годов приступила к массовому производству персональных компьютеров корпорация IBM (компьютеры так и назывались IBM Personal Computer — IBM PC). Достаточно скоро IBM-совместимые компьютеры стали выпускать многие фирмы, и их производство достигло сотен тысяч в год. Ежегодное производство персональных компьютеров постоянно росло и в 2000 году превысило 150 миллионов.

Персональный компьютер постоянно совершенствовался, его производительность возросла на три порядка, при этом, что очень важно, цена практически не изменилась. Персональный компьютер стал доступен массовому потребителю, и теперь в развитых странах мира компьютер имеется на большинстве рабочих мест и в большинстве семей.

Компьютерные сети.

В настоящее время существенной тенденцией в информатизации общества является переход от использования компьютеров в автономном режиме к использованию их в информационных сетях.

Информационные сети создают реальную возможность быстрого и удобного доступа пользователя ко всей информации, накопленной человечеством за всю свою историю. Электронная почта и телеконференции, поиск информации во Всемирной паутине и в файловых архивах, интерактивное общение, прослушивание радиостанций и просмотр телевизионных программ, покупки в Интернет-магазинах стали повседневной практикой многих пользователей компьютеров в развитых странах.

Развитие глобальных компьютерных сетей началось в 80-е годы. В 1981 году в сети Интернет насчитывалось лишь 213 компьютеров, к концу 80-х годов количество подключенных к сети компьютеров возросло до 150 тысяч, однако наиболее быстрый экспоненциальный рост их количества происходил в 90-е годы.

По количеству имеющихся серверов Интернета можно судить о степени информатизации отдельных стран. Наибольшее количество серверов зарегистрировано в доменах административного типа, которые находятся в основном в США (около 104 миллионов серверов), на втором месте, с большим отставанием, Япония (7,1 миллионов серверов), Россия занимает в этом списке 24-е место (около 400 тысяч серверов).

Развитие глобальной компьютерной сети требует наличия каналов связи с высокой пропускной способностью. Основой глобальной компьютерной сети Интернет являются

магистральные высокоскоростные линии связи, по которым передается информация между региональными сетями. В настоящее время наиболее мощные региональные сети функционируют в Северной Америке, Европе, Японии и Австралии. Они соединены между собой многочисленными оптоволоконными линиями связи с пропускной способностью до 20 Гбит/с и выше.

Внутри региональных сетей информация передается также преимущественно по оптоволоконным каналам с различной пропускной способностью (от 1 до 155 Мбит/с). В региональных сетях часто используются также выделенные линии (медные), а иногда (в пределах прямой видимости) и радиоканалы, пропускная способность которых может достигать 2 Мбит/с.

Для подключения отдаленных регионов наиболее экономически выгодным является подключение по спутниковым каналам, пропускная способность которых может достигать десятков мегабитов в секунду.

Однако для большинства индивидуальных пользователей (их сейчас в мире около 1 миллиарда) приемлемым по цене является доступ в Интернет только по коммутируемым телефонным каналам со скоростью до 56 Кбит/с. В России, по разным оценкам, таких пользователей от 3 до 5 миллионов.

Население, занятое в информационной сфере.

По данным ООН, в 90-е годы количество работников, занятых в информационной сфере (для которых обработка информации является основной производственной функцией), возросло примерно на 25%, тогда как количество занятых в сельском хозяйстве и промышленности сократилось соответственно на 10 и 15%.

Компьютеры и информационные технологии интенсивно проникают и в сферу материального производства. Инженер, фермер, специалисты других традиционных профессий все чаще имеют на своем рабочем месте компьютер и используют информационные и коммуникационные технологии в своей профессиональной деятельности.

С развитием коммуникационных технологий и мобильной связи все большее количество людей осуществляют свою производственную деятельность дистанционно, то есть работая дома, а не в офисе (в США более 10 миллионов человек). Все большее распространение получает дистанционное образование и поиск работы через Интернет. В 2000 году оборот мирового рынка информационных и коммуникационных технологий составил около 1 триллиона долларов. При этом на закупку аппаратных средств было потрачено менее половины этой суммы, большая часть была вложена в разработку программного обеспечения, проектирование компьютерных сетей и так далее.

Информационное общество — это общество, в котором большая часть населения занята получением, переработкой, передачей и хранением информации.

Курс информатики и информационных технологий играет особую роль в эпоху перехода от индустриального общества к информационному, так как готовит выпускников школы к жизни и деятельности в информационном обществе.

Информационная культура

Количество информации в современном обществе стремительно нарастает, человек оказывается погруженным в море информации. Для того чтобы в этом море «не утонуть», необходимо обладать информационной культурой, то есть знаниями и умениями в области информационных и коммуникационных технологий, а также быть знакомым с юридическими и этическими нормами в этой сфере.

Процесс информатизации общества меняет традиционные взгляды на перечень умений и навыков, необходимых для социальной адаптации. Возьмем традиционный навык письма. На заре цивилизации (Шумер, Египет), в античном мире (Эллада, Римская империя и др.) и в средние века (до изобретения книгопечатания) навык каллиграфического письма был залогом успешного продвижения по социальной лестнице. В индустриальном обществе (до изобретения персональных компьютеров) навыки письма ручкой также были необходимы для любого члена общества.

В настоящее время, на пороге информационного общества, социальная значимость навыка письма ручкой снижается и, наоборот, социальная значимость навыков ввода информации с помощью клавиатуры и работы с графическим интерфейсом приложений с помощью мыши возрастает.

Создание и редактирование документов с помощью компьютера, то есть овладение офисными информационными технологиями, становится в информационном обществе социально необходимым умением — достаточно просмотреть объявления о приеме на работу.

Умение работать с мультимедиа-документами, создавать компьютерные презентации становится важным в информационном обществе.

В современном информационном обществе вряд ли необходимы навыки традиционного черчения на ватмане. Вместо этого полезно получить первоначальное представление о назначении и возможностях компьютерных систем автоматизированного проектирования (САПР). Такие системы позволят вам быстро рассмотреть различные варианты планировки интерьера дома или квартиры, создать чертеж или схему.

Использование электронных таблиц сделает более простыми и наглядными процессы исследования и построения графиков функций в процессе изучения математики, планирования и ведения домашнего бюджета, построения и исследования моделей различных объектов и процессов.

Необходимость упорядочить информацию, например, о людях, с которыми вы контактируете, требует использования записной книжки. Однако часто удобнее использовать для хранения такой информации компьютерную базу данных «Записная книжка».

При поиске информации в современной библиотеке или в Интернете необходимо иметь навыки поиска информации в базах данных. В информационном обществе очень полезным является умение создавать базы данных, а также вести в них поиск данных.

Квалифицированный пользователь компьютера может на основе использования средств визуального объектно-ориентированного программирования создавать необходимые ему специализированные приложения. Например, можно создать приложение, которое автоматизирует заполнение многочисленных квитанций оплаты за квартиру, электроэнергию, газ и др.

Современному человеку необходимо овладеть коммуникативной культурой, то есть умениями создавать и посылать электронные письма, находить нужную информацию во Всемирной паутине или в файловых архивах, участвовать в чатах и так далее. Необходимым условием успешной профессиональной деятельности становится создание и публикация в Интернете Web-сайтов с информацией о деятельности организации или предприятия.

Информационная культура состоит не только в овладении определенным комплексом знаний и умений в области информационных и коммуникационных технологий, но предполагает знание и соблюдение юридических и этических норм и правил. Законы запрещают использование пиратского компьютерного обеспечения и пропаганду насилия, наркотиков и порнографии в Интернете. Общение с помощью электронной почты или в чатах, участие в телеконференциях предполагают соблюдение определенных правил: отвечать на письма и не рассылать знакомым и

незнакомым людям многочисленные рекламные сообщения (спам), не отклоняться от темы обсуждения в телеконференциях и чатах и так далее.

25. Этические и правовые аспекты информационной деятельности. Правовая охрана программ и данных. Защита информации.

Защита информации. Лицензионные, условно бесплатные и бесплатные программы.

Программы по их юридическому статусу можно разделить на три большие группы: лицензионные, условно бесплатные (shareware) и свободно распространяемые программы (freeware).

Дистрибутивы лицензионных программ (дискеты или диски CD-ROM, с которых производится установка программ на компьютеры пользователей) распространяются разработчиками на основании договоров с пользователями на платной основе, проще говоря, лицензионные программы продаются. Довольно часто разработчики предоставляют существенные скидки при покупке лицензий на использование программы на большом количестве компьютеров или на использование программы в учебных заведениях. В соответствии с лицензионным соглашением разработчики программы гарантируют ее нормальное функционирование в определенной операционной системе и несут за это ответственность.

Некоторые фирмы -- разработчики программного обеспечения предлагают пользователям условно бесплатные программы в целях их рекламы и продвижения на рынок. Пользователю предоставляется версия программы с ограниченным сроком действия (после истечения указанного срока программа перестает работать, если за нее не произведена оплата) или версия программы с ограниченными функциональными возможностями (в случае оплаты пользователю сообщается код, включающий все функции).

Многие производители программного обеспечения и компьютерного оборудования заинтересованы в широком бесплатном распространении программного обеспечения. К таким программным средствам можно отнести следующие:

- новые недоработанные (бета) версии программных продуктов (это позволяет провести их широкое тестирование);
- программные продукты, являющиеся частью принципиально новых технологий (это позволяет завоевать рынок);
- дополнения к ранее выпущенным программам, исправляющие найденные ошибки или расширяющие возможности;
- устаревшие версии программ;
- драйверы к новым устройствам или улучшенные драйверы к уже существующим.

Правовая охрана информации

Правовая охрана программ и баз данных. Правовая охрана программ для ЭВМ и баз данных впервые в полном объеме введена в Российской Федерации Законом РФ «О правовой охране программ для электронных вычислительных машин и баз данных», который вступил в силу в 1992 году.

Предоставляемая настоящим законом правовая охрана распространяется на все виды программ для ЭВМ (в том числе на операционные системы и программные комплексы), которые могут быть выражены на любом языке и в любой форме, включая исходный текст на языке программирования и машинный код. Однако правовая охрана не распространяется на идеи и принципы, лежащие в основе программы для ЭВМ, в том числе на идеи и принципы организации интерфейса и алгоритма.

Для признания и осуществления авторского права на программы для ЭВМ не требуется ее регистрация в какой-либо организации. Авторское право на программы для ЭВМ возникает автоматически при их создании.

Для оповещения о своих правах разработчик программы может, начиная с первого выпуска в свет программы, использовать знак охраны авторского права, состоящий из трех элементов:

- буквы С в окружности или круглых скобках ©;
- наименования (имени) правообладателя;
- года первого выпуска программы в свет.

Например, знак охраны авторских прав на текстовый редактор Word выглядит следующим образом:

© Корпорация Microsoft, 1993-1997.

Автору программы принадлежит исключительное право осуществлять воспроизведение и распространение программы любыми способами, а также модификацию программы.

Организация или пользователь, правомерно владеющий экземпляром программы (купивший лицензию на ее использование), вправе без получения дополнительного разрешения разработчика осуществлять любые действия, связанные с функционированием программы, в том числе ее запись и хранение в памяти ЭВМ. Запись и хранение в памяти ЭВМ допускаются в отношении одной ЭВМ или одного пользователя в сети, если другое не предусмотрено договором с разработчиком.

Необходимо знать и выполнять существующие законы, запрещающие нелегальное копирование и использование лицензионного программного обеспечения. В отношении организаций или пользователей, которые нарушают авторские права, разработчик может потребовать возмещения причиненных убытков и выплаты нарушителем компенсации в определяемой по усмотрению суда сумме от 5000-кратного до 50 000-кратного размера минимальной месячной оплаты труда.

Электронная подпись.

В 2002 году был принят Закон РФ «Об электронно-цифровой подписи», который стал законодательной основой электронного документооборота в России. По этому закону электронная цифровая подпись в электронном документе признается юридически равнозначной подписи в документе на бумажном носителе.

При регистрации электронно-цифровой подписи в специализированных центрах корреспондент получает два ключа: секретный и открытый. Секретный ключ хранится на дискете или смарт-карте и должен быть известен только самому корреспонденту. Открытый ключ должен быть у всех потенциальных получателей документов и обычно рассылается по электронной почте.

Процесс электронного подписания документа состоит в обработке с помощью секретного ключа текста сообщения. Далее зашифрованное сообщение посылается по электронной почте абоненту. Для проверки подлинности сообщения и электронной подписи абонент использует открытый ключ.

Защита информации.

Защита доступа к компьютеру.

Для предотвращения несанкционированного доступа к данным, хранящимся на компьютере, используются пароли. Компьютер разрешает доступ к своим ресурсам только тем

пользователям, которые зарегистрированы и ввели правильный пароль. Каждому конкретному пользователю может быть разрешен доступ только к определенным информационным ресурсам. При этом может производиться регистрация всех попыток несанкционированного доступа.

Защита пользовательских настроек имеется в операционной системе Windows (при загрузке системы пользователь должен ввести свой пароль), однако такая защита легко преодолима, так как пользователь может отказаться от введения пароля. Вход по паролю может быть установлен в программе BIOS Setup, компьютер не начнет загрузку операционной системы, если не введен правильный пароль. Преодолеть такую защиту нелегко, более того, возникнут серьезные проблемы доступа к данным, если пользователь забудет этот пароль.

В настоящее время для защиты от несанкционированного доступа к информации все более часто используются биометрические системы авторизации и идентификации пользователей. Используемые в этих системах характеристики являются неотъемлемыми качествами личности человека и поэтому не могут быть утерянными и подделанными. К биометрическим системам защиты информации относятся системы распознавания речи, системы идентификации по отпечаткам пальцев, а также системы идентификации по радужной оболочке глаза.

Защита программ от нелегального копирования и использования.

Компьютерные пираты, нелегально тиражируя программное обеспечение, обесценивают труд программистов, делают разработку программ экономически невыгодным бизнесом. Кроме того, компьютерные пираты нередко предлагают пользователям недоработанные программы, программы с ошибками или их демоверсии.

Для того чтобы программное обеспечение компьютера могло функционировать, оно должно быть установлено (инсталлировано). Программное обеспечение распространяется фирмами-производителями в форме дистрибутивов на CD-ROM. Каждый дистрибутив имеет свой серийный номер, что препятствует незаконному копированию и установке программ.

Для предотвращения нелегального копирования программ и данных, хранящихся на CD-ROM, может использоваться специальная защита. На CD-ROM может быть размещен закодированный программный ключ, который теряется при копировании и без которого программа не может быть установлена.

Защита от нелегального использования программ может быть реализована с помощью аппаратного ключа, который присоединяется обычно к параллельному порту компьютера. Защищаемая программа обращается к параллельному порту и запрашивает секретный код; если аппаратный ключ к компьютеру не присоединен, то защищаемая программа определяет ситуацию нарушения защиты и прекращает свое выполнение.

Защита данных на дисках.

Каждый диск, папка и файл локального компьютера, а также компьютера, подключенного к локальной сети, может быть защищен от несанкционированного доступа. Для них могут быть установлены определенные права доступа (полный, только чтение, по паролю), причем права могут быть различными для различных пользователей.

Для обеспечения большей надежности хранения данных на жестких дисках используются RAID-массивы (Redundant Arrays of Independent Disks — избыточный массив независимых дисков). Несколько жестких дисков подключаются к специальному RAID-контроллеру, который рассматривает их как единый логический носитель информации. При записи информации она дублируется и сохраняется на нескольких дисках одновременно, поэтому при выходе из строя одного из дисков данные не теряются.

Защита информации в Интернете. Если компьютер подключен к Интернету, то в принципе любой пользователь, также подключенный к Интернету, может получить доступ к информационным ресурсам этого компьютера. Если сервер имеет соединение с Интернетом и одновременно служит сервером локальной сети (Интранет-сервером), то возможно несанкционированное проникновение из Интернета в локальную сеть.

Механизмы проникновения из Интернета на локальный компьютер и в локальную сеть могут быть разными:

- загружаемые в браузер Web-страницы могут содержать активные элементы ActiveX или Java-апплеты, способные выполнять деструктивные действия на локальном компьютере;
- некоторые Web-серверы размещают на локальном компьютере текстовые файлы cookie, используя которые можно получить конфиденциальную информацию о пользователе локального компьютера;
- с помощью специальных утилит можно получить доступ к дискам и файлам локального компьютера и др.

Для того чтобы этого не происходило, устанавливается программный или аппаратный барьер между Интернетом и Интранетом с помощью брандмауэра (firewall — межсетевой экран). Брандмауэр отслеживает передачу данных между сетями, осуществляет контроль текущих соединений, выявляет подозрительные действия и тем самым предотвращает несанкционированный доступ из Интернета в локальную сеть.

Основной формой работы по дисциплине СИТ являются практические занятия по программным приложениям. Предлагаемые лабораторные работы включают в себя как задания, так и рекомендации по их выполнению.

Лабораторные работы

Пакет Microsoft Office.

Текстовый редактор Microsoft Word.

Лабораторная работа № 1. Форматирование текста в редакторе Word.

Что осваивается и изучается?

Форматирование текста в текстовом редакторе Word.

Задание 1. Освойте основные приемы работы в текстовом редакторе Word

Порядок выполнения работы

1. Запустите программу **Microsoft Word**, пользуясь меню **Пуск/Программы/Microsoft Word**. На экране появится окно программы **Word**. Изучите структуру и элементы окна. Для этого необходимо нажать комбинацию клавиш **Shift-F1** и при помощи курсора выделяйте нужные элементы.
2. Сверните и разверните окно программы.
3. Создайте новый документ и затем сверните и разверните окно документа.
4. Научитесь устанавливать и убирать панели инструментов и линейку при помощи команды **Вид** и с помощью контекстного меню.
5. Установите **Линейку** и панели **Стандартная** и **Форматирование**.
6. Изучите содержимое строки состояния. Выключите и включите отображение строки состояния.
7. Создайте новый документ. В заголовке окна программы появится имя нового документа. Теперь в окне программы **Word** открыто два документа: Создайте еще один новый документ.

Научитесь переключаться между окнами документов и упорядочивать окна всех документов с помощью меню **Окно**.

8. Закройте окна всех документов.

Задание 2. Форматирование информации в текстовом редакторе Word. Изучите команду **Формат**, ее подкоманды **Шрифт**, **Абзац**, **Список**.

Создайте новый документ, содержащий копию текста, изображенного на рис. 1.

Таблица 1

Требования к формату шрифтов

Строка	Шрифт
Заголовок	Times New Roman, 14, полужирный
Подзаголовок	Times New Roman, 12, полужирный курсив
Основной текст	Times New Roman, 11


Таблица 2

Требования к формату абзацев

Строка	Абзац		
	Выравнивание	Отступы, см	Интервалы, см
Заголовок	По центру	Слева – 0 Справа – 0 Первая строка – 0	Перед – 6 После – 6 Межстрочный – 1
Подзаголовок	По левому краю	Слева – 0 Справа – 0 Первая строка – отступ 1	Перед – 3 После – 3 Межстрочный – 1
Основной текст	По ширине	Слева – 0 Справа – 0 Первая строка – 1	Перед – 0 После – 0 Межстрочный – 1

Основы форматирования в Word

Шрифт

Настройка формата  символов осуществляется в диалоге [Формат-Шрифт] и включает такие характеристики:

шрифт (Arial, Times, Courier);

1. начертание (Обычный, Курсив, Полужирный, Полужирный курсив);
2. размер;
3. подчеркивание;
4. цвет;
5. эффекты (зачеркнутый, двойное зачеркивание, верхний индекс, нижний индекс, с тенью, контур, приподнятый, утопленный, МАЛЫЕ ПРОПИСНЫЕ, ВСЕ ПРОПИСНЫЕ,);
6. интервал (обычный, уплотненный, р а з р е ж е н н ы й);
7. смещение (нет, вверх, вниз).

Абзац

Формат абзаца (меню [Формат-Абзац]) включает такие параметры.

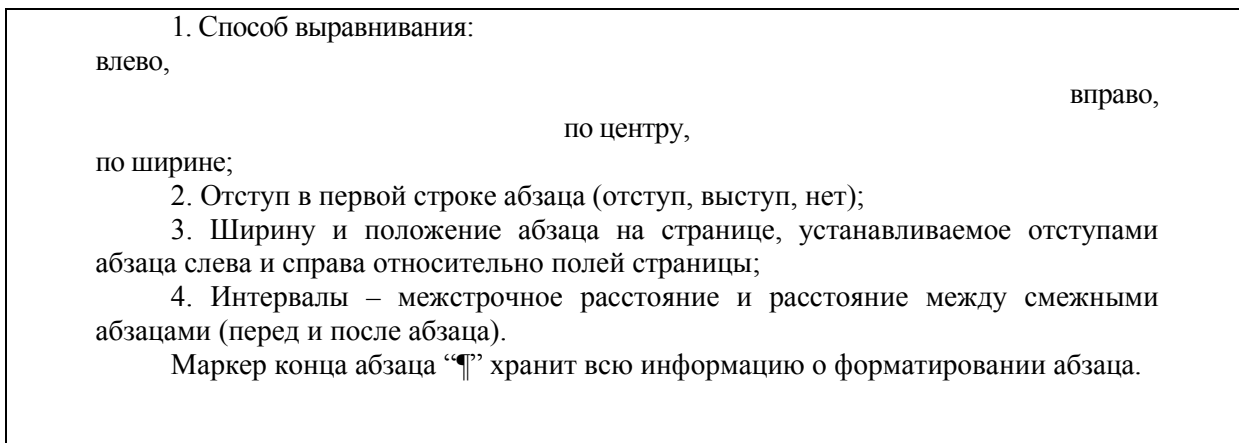


Рис. 1.

Для вставки специального символа “¶” примените команду **Вставка/Символ**.

Для нумерации строк необходимо строки выделить и нажать на кнопку нумерация на панели **Форматирование**.

Задайте следующие параметры страницы с помощью меню **Файл/Параметры страницы** или линейки: левое – 3,5; правое – 2,5; верхнее – 2; нижнее – 2.

Для вставки рисунка перенесите текущую позицию редактирования в начало текста и создайте новый абзац. Вставьте рисунок при помощи команды **Вставка/Рисунок**. Установите соответствующие размеры. Выполните команду контекстного меню **Формат объекта** и сбросьте флажок **Поверх текста** на вкладке **Положение**.

Для вставки рисунков в начале подзаголовков примените команду **Вставка/Символ**, выберите необходимый символ из группы Wingdings, вставьте и установите нужный размер его.

*Границы абзаца устанавливаются при помощи команды **Формат/Абзац** или **Линейки**.*

Сохраните документ в своей папке под именем “Задание № 2”. Закройте документ и откройте его снова.

Задание 3. Наберите текст обращения, приведенный на рис. 2.

Акционерам общества с ограниченной ответственностью NNN Ltd		
<i>О годовом собрании акционеров общества с ограниченной ответственностью NNN Ltd</i>		
Уважаемые господа!		
Правление общества с ограниченной ответственностью NNN Ltd имеет честь известить вас о том, что годовое собрание акционеров общества NNN Ltd состоится 15 марта 2001 г. во Дворце культуры и отдыха акционеров NNN Ltd по адресу ул. Солнечная, 25.		
После собрания состоится концерт мастеров искусств и банкет.		
Перечень филиалов, в которых производится выплата дивидендов:		
	Центральный	
Название акций	Номинал (тыс. руб.)	Дивиденд (тыс. руб.)
NNN-Айболит	10	560
NNN-123	100	6000
Председатель правления		
И.И.Иванов		

Рис.2. Текст обращения для упражнения.

Для размещения текста в строке по горизонтали можно использовать клавишу табуляции **Tab** или команды **Формат/Табуляция**. При нажатии на клавишу **Tab** курсор ввода перемещается вправо на некоторое число позиций. Количество этих позиций может быть изменено при помощи команд **Формат/Табуляция**.

Для этой же цели можно использовать и линейку, с размещенными на ней символами табуляции - “L” (выравнивание слева), “┌” выравнивание по центру и “┐”(выравнивание справа). Для размещения нужного символа на линейке вначале его нужно установить в области слева от линейки, а затем щелкнуть мышкой в нужной позиции серой области под линейкой. В этом случае, при нажатии на клавишу **Tab**, курсор ввода перемещается вправо в указанную позицию, и набираемый текст размещается в соответствии с типом символа табуляции.

Для размещения текста по горизонтали также можно использовать и таблицы.

Задание 4. Создайте многоуровневый список, указанный ниже:

Программное обеспечение ЭВМ.

1. Операционные системы

- 1.1. DOS
- 1.2. WINDOWS XP
- 1.3. WINDOWS NT
- 1.4. UNIX

2. Системы программирования

- 2.1. BASIC
- 2.2. PASCAL
- 2.3. C++

3. Прикладные программы

3.1. Текстовые процессоры

- 3.1.1. WORD PAD
- 3.1.2. WORD
- 3.1.3. WORD PERFECT

3.2. Электронные таблицы

- 3.2.1. EXCEL
- 3.2.2. LOTUS
- 3.2.3. QUATROPRO

3.3. Системы управления базами данных

- 3.3.1. FOXPROX
- 3.3.2. ACCESS
- 3.3.3. ORACLE

Методические указания.

Для построения этого списка наберите первую строку и выделите ее. Выполните команды **Формат/Список/Многоуровневый** и выберите нужный вид списка и нужную нумерацию. Установите курсор в конец первой строки и нажмите клавишу **Ввод**. Добавленная строка будет иметь тот же уровень вложенности, что и предыдущая. Для увеличения уровня вложенности нажмите клавишу **Tab**, для уменьшения – **Shift+Tab**. Последовательно наберите нужные строки, устанавливая нужный уровень вложенности. В случае, если уровень вложенности будет увеличиваться не последовательно, уменьшите размер табуляции по умолчанию до 0,5см.

Этот список можно построить и иначе. Для этого необходимо набрать только текст, нажимая в конце каждой строки клавишу **Enter**. Выделяя строки, находящиеся ниже первого уровня сдвигаем их вправо на одну или две позиции табулятора (в зависимости уровня вложенности) с помощью кнопки **Увеличить отступ** на панели **Форматирование** или с помощью клавиши **Tab**. Затем выделяем весь список и выполняем команды **Формат/Список /Многоуровневый**. Выбираем нужную нумерацию и нажимаем кнопку **ОК**. В случае, если уровень вложенности не будет нужным, уменьшите размер табуляции по умолчанию до 0,5см. повторите предыдущие действия.

Построить многоуровневый список можно и не используя табуляцию. В этом случае строки каждого уровня нужно набирать с помощью подчиненных стилей, например Заголовок 1, Заголовок 2, и заголовок 3.

Лабораторная работа №2. Таблицы, сортировка таблиц, вычисление в таблицах.

Что осваивается и изучается?

*Создание и форматирование таблиц. Сортировка в таблицах.
Вычисления в таблицах. Оформление и заливка.
Для работы с таблицами предназначены команды пункта главного меню **Таблица**.*

Задание 1. Создание таблиц.

Создайте журнал (таблицу) учета текущей успеваемости студентов вашей подгруппы по информатике в сентябре и октябре месяцах, следующего вида

Факультет

Курс 1

Название предмета

Подгруппа

№	Ф.И.О.	Сентябрь					Октябрь			
		2	9	16	23	30	7	14	21	28
1										
2										
3										
4										
5										
6										
7										
8										
9										

Выполнение.

Для этого создайте новый документ, установите шрифт Times New Roman и размер 14. В первой строке введите название факультета, выровняйте по центру. Для набора следующей строки, на линейке разместите символы табуляции в позиции 5,5 (выравнивание слева) и 14,4 (выравнивание справа) и установите размер 12. Введите “Курс 1”, затем нажмите клавишу табуляции и введите название предмета, снова нажмите клавишу табуляции и укажите номер группы.

Выполните команду меню **Таблица/Добавить таблицу**, в диалоговом окне **Вставка таблицы** укажите и число столбцов – 11 и число строк –10.

Выделите столбцы с номерами 3-11, и выполните команду меню **Таблица/ Высота и ширина ячейки**. В диалоговом окне **Высота и ширина ячеек** установите ширину столбцов 3-11 равной 1,2см., ширину столбца 2 – 3,8см. и ширину 1-го столбца равной 1см.

Выделите две верхние ячейки первого столбца и выполните команду меню **Таблица/Объединить ячейки** и установите выравнивание по центру. Выполните эти действия, последовательно выделяя две верхние ячейки второго столбца, пять следующих ячеек первой строки и последние 4 ячейки первой строки.

Введите данные в соответствующие ячейки таблицы. При вводе заглавий № и Ф.И.О. для выравнивания их по вертикали использовать команды **Формат/ Абзац** и в диалоговом окне

Абзаца установить нужное значение поля **Интервал перед**. Для автоматического ввода значений в первый столбец воспользуйтесь командой **Формат/ Список**.

Выделяя нужные области таблицы с помощью команды **Формат/ Границы и заливка** придайте таблице требуемый внешний вид

Задание 2. Создание и сортировка таблиц.

1. Создайте таблицу следующего вида:

	Фамилия И.О.	Должность	Оклад
1.	Сергеев В.В.	директор	20000000
2.	Петухов В.В.	водитель	2000000
3.	Петров В.В.	зам. директора	12000000
4.	Мишина В.В.	кассир	12000000
5.	Иванов В.В.	зам. директора	12000000
6.	Дубков В.Ф.	бухгалтер	15000000
7.	Веник В.В.	водитель	2000000
8.	Ванин В.В.	водитель	2300000
9.	Ванин В.П.	водитель	2000000
10.	Сычев Т.Т.	водитель	2300000

2. Отсортируйте строки таблицы по фамилиям в алфавитном порядке.

Методические указания.

Для упорядочения таблицы проделайте следующие действия: выделите в таблице строки, начиная со второй, и столбцы, начиная со второго; выполните команду меню **Таблица/Сортировка**, в диалоговом окне **Сортировка** установите в списке **Сортировать** Столбец 2 (сортировка по 2-му столбцу), способ сортировки- **Текст**, нажмите кнопку **Параметры** и установите флажок **Только столбцы** (чтобы не переставлялись клетки с номерами строк) и нажмите кнопку **ОК**. Сохраните полученную таблицу в файле с названием *лаб.2_1.doc*.

3. Отсортируйте строки таблицы по убыванию окладов и сохраните полученную таблицу в файле с названием *лаб.2_2.doc*.

4. Отсортируйте строки таблицы по должностям и для одинаковых должностей по возрастанию окладов. Сохраните полученную таблицу в файле с названием *лаб.2_3.doc*.

5. Соедините документы, записанные в файлы в один документ. Для этого примените команду **Вставка/Файл**. Пронумеруйте таблицы в объединенном документе при помощи команды **Вставка/Название**.

6. Сохраните полученный документ в файле *Лобаторная работа_2_2.doc*.

Задание 3. Визитная карточка.

Визитная карточка – небольшой документ, в котором находится основная информация о владельце. В нее, чаще всего, заносят следующую информацию:

- Фамилию, имя, отчество владельца. В зависимости от страны и происхождения владельца, отчество может не указываться.
- Место работы (учебы) и должность (курс, группа).
- Домашний адрес.
- Рабочий и домашний телефоны, а также факс и адрес электронной почты, если они имеются

Размер визитной карточки примерно - 8 см по горизонтали и 5 см по вертикали. Структура визитной карточки приведена ниже:

<i>Место работы (учебы)</i>	
Должность (курс, группа)	
Фамилия	
Имя и отчество	
Домашний адрес	Телефон раб.
	Телефон дом.
	Fax
	E-Mail

Методические указания.

Создать визитную карточку можно следующим образом

1. Создайте новый документ
2. Вставьте таблицу из 2-х строк и 2-х столбцов
3. Установите длину первого и второго столбца равной 4 см.
4. Выделите первую строку таблицы и выполните команду **“Объединить ячейки”**. В результате получится таблица, состоящая из трех ячеек 1.2 и3, следующего вида

1	
2	3

5. Занесите в ячейку №1 место работы, должность, фамилию, имя и отчество. В ячейку №2 домашний адрес, в ячейку № 3 – рабочий и домашний телефоны, факс и адрес электронной почты.
6. Подберите нужные шрифты и их размеры, Начертание фамилии должно выделяться по отношению к другой информации. Отцентрируйте текст в ячейке № 1, ячейку № 2 . выровняйте по левому, а ячейку №3 по правому краю.
7. Выделите всю таблицу и выполните команды **“Формат, Границы и заливка”**, В диалоговом окне выберите режим **“Рамка”**, для того чтобы ваша визитка взялась в рамочку.

Визитка практически готова, но она занимает лишь небольшую часть листа формата А4.

Разместим на листе 10 копий визитки в две колонки. Для этого:

1. Выполните команды **“Формат, Колонки”** и установите для листа две колонки для размещения текста.
2. Выделите таблицу и скопируйте ее в буфер обмена.
3. Установите курсор на одну строку ниже таблицы.
4. Вставьте содержимое буфера обмена (команды **“Правка, Вставить”**). Повторите эти действия пять раз. Если пятая копия не вмещается в первой колонке, или в ней остается свободное место, измените размеры верхнего и нижнего полей страницы. Аналогично заполните правую колонку.

Задание 4. Вычисление в таблицах.

Выполнение.

1. Подготовьте документ следующего вида:

Сведения

о доходах и расходах фирмы «Ритм» за январь-март 1997 г.

	Январь	Февраль	Март	Сумма
Объем продаж	45000000	50000000	48000000	143000000
Затраты на покупку	15000000	12000000	18000000	45000000
Затраты за доставку	6000000	8000000	10000000	24000000
Доход	24000000	30000000	20000000	74000000

Председатель правления

фирмы «Ритм»

И. И. Иванов

2. Для вычисления сумм, расположенных в пятом столбце, необходимо при помощи команды **Таблица/Формула** ввести в клетки этого столбца формулы: $=b2+c2+d2$, $=b3+c3+d3$, $=b4+c4+d4$ или формулу: $=SUM(LEFT)$.

Для вычисления доходов, расположенных в пятой строке, необходимо при помощи команды **Таблица/Формула** ввести в клетки этого столбца формулы: $=b2-(b3+b4)$, $=c2-(c3+c4)$, $=d2-(d3+d4)$.

3. Сделайте обрамление и заливку клеток с исходными данными при помощи панели **Таблицы и Границы** или при помощи команды **Формат/Граница и заливка**. Измените числа в клетках с исходными данными и выполните перерасчет таблицы. Сохраните документ в файле.

Задание 5. Подготовьте рекламу следующего вида:

Ярмарк

Работает постоянно
с 11.00 до 19.00
воскресенье - выходной
вход свободный

Минск, Толбухина,4
ст. м. "Парк Челюскинцев"
тел. 266-97-24
2-й этаж-
ОДЕЖДА, ОБУВЬ, ПОДАРКИ
3-й этаж-
ВСЕ ДЛЯ ДОМА

ВСЕ, ЧТО ВАМ СЕЙЧАС НУЖНО!

Методические указания.

Создайте таблицу, сделав невидимыми границы, расположения информации и в клетки заполните нужную информацию в соответствующем формате.

Для фигурного текста примените объекты Wordart, кнопка для работы с которыми находится на панели рисование.

Лабораторная работа №3. Создание и редактирование диаграмм в документах word.

Что осваивается и изучается?

Построение диаграмм. Типы диаграмм.

Редактирование объектов диаграммы.

В состав Word входит программа создания диаграмм Microsoft Graph, включающая почти все возможности наиболее универсальной программы управления электронными таблицами

Microsoft Excel.С помощью Microsoft Graph можно создавать высококачественные, информативные диаграммы и включать их в документы Word.

Создание диаграммы

Диаграммы строятся на основе данных, содержащихся в таблице данных, также внедряемой в документ Word.Созданная диаграмма связывается с таблицей данных, поэтому при изменении исходных данных диаграмма автоматически обновляется. Можно создавать диаграммы четырнадцати основных и двадцати дополнительных типов. Кроме того, внутри каждого из основных типов можно выбрать конкретный формат (подтип).

Например, таблица, отображающая данные по объемам продаж оргтехники

	Компьютеры	Модемы	Принтеры	Ксероксы
1998 год	12000	10000	11000	10000
1999 год	14000	9000	12000	9000
2000 год	14000	8000	13000	8000
2001 год	12000	10000	14000	10000

Если необходимо создать диаграмму на основе данных из таблицы, то нужно установить точку вставки в одну из ячеек таблицы и выбрать команду **Таблица, Выделить, Таблицу**. Выбрать команду **Вставка, Рисунок, Диаграмма**. Затем нажать **ОК**.

Если необходимо создать диаграмму на основе данных, набранных в документе и разделенных символами табуляции, то нужно выделить все эти данные, включая названия, которые будут использоваться в качестве меток легенды и названий категорий.

Редактирование таблицы данных

Работая с таблицей данных можно перемещаться, выделять ячейки, столбцы или строки, изменять ширину и т.д. При изменении исходных данных меняется и сама диаграмма. Чтобы изменить содержимое ячейки таблицы данных, нужно выделить ячейку и ввести новые данные. После нажатия клавиши Enter или перехода к другой ячейке таблицы все существующие в этой ячейке данные замещаются введенными данными.

Чтобы отредактировать содержимое ячейки, нужно выделить ее, а затем нажать клавишу F2 (переход в режим редактирования) или дважды щелкнуть по ней. Изменение содержимого ячейки ничем не отличается от редактирования обычного текста. После того как необходимые исправления произведены, нужно нажать клавишу Enter.

Можно также расширить или сузить набор данных, по которому строится диаграмма, путем добавления или удаления строк и столбцов таблицы данных. При этом диаграмма автоматически перестраивается с учетом внесенных в таблицу данных изменений. Чтобы вставить в таблицу строки или столбцы, нужно выделить нужное число строк или столбцов и выбрать команду **Вставка, Ячейки**. Чтобы удалить из таблицы строки или столбцы, нужно их выделить, а затем выбрать команду **Правка, Удалить**.

Тип диаграмм

Правильный выбор типа диаграммы позволяет представить данные самым выигрышным образом. Тип диаграммы может быть применен не только ко всей диаграмме, но и к отдельному ряду данных на ней или к нескольким рядам. Комбинирование различных типов диаграмм позволяет разделить данные разного типа или выделить какой-то ряд данных, например, можно скомбинировать график с гистограммой.

Наиболее просто изменить тип всей диаграммы или только одного ряда данных с помощью команды **Диаграмма, Тип диаграммы**. В появляющемся окне можно выбрать не только тип, но и формат выбранного типа диаграммы.

Чтобы изменить тип диаграммы:

1. Нужно выбрать команду **Диаграмма, Тип диаграммы**. Появится диалоговое окно **Тип диаграммы**
2. В этом диалоговом окне раскрыть вкладку **Стандартные** для выбора одного из основных типов диаграмм или вкладку **Нестандартные** для выбора одного из дополнительных типов диаграмм.
3. В списке **Тип** выделить нужный тип диаграммы.
4. Если выбрана вкладка **Стандартные**, то в галерее форматов **Вид** нужно выделить подтип диаграммы.
5. В конце необходимо нажать кнопку **ОК**, чтобы закрыть диалоговое окно и применить выбранный формат диаграммы.

Чтобы настроить существующий тип диаграммы:

1. Нужно выбрать команду **Диаграмма. Параметры диаграммы**. Появится диалоговое окно **Параметры диаграммы**, параметры в котором могут меняться в **зависимости** от типа диаграммы.
2. С помощью вкладок этого диалогового окна можно произвести настройку таких элементов диаграммы, как заголовки, оси, линии сетки, подписи данных и т.д.
3. После внесения необходимых изменений нажмите кнопку **ОК**.

Построение составных диаграмм

Составные диаграммы – это диаграммы, построенные с использованием одновременно двух или более типов диаграмм. На таких диаграммах некоторые ряды данных представляются с помощью одного типа диаграмм, а другие – с помощью другого. Например, можно построить один ряд в виде гистограммы, а второй – в виде графика, что облегчит сравнение рядов данных и поиск их возможных связи.

Созданную диаграмму можно сделать составной, изменить тип используемый для построения одного или нескольких рядов данных. Для этого:

1. Выделить на диаграмме ряды данных, тип должен быть изменен, и выберите команду **Диаграмма, Тип диаграммы**
2. В группе **Параметры** появившегося диалогового окна **Тип диаграммы** установите флажок **Применить**

3. Выделите тип диаграммы для выделенного ряда данных и нажмите кнопку **ОК**.

К элементам диаграммы относятся маркеры, легенды, оси, метки, надписи и т. д. Они могут сделать диаграмму более эффектной и информативной.

Созданную диаграмму можно также отформатировать нужным образом, если выбрать соответствующую цветовую гамму, шрифт, сделать акцент на важных элементах, убрать лишние детали.

Форматирование любого объекта диаграммы осуществляется с помощью диалогового окна **Формат**. Чтобы открыть это окно:

1. Выделить нужный объект диаграммы, щелкнув по нему.
2. Выбрать команду **Формат, Выделенный объект**, либо просто дважды щелкнуть по объекту.

Вкладки появившегося диалогового окна содержат множество параметров форматирования, с помощью которых можно настроить отображение выделенного элемента. Форматирование любого объекта диаграмм.

Задание 1. Построение диаграмм.

Методические указания. Вызовите программу **Microsoft Graph** при помощи команды **Вставка/ Объект/ Microsoft Graph** или **Вставка/ Рисунок/ Диаграмма**. Если в буфере обмена не содержалась таблица, то программа вставляет демонстрационный пример, данные этого примера можно заменить на другие исходные данные.

Задание 2. Ознакомьтесь с командами главного меню программы **Microsoft Graph**.

Задание 3. По таблице “Сведения о доходах и расходах фирмы «Ритм»” построить диаграмму, отражающие динамику доходов и расходов фирмы «Ритм».

**Сведения
о доходах и расходах фирмы «Ритм» за январь-март 1997 г.**

	Январь	Февраль	Март	Сумма
Объем продаж	45000000	50000000	48000000	143000000
Затраты на покупку	15000000	12000000	18000000	45000000
Затраты за доставку	6000000	8000000	10000000	24000000
Доход	24000000	30000000	20000000	74000000

Методические указания. Для этого скопируйте в буфер обмена необходимые строки исходной таблицы с заголовками строк и столбцов и вызовите команду **Вставка/ Рисунок/ Диаграмма**.

Задание 4. Постройте объемную круговую диаграмму для отображения доходов и расходов фирмы за март месяц (столбец «Март») в процентном выражении.

Задание 5. Постройте плоскую круговую диаграмму для отображения доходов фирмы за первый квартал (строка «Доход») в стоимостном выражении.

Задание 6. Постройте различные типы диаграмм (гистограммы различных типов, линейчатые, графики, лепестковые, кольцевые) по данным таблицы о закупках вычислительной техники

	Компьютеры	Модемы	Принтеры	Ксероксы
1999 год	1200	1000	1100	1000
2000 год	1400	900	1200	900
2001 год	1400	800	1300	800
2002 год	1200	1000	1400	1000

Задание 7. Постройте объемную диаграмму о закупках компьютеров и принтеров в 2001 и 2002 годах. Для объемных диаграмм изучите изменение вида диаграммы.

Задание 8. Освойте редактирование параметров диаграммы (легенды, названия диаграммы, выделение сегментов диаграммы, ввод названий сегментов, изменение окраски сегментов и других элементов).

Постройте круговую диаграмму, отображающую закупку вычислительной техники в 2002 году. Сектор компьютеры необходимо окрасить в красный цвет, принтеры – в синий, модемы – в зеленый, ксероксы – в коричневый. На секторах укажите значение в процентах.

Задание 9. При помощи команды меню **Вставка/Название** пронумеруйте построенные диаграммы, следующим образом: Диаграмма 1., Диаграмма 2., и т.д. Освойте редактирование названий.

Лабораторная работа №4. Применение стилей, автотекста, автозамены и макрокоманд.

Что осваивается и изучается?

Способы редактирование текста при помощи стилей, автотекста, Автозамены и макрокоманд.

1. Создание и применение элемента Автотекста.

Зачастую на практике приходится набирать в редакторе Word одни и те же фрагменты текста. Это можно сделать один раз и пользоваться по мере необходимости. Для этого создается новый элемент автотекста и затем вставляется нужный текст по команде **Вставка/Автотекст** и заданному имени.

Задание 1. Создайте элемент автотекста "Утверждаю".

Выполнение.

Наберите текст данного блока

и

"УТВЕРЖДАЮ"
Проректор по научно- Исследовательской работе
_____ И.О.Фамилия
“ ___ ” _____ 1999 г.

выделите его. Выполните команду **Вставка/Автотекст/Создать**. Примите имя элемента списка автотекста, предложенное по умолчанию или введите другое имя.

Для применения этого элемента **Автотекста** установите курсор в начало страницы, где должен быть данный блок. Выберите команду **Вставка/Автотекст**, на вкладке **Автотекст** из списка элементов автотекста выберите нужный элемент .

2. Применение элемента Автозамена.

Автозамена применяется для автоматического исправления ошибок, при вводе часто повторяющейся последовательности символов или специального значка. Для этого предназначена команда **Сервис/ Автозамена**. Например, при вводе слова ”волна” ошибочно может набираться слово ”влона”. Для автоматического исправления такой ошибки нужно выбрать команду **Сервис/Автозамена** и задать замену слова ”влона” на слово ”волна”. Теперь при вводе слово ”влона” будет автоматически исправляться.

Для автоматического ввода длинной последовательности символов можно при помощи команды **Сервис/Автозамена** указать замену некоторого сокращения на длинную последовательность символов. Так, можно вместо слов “БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ” вводить их сокращение БГУ.

Задание 2

Создать автозамену для следующих слов:

БГУ- БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ЭФ – Экономический факультет

ЭИМЭ – экономическая информатика и математическая экономика

3.Создание макрокоманд.

Предположим, что при редактировании текста необходимо выделять несколько символов другим шрифтом, например, весь текст набирается шрифтом **Times New Roman** обычного начертания и размера 12, а выделенные символы набираются шрифтом с параметрами **Arial**, курсив, размер 14, разреженный интервал, цвет-красный. Для этого удобно применить два макроса, один макрос установит новые параметры шрифта, а другой восстановит прежние

Задание 3. Создать макрокоманду с именем М1 для установки вышеописанного шрифта. Вызов макроса должен осуществляться горячими клавишами **Ctrl+Shift+1**

Выполнение.

Формирование первого макроса

Выделите произвольный текст. Выберите команду **Сервис/Макрос/Начать запись** (к курсору прикрепится кассета – признак что мы в режиме записи макроса) . В диалоговом окне **Запись макроса** укажите имя макроса, например, М1, присвойте макросу комбинацию клавиш

Ctrl+Shift+1, нажав на кнопку *клавишам* и введя в окошко *Новое сочетание клавиш* нужную комбинацию нажмите кнопки *назначить* и *заккрыть*. Нажмите кнопку **ОК**. Теперь все ваши операции (действия) будут записываться в макрос.

Выберите команду **Формат/Шрифт**. В диалоговом окне **Шрифт** (вкладки **Шрифт** и интервал) установите параметры шрифта: **Arial**, курсив, размер 14, разреженный интервал, цвет-красный.

Выполните команду **Сервис/Макрос/Остановить запись**. На этом запись первой макрокоманды завершается.

Для применения этого макроса необходимо выполнить команду **Сервис/Макрос/Макросы** из списка макросов выбрать нужный и нажать на кнопку **Выполнить**.

Задание 4. Создать макрокоманду с именем М2 для установки стандартного шрифта: обычного начертания, 12 размера, черного цвета.

Вызов макроса должен осуществляться горячими клавишами **Ctrl+Shift+2**

Выполнение.

Формирование второго макроса.

Второй макрос создается подобным образом. Только задайте имя макроса М2 и присвойте ему комбинацию клавиш **Ctrl+Shift+2**. В макрос запишите установку шрифта **Times New Roman** обычного начертания и размера 12:

Выполните редактирование текста, обращаясь к макросам по их именам или при помощи соответствующих комбинаций клавиш.

Задание 5

Запишите макрос с именем ТАБЛ3 для добавления таблицы из трех столбцов и четырех строк в текущую позицию курсора и присвойте ему комбинацию клавиш **Ctrl+Shift+3..**

Методические указания

Начните запись макроса ТАБЛ3 и выполните команду **Таблица/Добавить/Таблица**, задав нужное количество строк и столбцов. Остановите запись

Задание 6

Запишите макрос с именем АТУ для применения элемента автотекста **Утверждаю** и присвойте ему комбинацию клавиш **Ctrl+Shift+4..**

Методические указания

Начните запись макроса АТУ и выполните команду **Вставка/Автотекст/Автотекст** и выберите имя созданного ранее элемента автотекста «Утверждаю»

Задание 7

Запишите макрос с именем ВИ для перехода в набор верхнего индекса шрифта текста. Вызов макроса должен осуществляться горячими клавишами **Ctrl+Shift+V** либо с панели инструментов по значку **↑**.

Задание 8

Запишите макрос с именем Конверт для вставки символа ☒ в текущую позицию курсора. Вызов макроса должен осуществляться горячими клавишами **Ctrl+Shift+K**, либо с панели инструментов по значку ☒, либо по команде меню Вставка/Конверт.

Лабораторная работа №5. Слияние документов.

Что осваивается и изучается?
Построение источника данных, основного документа и слияние документов.

Немного теории.

Слияние документов - это объединение основного документа, содержащего постоянную часть информации, и источника данных, содержащих переменную часть. Примером слияния документов может быть персонализация писем. Текст делового письма постоянный, например, сообщение участникам математической олимпиады. Это основной документ. Такое письмо нужно выслать участникам олимпиады. Переменным является Фамилия И.О. участника, его адрес, набранные баллы. Данные об участниках представляют собой источник данных (список). Слияние проходит по следующей схеме.

Источник данных
(список)

Фамилия	Имя	Отчество	Индекс	Адрес	Сумма_баллов
Петров	Иван	Сергеевич	220015	г. Минск ул. Я. Мавра д.23 кв.12	25
Сергеев	Петр	Иванович	220088	г. Минск ул. Ленина, д.34 кв. 112	30

Основной документ

Поле слияния

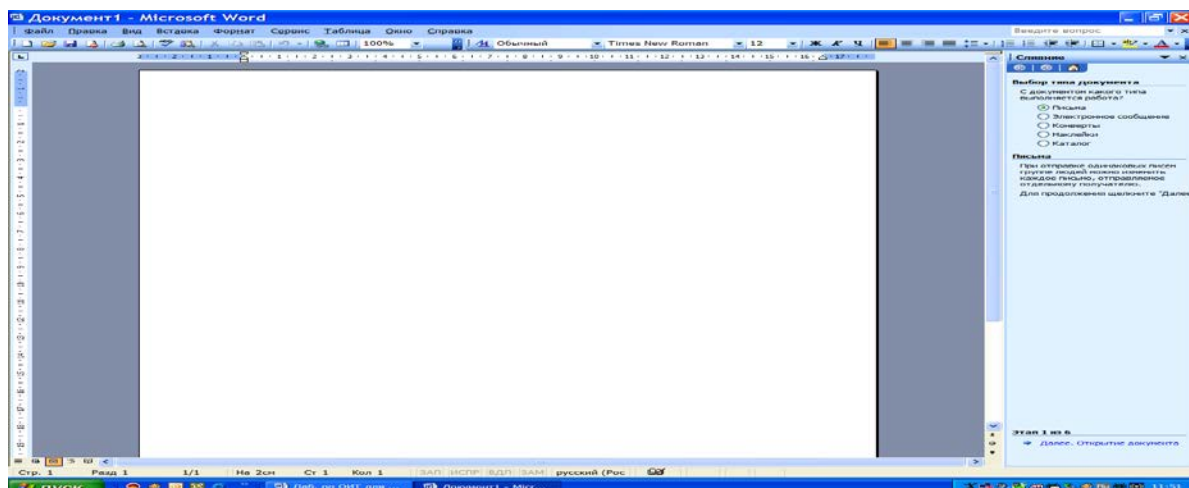
```
<<Индекс>>  
<<Адрес>>  
  
Уважаемый << Фамилия >> << Имя>> << Отчество >>!  
  
Сообщаем, что Вы, участвуя в математической олимпиаде, набрали  
<<Сумма_баллов>> баллов.  
  
Оргкомитет
```

Результат слияния

220015
г. Минск ул. Я. Мавра д.23 кв.12
Уважаемый Петров Иван Сергеевич!
Сообщаем, что Вы, участвуя в математической олимпиаде, набрали 25 баллов.
Оргкомитет
220080
г. Минск ул. Ленина, д.34 кв. 112
Уважаемый Сергеев Петр Иванович!
Сообщаем, что Вы, участвуя в математической олимпиаде, набрали 30 баллов.
Оргкомитет

В результате слияния основного документа и источника данных (списка) для каждого участника из списка готовится письмо. В итоге получается несколько писем одинакового содержания.

Слияние документов выполняется в диалоговом окне «Слияние», вызываемом командой **Сервис/Письма и рассылки/Слияние**.



Работа по слиянию документов состоит из шести этапов:

- выбор типа документа (письма, электронное сообщение, конверты, наклейки, каталог);
- выбор документа (текущий документ, шаблон, существующий документ);
- выбор получателей (создание списка, использование существующего списка, контакты Outlook);
- создание документа (основной документ с полями слияния);
- просмотр полученных документов (результат слияния);
- завершение слияния.

Кроме этого, пользователь может вносить изменения в основной документ и в список источника данных, т.е. возвращаться к любому этапу.

Задание 1. Выполните слияние документов, которые изображены на схеме, и получите письма приглашения на олимпиаду.

Выполнение.

Выполните команду Word **Файл/Создать**.

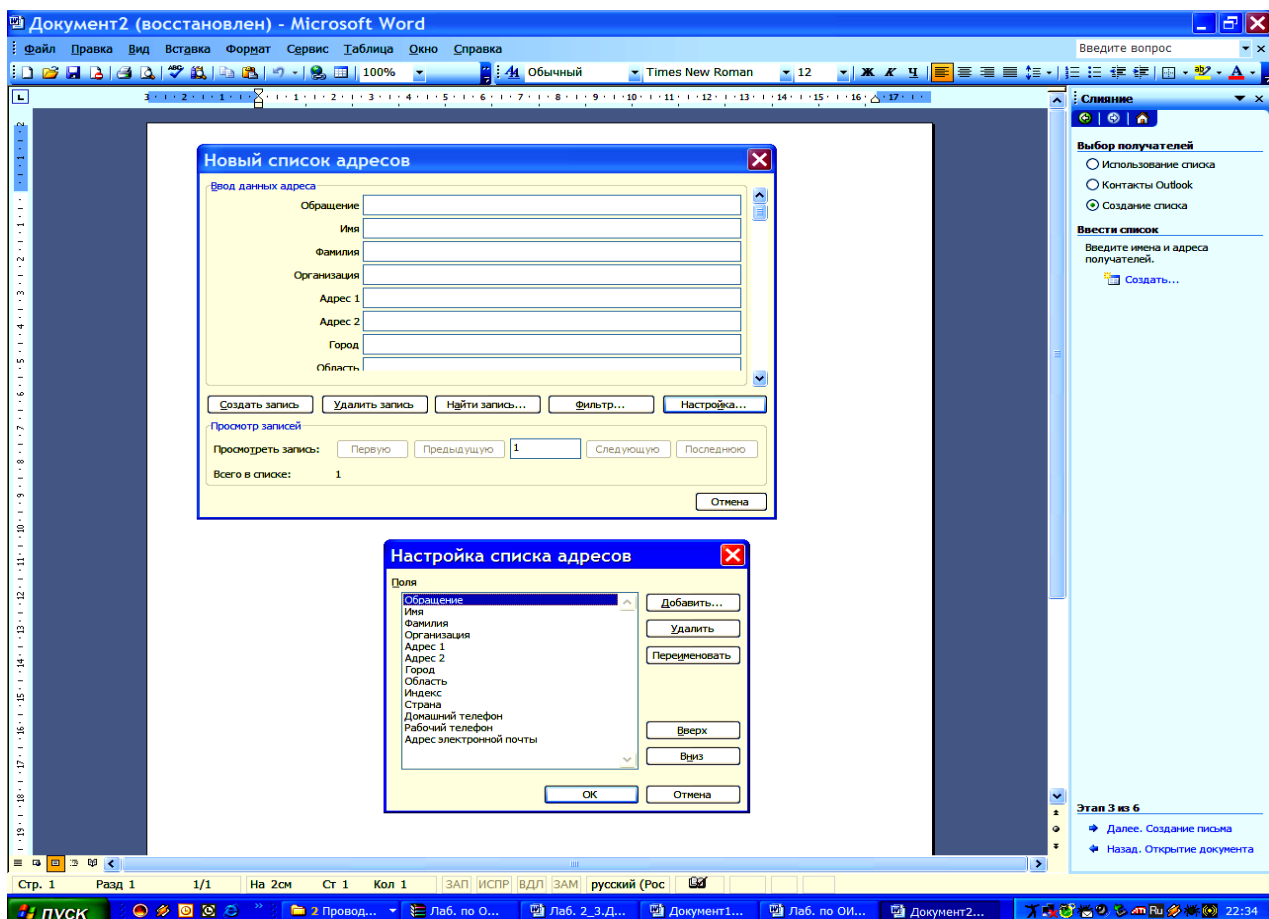
Запустите команду **Сервис/Письма и рассылки/Слияние**.

Выполняйте последовательно этапы друг за другом, используя кнопку **Вперед (Далее)**

Этап 1. Выбор типа документа – *письма*.

Этап 2. Выбор документа – *текущий документ*.

Этап 3. Выбор получателей – *создание списка* (нажмите кнопку **Создать**). В окне **Новый список адресов** нажмите кнопку **Настройка**. Используя кнопки настройки **Добавить**, **Удалить**, **Переименовать**, создайте нужный список получателей, **заполните 5 записей** произвольными данными и сохраните источник данных в файле.



Этап 4. Создание письма (основного документа).

Подготовьте основной документ, который изображен на схеме.

Поля отмеченные << >> выбирайте из закладки *Другие элементы* или воспользуйтесь кнопкой

Вставить поле слияния

Этап 5. Просмотр писем.

Просмотрите полученные письма.

Если нужно внести изменения в список или текст письма, вернитесь к соответствующему этапу.

Этап 6. Завершить слияние.

Запишите созданные письма в новый документ (команда – изменить часть писем)

Задание 2.

Выполните слияние основного документа и источника данных, только для тех записей, в которых сумма баллов **не меньше 20**. При этом измените источник данных так, чтобы в каждой строке “сумма_баллов” и соответствующее слово “баллов” были **согласованы**.

Сохраните результат в документе Ir-5

Методические указания.

Перейдите к **этапу 3** и воспользуйтесь командой *Изменить список*, задав в фильтре дополнительно условие отбора.

При этом нужно ввести **дополнительный столбец** (назовите его "баллы") в источник данных, в котором слова "баллы" будут находиться в соответствии с "суммой_баллов".

Задание 3.

Создайте источник данных с именем "Должностной список" (не менее 5 записей) и основной документ "Зачисление на работу" для получения форм следующего содержания:

Уважаемый <<Ф.И.О.>>!

Сообщаем Вам, что Вы зачислены на работу в должности <<должность>> с окладом <<xxxxxx>> рублей.

Председатель правления ООО "Фантазия" Иванов И.И.

Задание 4.

Модифицируйте основной документ "Зачисление на работу" и источник данных "Должностной список" так, чтобы в результирующем письме к лицам женского пола обращение было "Уважаемая", а к лицам мужского пола – "Уважаемый".

Результат добавьте в документ Ir-5

Методические указания.

Добавьте в источник данных поле "Пол" и заполните его. В основной документ вместо слова "Уважаемый" вставьте стандартное поле Word **IF...THEN...ELSE** и задайте соответствующее условие.

Электронные таблицы Microsoft Excel

Лабораторная работа №1.

Основные приемы работы с электронными таблицами

Упражнение 1

Введение основных понятий, связанных с работой электронных таблиц Excel.

1. Запустите программу Microsoft Excel, любым, известным вам способом. Внимательно рассмотрите окно программы Microsoft Excel. Первый взгляд на горизонтальное меню и панели инструментов несколько успокаивает, так как многие пункта горизонтального меню и кнопки панелей инструментов совпадают с пунктами меню и кнопками окна редактора Word.

Совсем другой вид имеет рабочая область и представляет из себя размеченную таблицу, состоящую из ячеек одинакового размера. Одна из ячеек явно выделена (обрамлена черной рамкой). Как выделить другую ячейку? Достаточно щелкнуть по ней мышью, причем указатель мыши в это время должен иметь вид светлого креста.

Попробуйте выделить различные ячейки таблицы. Для перемещения по таблице воспользуйтесь полосами прокрутки.

2. Для того, чтобы ввести текст в одну из ячеек таблицы, необходимо ее выделить и сразу же (не дожидаясь появления столь необходимого нам в процессоре Word текстового курсора) “писать”.

Выделите одну из ячеек таблицы и “напишите” в ней название сегодняшнего дня недели.

Основным отличием работы электронных таблиц от текстового процессора является то, что после ввода данных в ячейку, их необходимо зафиксировать, т. е. дать понять программе, что вы закончили вводить информацию в эту конкретную ячейку,

Зафиксировать данные можно одним из способов:

- нажать клавишу (Enter);
- щелкнуть мышью по другой ячейке,
- воспользоваться кнопками управления курсором на клавиатуре (перейти к другой ячейке).

Зафиксируйте введенные вами данные.

Итак, недостаточно ввести данные в ячейку, необходимо их еще и зафиксировать.



Выделите ячейку таблицы, содержащую день недели, и воспользуйтесь кнопками выравнивания абзацев. Каким образом происходит выравнивание? Сделайте вывод. После всех экспериментов обязательно верните исходное выравнивание — влево, в дальнейшем это будет важно.

3. Вы уже заметили, что таблица состоит из столбцов и строк, причем у каждого из столбцов есть свой заголовок (А, В, С...), и все строки пронумерованы (1, 2, 3...). Для того, чтобы выделить столбец целиком, достаточно щелкнуть мышью по его заголовку, чтобы выделить строку целиком, нужно щелкнуть мышью по ее заголовку.

Выделите целиком тот столбец таблицы, в котором расположено введенное вами название дня недели.

Каков заголовок этого столбца?

Выделите целиком ту строку таблицы, а которой расположено название дня недели-

Какой заголовок имеет эта строка?

Воспользуйтесь полосами прокрутки для того, чтобы определить сколько строк имеет таблица и каково имя последнего столбца.

4. *Выделите ту ячейку таблицы, которая находится в столбце С и строке 4.* Обратите внимание на то, что в Поле имени, расположенном выше заголовка столбца А, появился адрес выделенной ячейки С4. Выделите другую ячейку, и вы увидите, что в Поле имени адрес изменился.

Выделите ячейку D5; F2; A16.

Какой адрес имеет ячейка, содержащая день недели?

5. Давайте представим, что в ячейку, содержащую день недели нужно дописать еще и часть суток. *Выделите ячейку, содержащую день недели, введите с клавиатуры название текущей части суток, например, "утро" и зафиксируйте данные, нажав клавишу {Enter}.*

Что произошло? Часть суток не "дописалась" в ячейку, а новые данные заменили исходные и вместо дня недели вы подучили часть суток. То есть, если выделить ячейку таблицы, содержащую некоторые данные и ввести новые данные с клавиатуры, в ячейке таблицы окажется последняя информация.

Как же дополнить содержимое ячейки таблицы (отредактировать), не набирая заново все данные? Выделив ячейку, содержащую часть суток, вы увидите, что ее содержимое дублируется в Строке формул, расположенной выше заголовков столбцов. Именно в Строку формул можно щелчком мыши установить традиционный текстовый курсор, внести все требуемые изменения и затем зафиксировать окончательный вариант данных.

Выделите ячейку таблицы, содержащую часть суток, установите текстовый курсор перед текстом в Строке формул и наберите заново день недели. Зафиксируйте данные. У вас должна получиться следующая картина (рис.1.1):

	вторник, утро		

рис.1.1.

Видно, что запись вышла за пределы своей ячейки и заняла часть соседней. Это происходит только тогда, когда соседняя ячейка пуста. Давайте ее заполним и проверим, что изменится.

Выделите ячейку таблицы, расположенную правее ячейки, содержащей ваши данные (ячейку, на которую они "заехали ") и введите в нее любой текст.

Теперь видна только та часть ваших данных, которая помещается в ячейке (рис. 1.2). Как просмотреть всю запись? И опять к вам на помощь придет Строка Формул. Именно в ней можно увидеть все содержимое выделенной ячейки.

	вторник,	пятница	

рис.1.2.

Выделите ячейку, содержащую день недели и часть суток, и просмотрите в Строке формул полное содержимое ячейки. Итак, Строка формул позволяет:

- внести изменения в содержимое выделенной ячейки;
- просмотреть содержимое ячейки, если запись не видна целиком.

6. Как увеличить ширину столбца для того, чтобы в ячейке одновременно были видны и день недели, и часть суток?

Для этого подведите указатель мыши к правой границе заголовка столбца, "поймайте" момент, когда указатель мыши примет вид черной двойной стрелки, и, удерживая нажатой левую клавишу мыши, переместите границу столбца вправо. Столбец расширился. Аналогично можно сужать столбцы и изменять высоту строки.

Измените ширину столбца, содержащего день недели и часть суток таким образом, чтобы весь введенный текст был виден в ячейке таблицы.

7. Часто бывает нужно выделить не одну ячейку и не целый столбец, а блок ячеек (несколько ячеек, расположенных рядом).

Для этого нужно установить указатель мыши в крайней ячейке выделения и, при нажатой левой клавише, переместить мышь к противоположному краю выделения (весь выделенный блок "охвачен" рамкой, все ячейки, кроме той, с которой начали выделение, окрашены в черный цвет).

Обратите внимание, что в процессе выделения в Поле имени регистрируется количество строк и столбцов, попадающих в выделение. В тот же момент, когда вы отпустили левую клавишу, в Поле имени высвечивается адрес активной ячейки, ячейки, с которой начали выделение (адрес активной ячейки, выделенной цветом).

Выделите блок ячеек, начав с ячейки A1 и закончив ячейкой, содержащей "сегодня".

Для выделения всей таблицы используйте "пустую" угловую кнопку, расположенную над заголовком первой строки.

Выделите таблицу целиком. Снимите выделение, щелкнув мышью по любой ячейке.

8. Каким образом удалить содержимое ячейки? Для этого достаточно выделить ячейку (или блок ячеек) и нажать клавишу {Delete} или воспользоваться командой горизонтального меню **Правка⇒Очистить**.

Удалите все свои записи.

Упражнение 2

Применение основных приемов работы с электронными таблицами: ввод данных в ячейку. Форматирование шрифта. Изменение ширины столбца. Автозаполнение, ввод формулы, обрамление таблицы, выравнивание текста по центру выделения, набор нижних индексов.

Составим таблицу, вычисляющую n -й член и сумму арифметической прогрессии.

Для начала напомним формулу n -го члена арифметической прогрессии:

$$a_n = a_1 + d(n-1)$$

и формулу суммы n первых членов арифметической прогрессии:

$S_n = (a_1 + a_n) * n / 2$, где a_1 - первый член прогрессии, а d — разность арифметической прогрессии.

На рис. 1.3 представлена таблица для вычисления n -го члена и суммы арифметической прогрессии, первый член которой равен -2, а разность равна 0,725.

Перед выполнением упражнения придумайте свою арифметическую прогрессию, т. е. задайте собственные первый член прогрессии и разность.

Вычисление n-го члена и суммы арифметической прогрессии			
d	n	a _n	S _n
0,725	1	-2	-2
0,725	2	-1,275	-3,275
0,725	3	-0,55	-3,825
0,725	4	0,175	-3,65
0,725	5	0,9	-2,75
0,725	6	1,625	-1,125
0,725	7	2,35	1,225
0,725	8	3,075	4,3
0,725	9	3,8	8,1
0,725	10	4,525	12,625

Рис. 1.3.

Выполнение упражнения можно разложить по следующим этапам.

- Выделите ячейку A1 и введите в нее заголовок таблицы "Вычисление n-го члена и суммы арифметической прогрессии". Заголовок будет размещен в одну строчку и займет несколько ячеек правее A1.

- Сформатируйте строку заголовков таблицы. В ячейку A3 введите "d", в ячейку B3 - "n", в C3 - "a_n", в D3 - "S_n".

Для набора нижних индексов воспользуйтесь командой **Формат⇒Ячейки...**, выберите вкладку Шрифт и активизируйте переключатель *Подстрочный* в группе переключателей *Эффекты*.

Выделите заполненные четыре ячейки и при помощи соответствующих кнопок панели инструментов увеличьте размер шрифта на 1 пт выровняйте по центру и примените полужирный стиль начертания символов.

Строка-заголовок вашей таблицы оформлена. Можете приступить к заполнению.

- В ячейку A4 введите величину разности арифметической прогрессии (в нашем примере это 0,725).
- Далее нужно заполнить ряд нижних ячеек таким же числом. Набирать в каждой ячейке одно и то же число неинтересно и нерационально. В редакторах Paintbrush и Word мы пользовались приемом копировать—вставить. Excel позволяет еще больше упростить процедуру заполнения ячеек одинаковыми данными.

Выделите ячейку A4, в которой размещена разность арифметической прогрессии. Выделенная ячейка окаймлена рамкой, в правом нижнем углу которой есть маленький черный квадрат -маркер заполнения.

Если подвести указатель мыши к маркеру заполнения, и в тот момент, когда указатель мыши принимает форму черного крестика, протянуть маркер заполнения на несколько ячеек вниз, то весь ряд выделенных ячеек заполнится данными, расположенными в первой ячейке.

Заполните таким образом значением разности арифметической прогрессии еще девять ячеек ниже ячейки A4.

- В следующем столбце размещена последовательность чисел от 1 до 10. И опять нам поможет заполнить ряд маркер заполнения. Введите в ячейку B4 число 1, в ячейку B5 число 2, выделите **обе** эти ячейки и, ухватившись за маркер заполнения, протяните его вниз.

Отличие от заполнения одинаковыми данными заключается в том, что, выделив две ячейки, вы указали принцип, по которому следует заполнить оставшиеся ячейки.

- Маркер заполнения можно "протаскивать" не только вниз, но и вверх, влево или вправо, в этих же направлениях распространится и заполнение. Элементом заполнения может быть не только формула или число, но и текст. Можно ввести в ячейку "январь" и, заполнив ряд дальше вправо получить "февраль", "март", а "протянув" маркер заполнения от ячейки "январь" влево, соответственно получить "декабрь", "ноябрь" и т. д. Попробуйте.

Самое главное, прежде, чем распространять выделение, выделить именно ту ячейку (или те ячейки), по которой форматируется заполнение.

- В третьем столбце размещаются n-е члены прогрессии. Введите в ячейку C4 значение

первого члена арифметической прогрессии.

В ячейку C5 нужно поместить формулу для вычисления n-го члена прогрессии, которая заключается в том, что каждая ячейка столбца отличается от предыдущей прибавлением разности арифметической прогрессии.

Все формулы начинаются со знака равенства.

Для того, чтобы ввести формулу необходимо выделить ячейку, в которую хотите поместить формулу, набрать знак равенства и затем набрать саму формулу со ссылками на соответствующие ячейки таблицы (не забудьте, что заголовки столбцов определяются латинскими буквами и русские А, С, В, хоть и похожи на такие же буквы латинского алфавита, но не являются равноценной заменой).

Выделите ячейку C5 и наберите в ней формулу $=C4+A4$ (не забудьте перейти на латиницу, а вместо ссылки на ячейку A4 можно ввести конкретное значение разности вашей арифметической прогрессии).

Можно и не набирать с клавиатуры адрес той ячейки, на которую делается ссылка. Набрав знак равенства, щелкните мышью по ячейке C4 и в строке формул появится ее адрес, затем продолжите набор формулы. В этом случае вам не нужно переключаться на латиницу.

Полностью введя формулу, зафиксируйте ее нажатием {Enter}, в ячейке окажется результат вычисления по формуле, а в Строке формул сама формула.

Вот проявилась и еще одна функция Строки формул: если в ячейке вы увидите результат вычислений по формуле, то саму формулу можно просмотреть в Строке формул, выделив соответствующую ячейку.

Если вы неправильно набрали формулу, исправить ее можно в Строке формул, предварительно выделив ячейку.

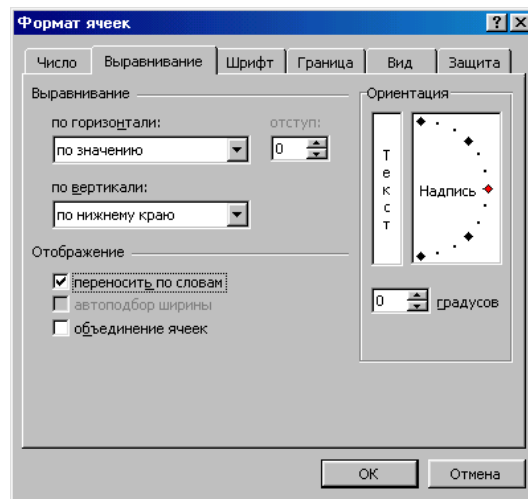
- Выделите ячейку C5 и, аналогично заполнению ячеек разностью прогрессии, заполните формулой, "протаскив" маркер заполнения вниз, ряд ячеек, ниже C5.

Выделите ячейку C8 и посмотрите в Строке формул, как выглядит формула, она приняла вид $=C7+A7$. Заметно, что ссылки в формуле изменились относительно смещению самой формулы.

- Аналогично введите в ячейку D4 формулу $=(-2+C4)*n/2$ для подсчета суммы n первых членов арифметической прогрессии, где вместо -2 должен быть первый член вашей арифметической прогрессии.
- Выделите ячейку D4 и заполните формулами нижние ячейки, протаскив вниз маркер заполнения.
- Теперь данными заполнены все ячейки, остается их только оформить.

Все столбцы одинаковой ширины, хотя и содержат информацию разного объема. Можно вручную (используя мышь) изменить ширину отдельных столбцов, а можно автоматически подогнать ширину.

Выделите все ячейки таблицы, содержащие данные (не столбцы целиком, а только блок заполненных ячеек без заголовка "Вычисление n-го члена и суммы арифметической прогрессии") и выполните команду



Формат⇒Столбец⇒Подгон ширины

Пришла пора заняться заголовком таблицы "Вычисление n-го члена и суммы арифметической прогрессии".

Рис. 1.5.

Выделите ячейку A1 и примените полужирное начертание символов к содержимому ячейки. Заголовок довольно неэстетично "вылезает" вправо за пределы нашей маленькой таблички.

Выделите четыре ячейки от A1 до D1 и выполните команду **Формат⇒Ячейки...**, выберите закладку Выравнивание и установите переключатели в положение "Центрировать по выделению" (Горизонтальное выравнивание) и "Переносить по словам" (рис. 1.5). Это позволит расположить заголовок в несколько строчек и по центру **выделенного блока** ячеек.

- Таблицу почти привели к виду образца. Если а этот момент выполнить просмотр **Файл⇒Предварительный просмотр**, то окажется, что остается выполнить оформление таблицы.

Для этого выделите таблицу (без заголовка) и выполните команду **Формат-Ячейки...**, выберите вкладку Граница, определите стиль линии и активизируйте переключатели Сверху, Снизу, Слева, Справа (рис. 1.6). Данная процедура распространяется на каждую из ячеек.

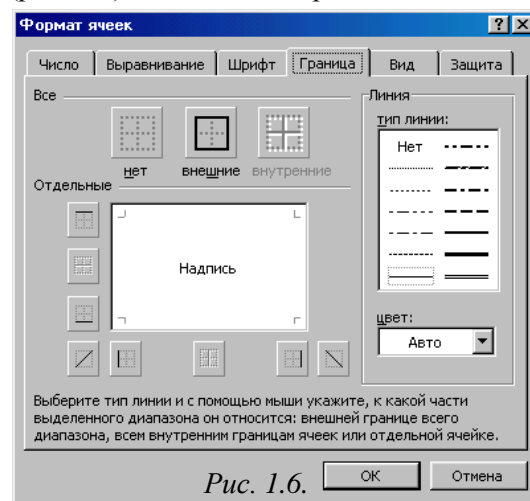


Рис. 1.6.

Затем выделите блок ячеек, относящихся к заголовку: от A1 до D2 и, проделав те же операции, установите переключатель Контур. В этом случае получается рамка вокруг всех выделенных ячеек, а не каждой.

- Выполните просмотр.

Лабораторная работа №2.

Форматирование таблиц и оперирование данными отдельных ячеек

Упражнение 1

Закреплена основных навыков работы с электронными таблицами, знакомство с понятиями: сортировка данных, типы выравнивания текста в ячейке, формат числа.

Грузоотправитель и его адрес

Грузополучатель и его адрес

К Реестру №

Дата получения

«__»_____200__г.

СЧЕТ № 123 от 15.11.2000

Поставщик Торговый Дом Рога и Копыта

Адрес 660045, Красноярск, ул. Пушкина, 23

Р/счет № 45638078 в МММ-банке, МФО 985435

Дополнения:

Руководитель предприятия

Сидоркин А.Ю.

Главный бухгалтер

Иванова А.Н.

№	Наименование	Ед.измерени я	Кол-во	Цена	Сумма
1					
2					
3					
4					
5					
6					
ИТОГО					

Упражнение заключается в создании и заполнении бланка товарного счета.

Выполнение упражнения лучше всего разбить на три этапа:

1-й этап. Создание таблицы бланка счета.

2-й этап. Заполнение таблицы.

3-й этап. Оформление бланки.

1-й этап.

Заключается в создании таблицы.

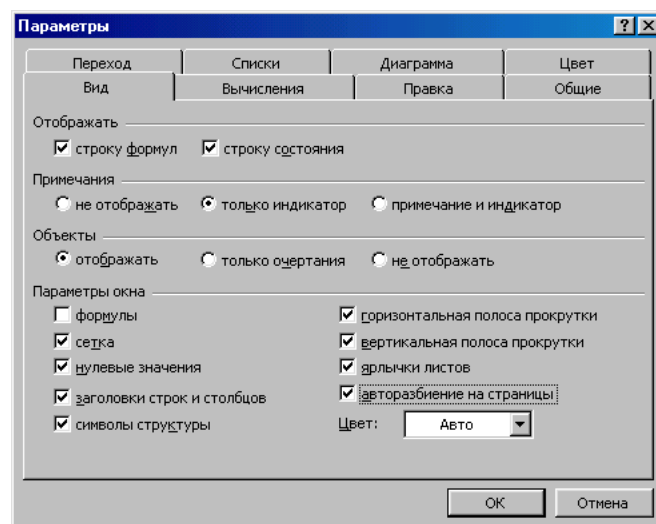
Основная задача уместить таблицу по ширине листа. Для этого:

- предварительно установите поля, размер и ориентацию бумаги (**Файл**⇒**Параметры страницы...**),
- выполнив команду **Сервис**⇒**Параметры...**, в группе переключателей *Параметры окна* активизируйте переключатель *Авторазбиение на страницы* (рис. 2.1).

В результате вы получите в виде вертикальной пунктирной линии правую границу полосы набора (если ее не видно, переместитесь при помощи горизонтальной полосы прокрутки вправо) и нижнюю границу полосы набора (для того, чтобы ее увидеть переместитесь при помощи вертикальной полосы прокрутки вниз).

Рис. 2.1.

Авторазбиение на страницы позволяет уже в процессе набора данные и форматирования таблицы следить за тем, какие столбцы помещаются на странице, а какие нет.



№	Наименование	Ед.измерени я	Кол-во	Цена	Сумма
1					
2					
3					
4					
5					
6					
ИТОГО					

- Создайте таблицу по предлагаемому образцу с таким же числом строк и столбцов.
- Выверните и сформатируйте шрифт в ячейках-заголовках, подберите ширину столбцов, изменяя ее при помощи мыши.
- Введите нумерацию в первом столбце таблицы, воспользовавшись помощью маркера заполнения.
- "Разлините" таблицу, используя линии различной толщины. Обратите внимание на то, что в

последней строке пять соседних ячеек не имеют внутреннего обрамления.

Проще всего добиться этого следующим путем:

- выделить всю таблицу и установить рамку — "Контур" жирной линией;
- затем выделить все строки, кроме последней и установить рамку тонкой линией "Справа", "Слева", "Сверху", "Снизу";
- после этого выделить отдельно самую правую ячейку нижней строки и установить для нее рамку "Слева" тонкой линией;
- останется выделить первую строку таблицы и установить для нее рамку "Снизу" жирной линией.

Хотя можно действовать и наоборот. Сначала "разлиновать" всю таблицу, а затем снять лишние линии обрамления,

- На этом этапе желательно выполнить команду **Файл⇒Предварительный Просмотр**, чтобы убедиться, что таблица целиком вмещается на листе по ширине и все линии обрамления на нужном месте.

2-й этап

Заключается в заполнении таблицы, сортировке данных и использовании различных форматов числа.

- Заполните столбцы "Наименование", "Кол-во" и "Цена" по своему усмотрению.
- Установите денежный формат числа в тех ячейках, в которых будут размещены суммы и установите требуемое число десятичных знаков, если они вообще нужны.

В нашем случае это пустые ячейки столбцов "Цена" и "Сумма". Их нужно выделить и выполнить команду **Формат⇒Ячейки...**, выбрать вкладку Число и выбрать категорию Денежный (рис. 2.2). Это даст вам разделение на тысячи, чтобы удобнее было ориентироваться в крупных суммах.

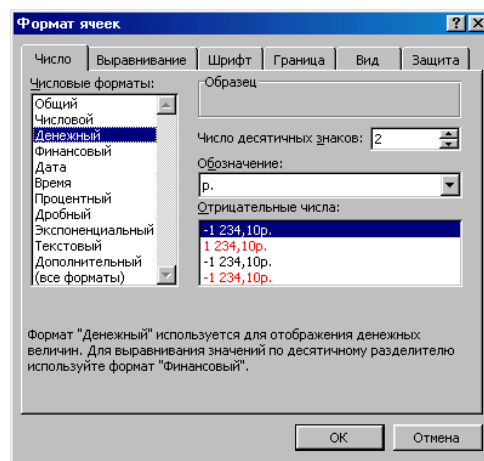



Рис. 2.2.

- Введите формулу для подсчета суммы, которая заключается в умножении цены на количество, и заполните формулой ряд ячеек вниз.
- Введите формулу в ячейку для итоговой суммы- Для этого выделите ячейку, в которую нужно поместить результат, нажмите кнопку  панели инструментов и выделите блок тех ячеек, которые нужно сложить.
- Попробуйте изменить данные в отдельных ячейках и проследите, как изменится результат вычислений.
- Отсортируйте записи по алфавиту.

Для этого выделите все строки таблицы, кроме первой (заголовка) и последней ("Итого"), можно не выделять и нумерацию.

Выполните команду **Данные⇒Сортировка...** (рис. 2.3), выберите столбец, по которому нужно отсортировать данные (в нашем случае это столбец В, так как именно он содержит перечень товаров, подлежащих сортировке), и установите переключатель в положение "По возрастанию".

3-й этап

- Для оформления счета вставьте дополнительные строки перед таблицей.

Для этого выделите несколько первых строк таблицы и выполните команду **Вставка**⇒**Строки**. Вставится столько же строк, сколько вы выделили.

- Наберите необходимый текст до и после таблицы. Следите за выравниванием.

Обратите внимание, что текст "Дата получения" " __ " _____ 200_г." и фамилии руководителей предприятия внесены в тот же столбец, в котором находится столбик таблицы "Сумма" (самый правый столбец нашей таблички), только применено выравнивание вправо.

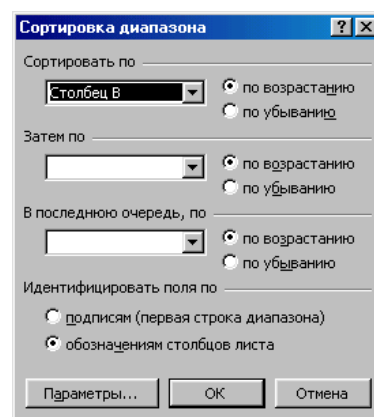


Рис. 2.3.

- Текст "СЧЕТ №" внесен в ячейку самого левого столбца, и применено выравнивание по центру выделения (предварительно выделены ячейки одной строки по всей ширине таблицы счета). Применена рамка для этих ячеек сверху и снизу.
- Вся остальная текстовая информация до и после таблицы внесена в самый левый столбец, выравнивание влево.
- Выполните просмотр.

Упражнение 2

Введение понятия "абсолютная ссылка", установка точного значения ширины столбца при помощи команд горизонтального меню. Вставка функции при помощи мастера функций.

Новое понятие "абсолютная ссылка" можно рассмотреть на конкретном примере. Подготовим традиционную таблицу квадратов двузначных чисел (рис. 2.4), так хорошо знакомую каждому из курса алгебры.

ТАБЛИЦА КВАДРАТОВ										
	0	1	2	3	4	5	6	7	8	9
1	100	121	144	169	196	225	256	289	324	361
2	400	441	484	529	576	625	676	729	784	841
3	900	961	1024	1089	1156	1225	1296	1369	1444	1521
4	1600	1681	1764	1849	1936	2025	2116	2209	2304	2401
5	2500	2601	2704	2809	2916	3025	3136	3249	3364	3481
6	3600	3721	3844	3969	4096	4225	4356	4489	4624	4761
7	4900	5041	5184	5329	5476	5625	5776	5929	6084	6241
8	6400	6561	6724	6889	7056	7225	7396	7569	7744	7921
9	8100	8281	8464	8649	8836	9025	9216	9409	9604	9801

Рис. 2.4

- В ячейку A3 введите число 1, в ячейку A4 - число 2, выделите обе ячейки и протащите маркер выделения вниз, чтобы заполнить столбец числами от 1 до 9.
- Аналогично заполните ячейки B2 - K2 числами от 0 до 9.
- Когда вы заполнили строчку числами от 0 до 9, то все необходимые вам для работы ячейки одновременно не видны на экране. Давайте сузим их, но так, чтобы все столбцы имели одинаковую ширину (чего нельзя добиться, изменяя ширину столбцов мышкой).

Для этого выделите столбцы от A до K и выполните команду **Формат**⇒**Столбец Ширина...**, в поле ввода *Ширина столбца* введите значение, например, 5.

- Разумеется, каждому понятно, что в ячейку B3 нужно поместить формулу, которая возводит в квадрат число, составленное из десятков, указанных в столбце A и единиц, соответствующих значению, размещенному в строке 2. Таким образом, само число, которое должно возводиться в квадрат в ячейке B3 можно задать формулой =A3*10+B2 (число десятков, умноженное на десять плюс число единиц). Остается возвести это число в квадрат.
- Попробуем воспользоваться Мастером функций.

Для этого выделите ячейку, в которой должен разместиться результат вычислений(B3), и выполните команду **Вставка**⇒**функция...**] (рис. 2.5).

Среди предложенных категорий функций выберите "Мат. и тригонометрия". Имя функции: "Степень", нажмите кнопку Шаг.

В следующем диалоговом окне введите число (основание степени) - A3*10+B2 и показатель степени - 2. Так же, как и при наборе формулы непосредственно в ячейке электронной таблицы, нет необходимости вводить адрес каждой ячейки, на которую ссылается формула, с клавиатуры. Работая с Мастером функций, достаточно указать мышью на соответствующую ячейку электронной таблицы, и ее адрес появится в поле ввода "Число" диалогового окна. Вам останется ввести только арифметические знаки (*, +) и число 10.

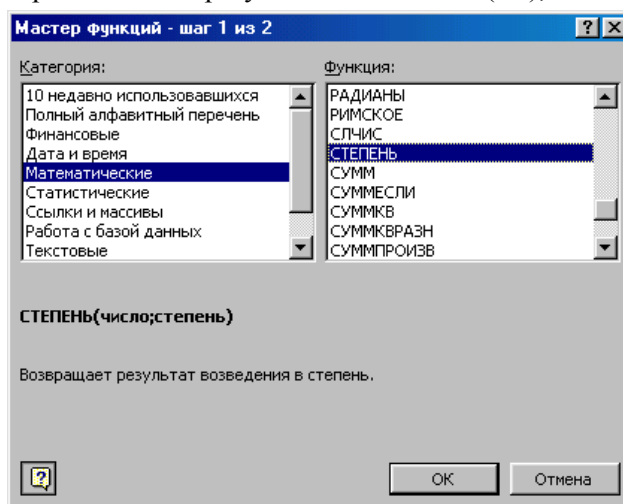


Рис. 2.5.

Если диалоговое окно загромождает нужные ячейки электронной таблицы, переместите его в сторону, "схватив" мышью за заголовок. В этом же диалоговом окне можно увидеть значение самого числа (10) и результат вычисления степени (100).

Остается только нажать кнопку *Закончить*.

В ячейке B3 появился результат вычислений.

- Хотелось бы распространить эту формулу и на остальные ячейки таблицы. Выделите ячейку B3 и заполните, протянув маркер выделения вправо, соседние ячейки. Что произошло (рис. 2.6)?

Почему результат не оправдал наших ожиданий? В ячейке C3 не видно числа, т. к. оно не помещается целиком в ячейку-

Расширьте мышью столбец С. Число появилось на экране, но оно явно не соответствует квадрату числа 11 (рис. 2.7).

	0	1	2	3
1	100	#####	#####	#####

Рис. 2.6

	0	1	2	3
1	100	1002001	#####	#####

Рис. 2.7

Почему? Дело в том, что когда мы распространили формулу вправо. Excel автоматически изменил с учетом нашего смещения адреса ячеек, на которые ссылается формула, и в ячейке С3 возводится в квадрат не число 11, а число, вычисленное по формуле = В3*10+С2.

Во всех предыдущих упражнениях нас вполне устраивали относительные ссылки на ячейки таблицы (при перемещении формулы по такому же закону смещаются и ссылки), однако здесь возникла необходимость зафиксировать определенные ссылки, т. е. указать, что число десятков можно брать только из столбца А, а число единиц только из строки 2 (для того, чтобы формулу можно было распространить вниз). В этом случае применяют абсолютные ссылки.

Для фиксирования любой позиции адреса ячейки перед ней ставят знак \$.

Таким образом, верните ширину столбца С в исходное положение и выполните следующие действия-

- Выделите ячейку В3 и, установив текстовый курсор в Строку формул, исправьте имеющуюся формулу =СТЕПЕНЬ(А3*10+В2;2) на правильную =СТЕПЕНЬ(\$А3*10+В\$2,2).
- Теперь, воспользовавшись услугами маркера заполнения, можно заполнить этой формулой все свободные ячейки таблицы (сначала протянуть маркер заполнения вправо, затем, не снимая выделения с полученного блока ячеек, вниз).
- Осталось оформить таблицу: ввести в ячейку А1 заголовок, сформатировать его и отцентрировать по выделению, выполнить оформление таблицы и заполнение фоном отдельных ячеек.

Упражнение 3

Введение понятия "имя ячейки".

Представьте, что вы имеете собственную фирму по продаже какой-либо продукции и вам ежедневно приходится распечатывать прайс-лист с ценами на товары в зависимости от курса доллара.

- Подготовьте таблицу, состоящую из столбцов: "Наименование товара", "Эквивалент \$ US", "Цена в р.". Заполните все столбцы, кроме "Цена в р." Столбец "Наименование товара" заполните текстовыми данными (перечень товаров по вашему усмотрению), а столбец "Эквивалент \$ US" числами (цены в долл.).
- Понятно, что в столбце "Цена в р." должна разместиться формула: "Эквивалент \$ US"*Курс доллара".

Почему неудобно в этой формуле умножать на конкретное значение курса? Да потому, что при каждом изменении курса, вам придется менять свою формулу в каждой ячейке.

Проще отвести под значение курса доллара отдельную ячейку, на которую и ссылаться в формуле. Ясно, что ссылка должна быть абсолютной, т. е. значение курса доллара можно брать только из этой конкретной ячейки с зафиксированным адресом.

Как задавать абсолютные ссылки, мы рассмотрели выше, однако существует еще один удобный способ: ссылаться не на адрес ячейки, а на имя, которое можно присвоить ячейке.

- Выделите ячейку, в которую будет вводиться курс доллара (выше таблицы), введите в нее значение курса доллара на сегодняшний день и выполните команду **Вставка⇒Имя⇒Присвоить...** (рис. 2.8).

Примечание: Имя может иметь в длину до 255 символов и содержать буквы, цифры, подчеркивания (_), символы: обратная косая черта (\), точки и вопросительные знаки. Однако первый символ должен быть буквой, подчеркиванием (_) или символом обратной косой черты (\). Не допускаются имена, которые воспринимаются как числа или ссылки на ячейки.

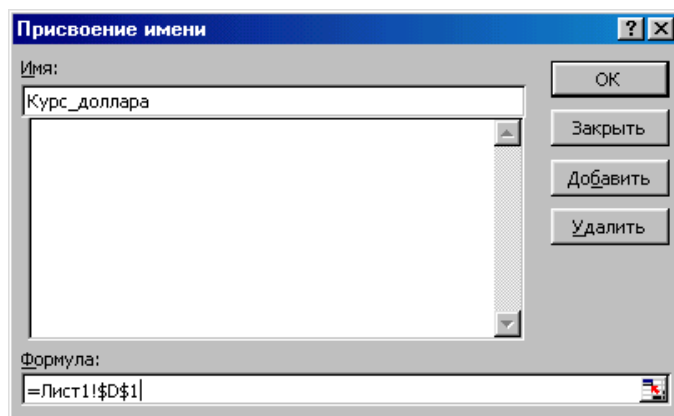


Рис. 2.8.

В появившемся диалоговом окне вам остается только ввести имя ячейки (ее точный адрес уже приведен в поле ввода "Ссылается на") и нажать кнопку ОК. Обратите внимание на то, что в Поле имени, вместо адреса ячейки, теперь размещено ее имя.

- В ячейку, расположенную левее ячейки "Курс_доллара", можно ввести текст "Курс доллара".
- Теперь остается ввести формулу для подсчета цены в рублях.

Для этого выделите самую верхнюю пустую ячейку столбца "Цена в рублях" и введите формулу следующим образом: введите знак "=", затем щелкните мышью по ячейке, расположенной левее (в которой размещена цена в долл.), после этого введите знак "*" и в раскрывающемся списке Поля имени выберите мышью имя ячейки "Курс доллара". Формула должна выглядеть приблизительно так: =B7*Курс_доллара.

- Заполните формулу вниз, воспользовавшись услугами маркера заполнения.
- Выделите соответствующие ячейки и примените к ним денежный формат числа.
- Оформите заголовок таблицы: выровняйте по центру, примените полужирный стиль начертания шрифта, расширьте строку и примените вертикальное выравнивание по центру, воспользовавшись командой **Формат⇒Ячейки...**, выберите вкладку *Выравнивание* и в группе выбора *Вертикальное* выберите *По центру*. В этом же диалоговом окне активизируйте переключатель *Переносить по словам* на случай, если какой-то заголовок не поместится в одну строчку.
- Измените ширину столбцов.
- Выделите таблицу и задайте для нее обрамление. Можно на этом и остановиться, а можно рассмотреть еще несколько дополнительных заданий упражнения.

Лабораторная работа №3.

Создание и заполнение таблиц. Сортировка данных

Изменение ориентации текста в ячейке, ознакомление с возможностями баз данных Excel. Сортировка данных по нескольким ключам. Подготовка документа к печати.

Представьте себя владельцем маленького магазина. Необходимо вести строгий учет прихода и расхода товаров, ежедневно иметь перед глазами реальный остаток, иметь возможность распечатать наименование товаров по отделам и т. д. Даже в таком непростом деле Excel может заметно облегчить работу.

Разобьем данное упражнение на несколько заданий в логической последовательности:

- создание таблицы;
- заполнение таблицы данными традиционным способом и с применением формы;
- подбор данных по определенному признаку.

Создание таблицы

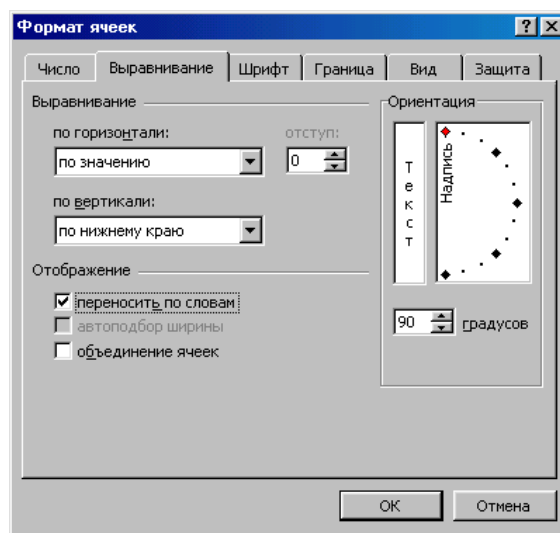
- Введите заголовки таблицы в соответствии с предложенным образцом. Учтите, что заголовок располагается в двух строках таблицы: в верхней строке "Приход", "Расход", "Остаток", а строкой ниже остальные пункты заголовка.

				Приход		Расход		Остаток	
№	Отдел	Наименование товара	Единица измерения	Цена прихода	Кол-во прихода	Цена расхода	Кол-во расхода	Кол-во остатка	Сумма остатка
1									
2									
3									
4									
5									
6									

- Ввод текста заголовка лучше начать со второй строки. Вы уже обратили внимание на то, что графа "Приход" распространяется на две ячейки. Слово "Приход" набрано в том же столбце, что и "Цена прихода", затем выделены две соседние ячейки, и текст отцентрирован по выделению (данная операция неоднократно рассматривалась в предыдущих упражнениях). Аналогично сформатированы ячейки "Расход" и "Остаток".

- Выделите вторую строку заголовка и выровняйте по центру.

- Также видно, что для того, чтобы вся таблица могла разместиться по ширине на листе, в некоторых ячейках текст "повернут на 90°". Выделите те ячейки, в которых нужно "развернуть" текст и выберите



команду **Формат⇒Ячейки...**, на вкладке *Выравнивание* (рис. 3.1) выберите Ориентацию текста и обязательно активизируйте переключатель *Переносить по словам* (вертикальное выравнивание оставьте *По нижнему краю*).

- Для оставшихся (неразвернутых) ячеек примените вертикальное выравнивание *По центру* **Формат⇒Ячейки...**

- Задайте оформление таблицы (**Формат⇒Ячейки...**, вкладка *Рамка*).
- Установите в ячейках, содержащих цены денежный формат числа(**Формат⇒Ячейки...**, вкладка *Число*).

- Введите нумерацию строк таблицы (столбец №), при помощи маркера заполнения.
- Вставьте формулы для количества остатка ("Кол-во Прихода" минус "Кол-во Расхода") и суммы остатка ("Кол-во Остатка" умножить на "Цену Расхода"). Распространите эти формулы вниз по таблице.

В процессе выполнения задания во многих случаях удобнее пользоваться контекстным меню, вызываемым нажатием правой клавиши мыши.

Так, для форматирования ячеек их достаточно выделить, щелкнуть правой клавишей мыши в тот момент, когда указатель мыши находится внутри выделения и выбрать команду **Формат⇒Ячеек...**, вы перейдете к тому же диалоговому окну Формат ячеек (рис. 3.1). Да и редактировать содержимое ячейки (исправлять, изменять данные) совсем не обязательно в Строке формул. Если дважды щелкнуть мышью по ячейке, в ней появится текстовый курсор, и можно произвести все необходимые исправления.

Заполнение таблицы

- Определитесь, каким видом товаров вы собираетесь торговать и какие отделы будут в вашем магазине.

Вносите данные в таблицу не по отделам, а вперемешку (в порядке поступления товаров).

Заполните все ячейки, кроме тех, которые содержат формулы ("Остаток").

Обязательно оставьте последнюю строку таблицы пустой (но эта строка должна содержать все формулы и нумерацию).

Вводите данные таким образом, чтобы встречались разные товары из одного отдела (но не подряд) и обязательно присутствовали товары с нулевым остатком (все продано).

				Приход		Расход		Остаток	
№	Отдел	Наименование товара	Единица измерения	Цена прихода	Кол-во прихода	Цена расхода	Кол-во расхода	Кол-во остатка	Сумма остатка
1	Кондитерский	Зефир в шоколаде	упак.	20 р.	15	25р.	15	0	0 р.
2	Молочный	Сыр	кг.	65 р.	10	85 р.	8	2	170 р.
3	Мясной	Колбаса Московская	кг.	110 р.	20	120р.	15	5	600 р.
4	Мясной	Балык	кг.	120 р.	10	140 р.	5	5	700 р.

5	Вино-водочный	Водка «Абсолют»	бут. 2 л.	400 р.	100	450 р.	99	1	450 р.
6									0 р.

- Согласитесь, что традиционный способ заполнения таблицы не особенно удобен. Воспользуемся возможностями баз данных Excel.

- Выделите таблицу без верхней строки заголовка (так как Excel определяет поля базы данных по одной первой строке выделенной таблицы), но обязательно захватите нижнюю незаполненную строку (иначе вы не сможете продолжить заполнение таблицы в режиме формы).

- Выберите команду **Данные**⇒**Форма...** Вы получите форму данных (рис. 3.2), содержащую статический текст (имена полей базы данных) и окон редактирования, в которых можно вводить и редактировать текст.

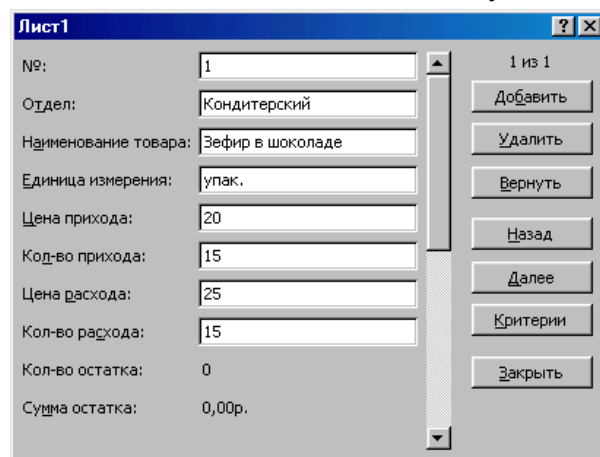


Рис. 3.2.

Вычисляемые поля (в которых размещены формулы) выводятся на экран без окон редактирования ("Кол-во Остатка" и "Сумма Остатка").

Теперь вы имеете свою таблицу как бы в форме отдельных карточек-записей (каждая из которых представляет строку таблицы).

Перемещаться между записями можно либо при помощи кнопок "Предыдущая", "Следующая", либо клавишами управления курсором (вверх, вниз), либо перемещая бегунок на полосе прокрутки формы данных.

- Дойдя до последней записи (мы специально оставили ее пустой, но распространили на нее формулы и нумерацию), заполните ее новыми данными.

Перемещаться между окнами редактирования (в которые вносятся данные) удобно клавишей (Tab).

Когда заполните всю запись, нажмите клавишу {Enter}, и вы автоматически перейдете к новой чистой карточке-записи

- Как только вы заполните новую запись, вся внесенная вами информация автоматически воспроизведется и исходной таблице.

Заполните несколько новых записей и затем нажмите кнопку **Заккрыть**.

Как видно, заполнять таблицу в режиме формы довольно удобно.

Оперирование данными

Итак, вы заполняли таблицу в порядке поступления товаров, а хотелось бы иметь список товаров по отделам, для этого применим сортировку строк.

Выделите таблицу без заголовка и выберите команду **Данные**⇒**Сортировка...** (Рис. 3.3).

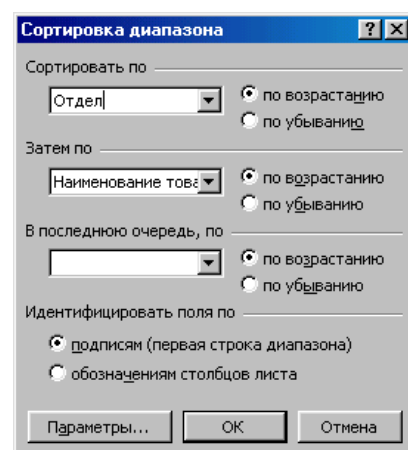


Рис. 3.3.

- Выберите первый ключ сортировки: в раскрывающемся списке "Сортировать" выберите "Отдел"⁵ и установите переключатель в положение "По возрастанию" (все отделы в таблице расположатся по алфавиту).

Если же вы хотите, чтобы внутри отдела все товары размещались по алфавиту, то выберите второй ключ сортировки: в раскрывающемся списке "Затем по" выберите "Наименование товара", установите переключатель в положение "По возрастанию". Теперь вы имеете полный список товаров по отделам.

Продолжим знакомство с возможностями баз данных Excel.

Вспомним, что нам ежедневно нужно распечатывать список товаров, оставшихся в магазине (имеющих ненулевой остаток), но для этого сначала нужно получить такой список, т. е. отфильтровать данные.

- Выделите таблицу со второй строкой заголовка (как перед созданием формы данных).
- Выберите команду меню **Данные**⇒**Фильтр...** ⇒**Автофильтр**.
- Снимите выделение с таблицы.
- У каждой ячейки заголовка таблицы появилась кнопка (она не выводится на печать), позволяющая задать критерий фильтра. Мы хотим оставить все записи с ненулевым остатком.
 - Раскройте список ячейки "Кол-во Остатка", выберите команду **Настройка...** и, в появившемся диалоговом окне установите соответствующие параметры (>0).
 - Вместо полного списка товаров, вы получили список непроданных на сегодняшний день товаров. Можно известным нам способом вставить формулу подсчета общей суммы остатка (в режиме фильтра будет подсчитана сумма выведенных на экран данных). Эту таблицу можно распечатать.
 - Фильтр можно усилить. Если дополнительно выбрать какой-нибудь конкретный отдел, то можно получить список непроданных товаров по отделу.
 - Для того, чтобы снова увидеть перечень всех непроданных товаров по всем отделам, нужно в списке Отдел выбрать критерий Все.
 - Но и это еще не все возможности баз данных Excel. Разумеется ежедневно нет необходимости распечатывать все сведения о непроданных товарах, нас интересует только "Отдел", "Наименование" и "Кол-во Остатка".

Можно временно скрыть остальные столбцы. Для этого выделите столбец №, вызовите контекстное меню (правой клавишей мыши в тот момент, когда указатель мыши находится внутри выделения) и выберите команду **Скрыть**.

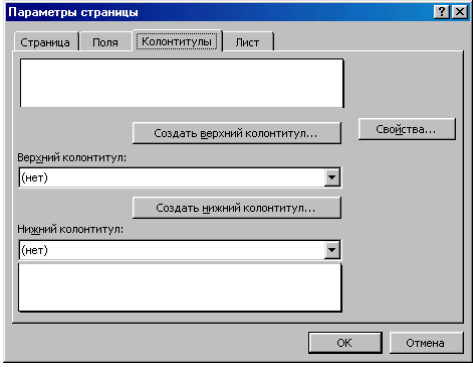
Таким же образом можно скрыть и остальные столбцы, связанные с приходом, расходом и суммой остатка.

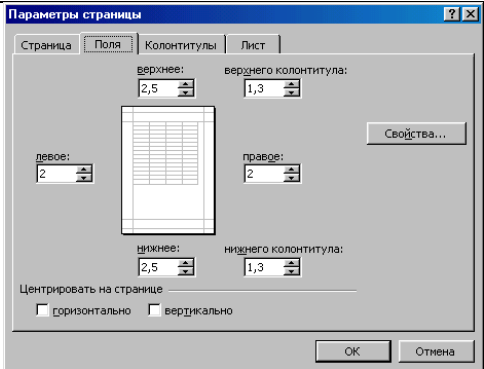
Вместо команды контекстного меню можно воспользоваться командой горизонтального меню **Формат**⇒**Столбец**⇒**Скрыть**.

- Чтобы не запутаться в своих распечатках вставьте дату, которая автоматически будет изменяться в соответствии с установленным на вашем компьютере временем **Вставка**⇒**Функция...**, имя функции — "Сегодня").
 - Теперь уже точно можно распечатать и иметь подшивку ежедневных сведений о наличии товара.
 - Как вернуть скрытые столбцы? Проще всего выделить таблицу **Формат**⇒**Столбец**⇒**Показать**.

- Для того, чтобы восстановить все данные (воспроизвести таблицу в полном виде с перечнем всех товаров), достаточно убрать отметку команды **Автофильтр** (команда **Фильтр...** меню **Данные**).

Прежде чем напечатать любой документ, выполните просмотр (**Файл**⇒**Предварительный просмотр** или воспользуйтесь кнопкой Предварительный просмотр панели инструментов). Вам может не понравиться несколько моментов:

<p>В верхней части листа появилась запись "Лист I". Нужно ее удалить.</p>	<ul style="list-style-type: none"> • Находясь в режиме просмотра, выберите кнопку <i>Страница...</i>; • в появившемся диалоговом окне выберите вкладку <i>Колонтитулы</i>; • в поле выбора <i>Верхние колонтитулы</i> установите <i>Нет</i> (можно выбрать в раскрывающемся списке, в случае необходимости воспользовавшись полосами прокрутки). 
<p>В нижней части листа появилась запись "СТР. I". Нужно ее удалить.</p>	<ul style="list-style-type: none"> • Находясь в режиме просмотра, выберите кнопку <i>Страница...</i>; • в появившемся диалоговом окне выберите вкладку <i>Колонтитулы</i>; <p>в поле выбора <i>Нижние колонтитулы</i> установите <i>Нет</i> (можно выбрать в раскрывающемся списке, в случае необходимости воспользовавшись полосами прокрутки).</p>
<p>На просмотре выводится разметочная сетка, ограничивающая те ячейки, для которых не задано оформление. Нужно удалить сетку.</p>	<p>Находясь в режиме просмотра, выберите кнопку <i>Страница...</i>, в появившемся диалоговом окне вкладку <i>Лист</i> и отключите переключатель <i>Печатать сетку</i>.</p>
<p>Таблица не помещается по ширине на странице, хотелось бы уменьшить левое и правое поля.</p>	<ol style="list-style-type: none"> 1. Находясь в режиме просмотра, выберите кнопку <i>Страница...</i>, в появившемся диалоговом окне вкладку <i>Поля</i> и установите желаемые поля. 2. Находясь в режиме просмотра, выберите кнопку <i>Поля</i>, появятся ограничители полей, которые можно перемещать мышью.

	
<p>Размер полей уменьшен, а таблица так и не помещается по ширине на странице. Хотелось бы изменить ориентацию листа.</p>	<p>Находясь в режиме просмотра, выберите кнопку <i>Страница...</i>, в появившемся диалоговом окне вкладку <i>Страница</i> и измените ориентацию листа на Альбомная. Здесь же можно задать размер бумаги.</p>

Диалоговое окно <Параметры страницы> можно вызвать, находясь в режиме таблицы (не выходя в режим просмотра), выполнив команду **Файл**⇒**Параметры страницы...**

Лабораторная работа №4.

Управление листами книги. Мастер диаграмм

Проверка уровня сформированности основных навыков работы с электронными таблицами. Знакомство с общими сведениями об управлении листами рабочей книги, удалении, переименовании листов. формулы, имеющие ссылки на ячейки другого листа рабочей книги. Мастер диаграмм. Выделение ячеек таблицы, не являющихся соседними.

Подготовим ведомость на выдачу заработной платы (естественно, несколько упрощенный вариант).

Обратите внимание на то, что в нижней части экрана горизонтальная полоса прокрутки состоит из двух частей. Правая часть служит для перемещения по таблице (вправо, влево), а левая часть, содержащая ярлычки листов, позволяет перемещаться между листами.

По умолчанию рабочая книга открывается с 16-ю рабочими листами, имена которых Лист1, ..., Лист16. Имена листов выведены на ярлычках в нижней части окна рабочей книги.

Щелкая по ярлычкам, можно переходить от листа к листу внутри рабочей книги.

Ярлычок активного листа выделяется цветом, к надписи на нем применен полужирный стиль.

Для выбора конкретного листа, достаточно щелкнуть по его ярлычку мышью.

Для выполнения упражнения нам понадобятся только четыре листа:

- на первом разместим сведения о начислениях,

- на втором - диаграмму,
- на третьем — ведомость на выдачу заработной платы,
- а на четвертом — ведомость на выдачу компенсаций на детей.

Остальные листы будут только мешать, поэтому их лучше удалить.

• Выделите листы с 5 по 16. Для этого щелкните мышью по ярлычку листа 5, затем, воспользовавшись кнопкой перейдите к ярлычку листа 16 и, удерживая клавишу (Shift), щелкните по нему мышью. Ярлычки листов с 5 по 16 выделяются цветом.

• Удалите выделенные листы, вызвав команду контекстного меню Удалить или воспользовавшись командой горизонтального меню **Правка⇒Удалить лист**.

Теперь выглядят ярлычки только четырех листов.

Активен (ярлычок выделен цветом) Лист 1. Именно на нем мы и начнем создавать таблицу.

Создание таблицы

Создайте заготовки таблицы самостоятельно, применяя следующие операции:

- запуск Excel;
- форматирование строки заголовка. Заголовок размещен в двух строках таблицы, применен полужирный стиль начертания шрифта, весь текст выровнен по центру, а "Налоги" — по центру выделения;
- изменение ширины столбца (в зависимости от объема вводимой информации);
- обрамление таблицы. В данном случае использовано сложное обрамление, когда снята часть рамок. Важно по предложенному образцу определить реальное положение ячеек и выполнить соответствующее обрамление, выделяя различные блоки ячеек;
- задание формата числа "денежный" для ячеек, содержащих суммы. Можно сделать это до ввода данных в таблицу (выделить соответствующие ячейки и установить для них формат числа "денежный");
- заполнение ячеек столбца последовательностью чисел 1, 2, ...;
- ввод формулы в верхнюю ячейку столбца;
- распространение формулы вниз по столбцу и в некоторых случаях вправо по ряду;
- заполнение таблицы текстовой и фиксированной числовой информацией (столбцы "ФИО", "Оклад", "Число детей");
- сортировка строк (сначала отсортировать по фамилиям по алфавиту, затем отсортировать по суммам).

№	Фамилия, имя отчество	Оклад	Налоги			Сумма к	Чис ло
			профс	пенс.	подох.		
1							
2							
3							
4							
5							
6							
7							
8							

Для форматирования формул вам наверняка понадобится дополнительная информация. Примем профсоюзный и пенсионный налоги, составляющими по 1% от оклада. Удобно ввести формулу в одну ячейку, а затем распространить ее на оба столбца. Самое важное не забыть про абсолютные ссылки, так как и профсоюзный и пенсионный налоги нужно брать от оклада, т. е. ссылаться только на столбец "Оклад". Примерный вид формулы:

=C3*1 % или =C3*0,01 или =C3*1/100. После ввода формулы в ячейку D3 ее нужно распространить вниз (протянув за маркер выделения) и затем вправо на один столбец.

Подоходный налог подсчитаем по формуле: 12% от Оклада за вычетом минимальной заработной платы и пенсионного налога. Примерный вид формулы: =(C3-E3-86)*12% или =(C3-E3-86)*12/100 или =(C3-E3-86)*0,12. После ввода формулы в ячейку F3, ее нужно распространить вниз.

Для подсчета Суммы к выдаче примените формулу, вычисляющую разность оклада и налогов. Примерный вид формулы: =C3-D3-E3-F3, размещенной в ячейке G3 и распространенной вниз.

Заполняйте столбцы "Фамилия, имя, отчество", "Оклад", и "Число детей" после того, как введены все формулы. Результат будет вычисляться сразу же после ввода данных в ячейку. При желании можно воспользоваться режимом формы для заполнения таблицы.

После ввода всех данных желательно выполнить их сортировку (не забудьте перед сортировкой выделить все строки от фамилий до сведений о детях).

В окончательном виде таблица будет соответствовать образцу:

№	Фамилия, имя отчество	Оклад	Налоги			Сумма к выдаче	Число детей
			профс.	пенс.	подох.		
1	Иванов А-Ф.	230000	2300	2300	18216	207184	1
2	Иванова Е.П.	450 000	4500	4500	44352	396 648	2
3	Китов а В. К	430 000	4300	4300	41 976	379 424	0
4	Котов И.П	378000	3780	3780	35 798	334642	0
5	Кругло ва АД	230000	2300	2300	18 216	207184	1
6	Леонов И И	560 000	5600	5600	57 420	491 380	3
7	Петров М.В.	348 000	3490	3490	32353	309667	1
8	Сидоров И.В.	450000	4500	4500	44352	396 648	1
9	Симонов К.Е	349 000	3490	3490	32 353	309667	0
10	Храмов А.К	430 000	4300	4300	41 976	379 424	0
11	Чудов АН,	673 000	6730	6730	70844	588 696	2

Можно ввести строку для подсчета общей суммы начислений и на этом закончить проверочную работу и приступить к совместным действиям.

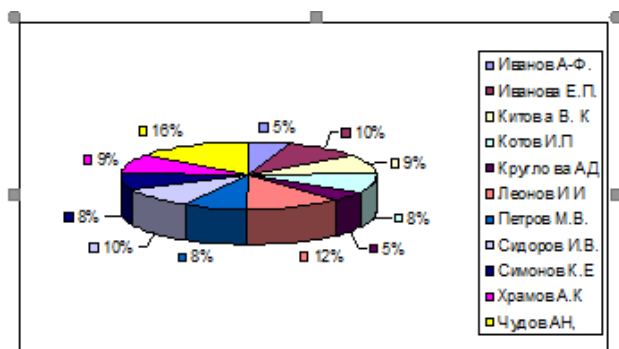
Поскольку мы собираемся в дальнейшем работать сразу с несколькими листами, имеет смысл переименовать их ярлычки в соответствии с содержимым. Переименуем активный в настоящий момент лист. Для этого выполните команду **Формат⇒Лист⇒Переименовать...** и в поле ввода *Имя листа* введите новое название листа, например, "Начисления".

Построение диаграммы на основе готовой таблицы и размещение ее на новом листе рабочей книги

Построим диаграмму, отражающую начисления каждого сотрудника. Понятно, что требуется выделить два столбца таблицы: "Фамилия, имя, отчество" и "Сумма к выдаче". Но эти столбцы не расположены рядом, и традиционным способом мы не сможем их выделить. Для Excel это не проблема.

Если удерживать нажатой клавишу (Ctrl), то можно одновременно выделять ячейки в разных местах таблицы.

- Выделите заполненные данными ячейки таблицы, относящиеся к столбцам "Фамилия, имя, отчество" и "Сумма к выдаче".
- Запустите Мастер диаграмм одним из способов: либо выбрав кнопку Мастер диаграмм панели инструментов, либо команду меню **Вставка**⇒**Диаграмма....**
- Передвигаясь по шагам с Мастером диаграмм, выберите тип диаграммы — объемная круговая, подтип седьмой (с метками данных). Приблизительный вид приведен на рисунке.



- Переименуйте лист, содержащий диаграмму из "Лист1" в "Диаграмма".
- Для того, чтобы проверить, какая связь существует между таблицей начислений и диаграммой, перейдите на лист "Начисления", в середину таблицы вставьте новую строку (выделите строку таблицы и выполните команду **Вставка**⇒**Строки**). Распространите на новую строку формулы, заполните данные на нового сотрудника. Теперь перейдите на лист "Диаграмма" и проверьте, как новые данные отразились на диаграмме - новый сотрудник сразу же внесен в диаграмму.

Создание ведомости на получение компенсации на детей на основе таблицы начислений. Ссылки на ячейки другого листа рабочей книги

- Перейдите к Листу 3. Сразу же переименуйте его в "Детские".

ФИО	Сумма	Подпись
Иванов А.Ф.	53 130	
Иванова Е.П.	106260	
Кругло ва А.Д.	53130	
Леонов И.И.	159390	
Петров М.В.	53 130	
Сидоров И.В.	53 130	
Чудов А.Н.	106260	

- Мы хотим подготовить ведомость, поэтому в ней будут три столбца: "ФИО", "Сумма" и "Подпись". Сформируйте заголовки таблицы.
- В графу "ФИО" нужно поместить список сотрудников, который мы имеем на листе

"Начисления". Можно скопировать на одном листе и вставить на другой, но хотелось бы установить связь между листами (как это выполняется для диаграммы и листа начислений). Для этого на листе "Детские" поместим формулу, по которой данные будут вставляться из листа "Начисления".

- Выделите ячейку A2 листа "Детские" и введите формулу: =Начисления!ВЗ, где имя листа определяется восклицательным знаком, а ВЗ - адрес ячейки, в которой размещена первая фамилия сотрудника на листе "Начисления". Можно набрать формулу с клавиатуры, а можно после набора знака равенства перейти на лист "Начисления", выделить ячейку, содержащую первую фамилию и нажать (Enter) (не возвращаясь к листу "Детские").

- Перейдите на лист "Детские", проверьте полученную формулу и распространите ее вниз. Список фамилий сотрудников теперь есть и на листе "Детские". Больше того, если внести новые данные в таблицу начислений, они отразятся и на листе "Детские". (Нужно будет только распространить формулу ниже в случае необходимости.)

- В графе "Сумма" аналогичным образом нужно разместить формулу =Начисления!НЗ*53130, где НЗ адрес первой ячейки на листе "Начисления", содержащей число детей. Заполните эту формулу вниз и примените денежный формат числа.

- Выполните оформление таблицы.

- Для того, чтобы список состоял только из сотрудников, имеющих детей, установите фильтр по наличию детей (**Данные⇒фильтр⇒Автофильтр**, в раскрывающемся списке "Сумма" выберите "Настройка..." и установите критерий >0). Приблизительный вид ведомости приведен ниже.

- Осталось поместить выше таблицы заголовки ведомости (для этого может понадобиться вставка дополнительных строк) и можно распечатывать.

-

Лабораторная работа №5.

Создание шаблона и работа с ним. Совместное использование

Word и Excel

Упражнение 1

Создание шаблона. Работа с шаблонами документов. Совместное использование Word и Excel.

Представьте себя работником Отдела кадров, которому ежемесячно предстоит заполнять Табель учета рабочего времени на сотрудников предприятия. Разумеется, хотелось бы максимально автоматизировать эту операцию. Удобно создать шаблон заготовки бланка и применить специальные функции.

Создание бланка-шаблона

1. Оставьте в рабочей книге только один лист.

2. Сформируйте заголовок табеля учета рабочего времени за текущий месяц и подготовьте таблицу-бланк по

Фамилия, и.о.	Профессия	Разряд	Числа			Месяца			Дни неявок			Отработано часов	
			1	2	3	29	30	31	Дни явок	отпуск	болезнь		прогул

Рис.1

образцу, приведенному на рис. 1

Воспользуйтесь всеми известными вам приемами форматирования. Сформируйте заголовок, применив различные способы выравнивания текста.

Введите числа месяца с 1-го по 31-е. Для столбцов, содержащих даты, установите ширину столбца, равную 2.

Если на вашем предприятии постоянный состав сотрудников, внесите в шаблон фамилии и профессии.

3. Для сохранения подготовленного файла в качестве шаблона:

- выполните команду **Сохранить как...** меню **Файл**;
- введите имя сохраняемого файла в поле ввода *Имя файла*: Табель;
- в списке типов файлов выберите **Шаблон**, расширение файла сменится на .xlt;
- нажмите ОК;
- закройте файл.

Применение шаблона

Для создания нового файла с применением шаблона выполните следующие действия:

- В меню **Файл** выберите **Создать**.
- В списке *Общие* диалогового окна <Создание документа> выделите шаблон, на основе которого хотите создать новую рабочую книгу (рис.2).
- Выберите кнопку ОК.

Таким образом, вы получите рабочую копию шаблона.

1. Введите название текущего месяца в заголовок табеля.
2. Сразу же выделите цветом столбцы, соответствующие нерабочим дням недели (чтобы случайно не ошибиться при заполнении табеля).

3. Проставьте для каждого сотрудника:

- количество часов, отработанных за день, или
- **о**, если он находится в отпуске, или
- **б**, если в этот день сотрудник болеет, или
- **п**, если прогуливает.

о, б, п — русские буквы, проставляются без кавычек.

Имея такую широкую таблицу, как ваша, можно столкнуться с неудобствами при заполнении. Дело в том, что, перемещаясь вправо для заполнения таблицы, вы теряете из вида столбец с фамилиями и становится трудно определить, кому из сотрудников проставляете рабочие часы.

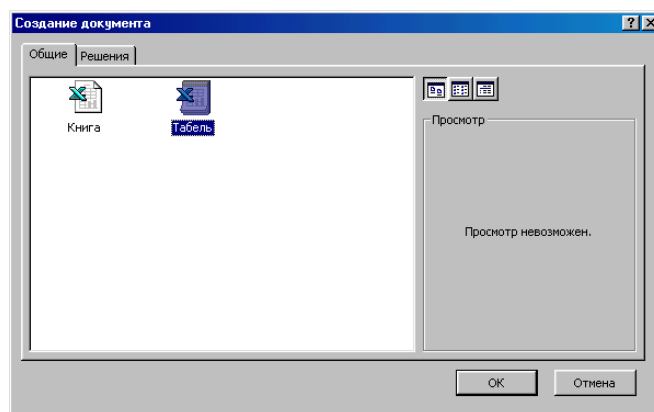


Рис. 2

Помните, в Microsoft Word существовала возможность зафиксировать заголовок таблицы, чтобы он автоматически появлялся на каждой новой странице?

Microsoft Excel позволяет зафиксировать заголовок на странице, чтобы при перемещении нужные вам столбцы (или строки) оставались на своем месте. Для того, чтобы зафиксировать столбец "Фамилия":

- выделите столбец справа от столбца "Фамилия" ("Профессия");
- в меню **Окно** выберите команду **Закрепить области**;
- далее свободно пользуйтесь горизонтальной полосой прокрутки, фамилии ваших сотрудников не исчезнут с экрана.

Работая с большими таблицами, можете пользоваться следующими возможностями фиксации заголовков.

- Чтобы зафиксировать горизонтальные заголовки, выделите строку ниже заголовков.
- Чтобы зафиксировать вертикальные заголовки, выделите столбец справа от заголовков.
- Чтобы зафиксировать и вертикальные, и горизонтальные заголовки выделите ячейку, по которой хотите зафиксировать заголовки.

В меню **Окно** выберите команду **Закрепить области**. Все строки выше выделенной строки (ячейки) будут зафиксированы и все столбцы слева от выделенного столбца (ячейки) будут зафиксированы.

Чтобы отменить фиксацию заголовков в меню **Окно** выберите команду **Снять закрепление областей**.

Хотелось бы ввести формулы для подсчета дней явок, неявок и отработанных часов.

4. Самостоятельно вставьте формулу суммирования соответствующих ячеек строки для подсчета отработанных часов. Заполните формулу вниз.

5. Для подсчета дней явок необходимо в каждой строке (для каждого сотрудника) подсчитать количество ячеек, содержащих числа (не суммируя эти числа). Для этого:

- выделите ячейку таблицы, в которую нужно разместить формулу (для первого сотрудника);
- выполните команду **Вставка**⇒ **Функция...**;
- в списке *Имя функции* окна диалога <Мастер функций> выберите функцию СЧЕТ (рис. 3). Если вы не знаете, к какой категории относится искомая функция, выберите категорию *Полный алфавитный перечень* и дальше ищите по алфавиту. Нажмите кнопку *Ок*.
- В следующем окне нужно указать диапазон значений.

Нет необходимости вводить адреса ячеек с клавиатуры.

Отодвиньте окно диалога, чтобы оно не загромождало таблицу, и выделите мышью интервал ячеек, в которых размещена информация о первом сотруднике.

Нажмите кнопку *Ок*.

- Заполните формулу вниз.

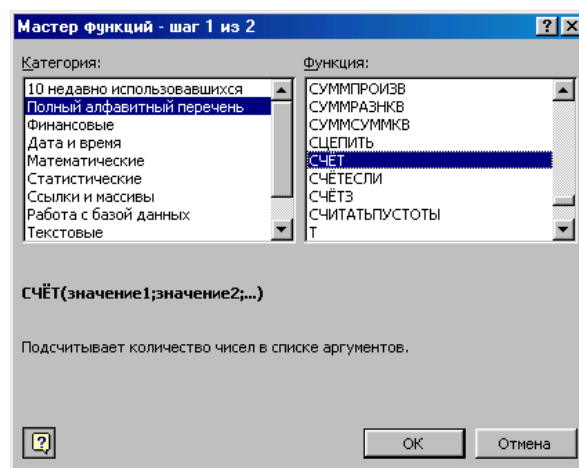


Рис. 3

б. Для подсчета количества дней, проведенных в отпуске, вставьте функцию СЧЕТЕСЛИ и, в качестве критерия введите образец (что нужно подсчитывать) русскую букву **о**, т. е. тот символ, который вы вносили в таблицу, отмечая отпуск.

Заполните формулу вниз по столбцу.

В результате вы получите приблизительно следующее.

20	23	24	25	26	27	30	Дни явок				
							Дни явок	отпуск	болезнь	пропуск	Отработано часов
8	8	8	8	8	8	8	22	0			176
8	8	8	8	8	8	8	22	0			176
4	4	4	4	4	4	4	22	0			88
8	6	6	6	6	6	6	16	0			128
8	0	0	0	0	0	8	17	5			136
8	8	8	8		8	8	21	0			168

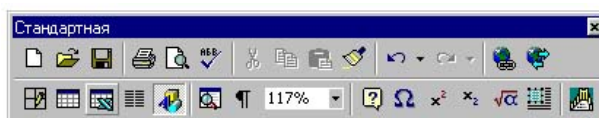
Упражнение 2

Совместное использование Word и Excel.

Microsoft Excel — это мощный инструмент анализа данных, позволяющий создавать электронные таблицы, диаграммы и другие формы представления информации. В свою очередь, Microsoft Word, как вы уже знаете, — это мощный инструмент для создания профессионально выглядящих документов. В этой работе вы узнаете, как Word и Excel могут работать вместе и какие возможности предоставляет это сотрудничество.

Использование кнопок Excel

Панели инструментов Word содержат две кнопки для работы с Excel: одна на стандартной панели инструментов и другая — на панели инструментов Microsoft, как показано ниже. Чтобы вывести на экран панель инструментов Microsoft, выберите команду **Вид**⇒**Панели инструментов** и установите флажок Microsoft, после чего щелкните по ОК.



Добавить таблицу Excel



Microsoft Excel

Обратите внимание, что кнопка Microsoft Excel на стандартной панели инструментов содержит изображение электронной таблицы, на фоне которой расположен значок Excel, в то время как изображение на кнопке панели инструментов Microsoft состоит только из значка Excel.

Кроме того, обратите внимание, что всплывающие подсказки для этих двух кнопок также различаются, как отличаются и пояснения, выдаваемые в строке состояния при выборе одной из этих двух кнопок.

Функции этих двух кнопок кратко можно описать следующим образом:

- Кнопка **Добавить таблицу Excel** на стандартной панели инструментов приводит к внедрению в документ Word электронной таблицы — то есть при этом вы сможете редактировать электронную таблицу Excel прямо в документе Word.
- Кнопка **Microsoft Excel** на панели инструментов Microsoft приводит к связыванию электронной таблицы или вставке базы данных из Excel; щелчок по этой кнопке приводит к запуску Excel или (если он уже запущен) переключению в окно Excel.

Обмен информацией с Excel

Информация из книги Microsoft Excel может копироваться, внедряться, связываться или извлекаться в зависимости от ваших потребностей и того, какова будет дальнейшая судьба документа Word и информации из Excel. Выбирая один из этих четырех способов использования информации Excel, имейте в виду следующее:

- Копировать информацию из существующей книги Excel и вставлять ее в документ Word имеет смысл в том случае, если вы не собираетесь изменять информацию в этой книге или если вы не хотите, чтобы информация в документе Word отражала будущие изменения в документе Excel. При этом информация копируется в документ Word в виде таблицы Word или графического изображения Word.
- Внедрение информации допустимо в том случае, если используемый документ Excel относится только к данному документу Word и никогда не понадобится ни в других документах Word, ни для каких-либо иных нужд, и если при этом информация, скорее всего, будет в будущем меняться. Внедрять в документ Word можно как существующую электронную таблицу, так и заново созданную. При этом информация электронной таблицы хранится в документе Word.
- Связывание информации удобно в тех случаях, когда используемый документ Excel будет нужен либо в самом Excel, либо в других документах и приложениях (в частности, в других документах Word). Все изменения, которые вносятся при этом в исходную электронную таблицу, будут отображаться во всех документах, связанных с этой таблицей (в том числе и в документах Word). Прежде чем вы сможете связать документ Word с электронной таблицей, эта таблица должна существовать, то есть ей должно быть присвоено имя и она должна быть сохранена в файле. При этом информация электронной таблицы хранится в файле Excel.
- Извлечение информации применяется в тех случаях, когда вам нужна только часть информации из существующей книги Excel, выбранная по каким-либо критериям. При этом, если извлеченная информация вставлена в документ Word в виде поля, то любые изменения в книге Excel, относящиеся к извлеченным данным, будут отображаться в документе Word. Извлеченная информация хранится в документе Word, а исходная информация книги Excel хранится в файле Excel.

Использование ячеек таблицы Excel

Любое количество ячеек из электронной таблицы Excel можно скопировать в документ Word с помощью операций вставки, внедрения или связывания.

Вставка ячеек

Чтобы вставить в документ Word ячейки электронной таблицы Excel, поступайте следующим образом:

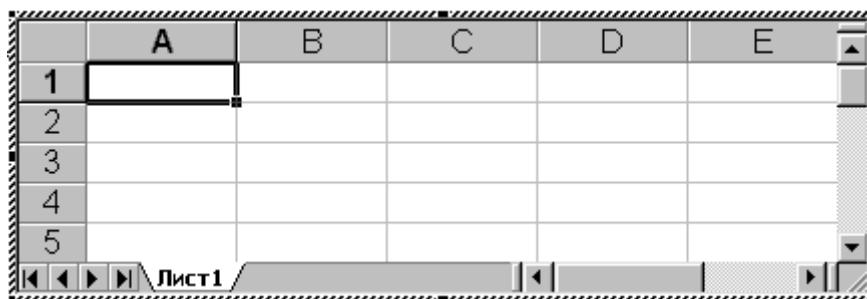
1. Щелкните по кнопке Microsoft Excel на панели инструментов Microsoft, чтобы запустить Excel.
2. Либо откройте одну из существующих книг, либо введите нужное содержимое в новую таблицу.
3. Выделите ячейки, которые вы хотите скопировать в документ Word, и выберите команду **Правка** ⇒ **Копировать**.
4. Переключитесь в документ Word, поместите курсор вставки в том месте, где вы хотите вставить ячейки, и выберите команду **Правка** ⇒ **Вставить**. С помощью команды **Правка** ⇒ **Специальная вставка** вы можете также вставить форматированное содержимое ячеек в документ Word.

После вставки содержимое ячеек будет оформлено в виде таблицы Word и не будет иметь никакой связи с Excel или файлом, созданным в Excel.

Внедрение ячеек

Чтобы внедрить ячейки таблицы Excel в документ Word, поступайте следующим образом:

1. Щелкните по кнопке Добавить таблицу Excel на стандартной панели инструментов и протаскиванием в появившейся сетке укажите число строк и столбцов, которые вы хотите вставить в документ Word (точно так же, как при использовании кнопки Вставить таблицу). После этого вы увидите в своем документе объект Excel, который выглядит так:



2. В ячейках созданной таблицы вводите текст, числа и формулы, которые вы хотите использовать.
3. Щелкните в документе Word за пределами таблицы, чтобы вернуться к работе с документом. Тех же самых результатов можно добиться, выбрав команду **Вставка** ⇒ **Объект**, указав вкладку Создание, выбрав из списка Тип объекта пункт Лист Microsoft Excel и щелкнув по ОК.

Связывание ячеек

Чтобы связать ячейки книги Excel с документом Word, поступайте так:

1. Щелкните по кнопке Microsoft Excel на панели инструментов Microsoft, чтобы запустить Excel.
2. Либо откройте одну из существующих книг, либо введите нужное содержимое в новую таблицу. Если вы создаете новую таблицу, не забудьте потом сохранить ее.

3. Выделите ячейки, которые вы хотите связать с документом Word, и выберите команду **Правка⇒Копировать**.
4. Переключитесь в документ Word и поместите курсор вставки в том месте, где вы хотите расположить связываемые ячейки.
5. Выберите команду **Правка⇒Специальная вставка**.
6. В диалоговом окне *Специальная вставка* установите опцию **Форматированный текст (RTF)**. Установите флажок **Связать** и щелкните по **ОК**.

После этого вставленные ячейки сохранят связь с Excel. Содержимое этих ячеек будет храниться в файле Excel.

Использование диаграмм Excel

Вставка диаграммы Excel в документ Word осуществляется теми же методами, что и вставка ячеек таблицы. Для этого вы можете использовать как обычную вставку через буфер, так и связывание или внедрение диаграммы Microsoft Excel.

Самостоятельно создайте в Excel диаграмму и выполните вставку и внедрение диаграммы в Word.

Программа Adobe Photoshop.

Лабораторная работа №1. Текст, охваченный огнём

Из этого подробного урока, вы узнаете, как создать текст, охваченный реалистичным огнём.

Для этого мы будем использовать:

фильтр Ветер (Wind filter) - для создания пламя;

фильтр Пластика (Liquify filter) - для корректировки пламя;

Корректирующие слои (adjustment layers) - для придания цвета пламени;

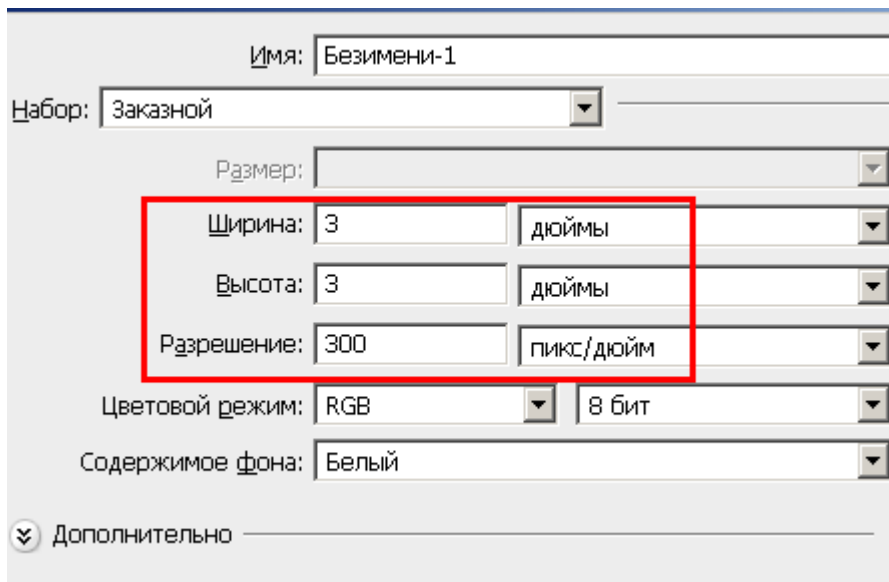
Стиль слоя (layer styles) - для того чтобы добавить цвет самому тексту;

Слой-маску (layer masks) - для того чтобы более реалистично соединить огонь с текстом.

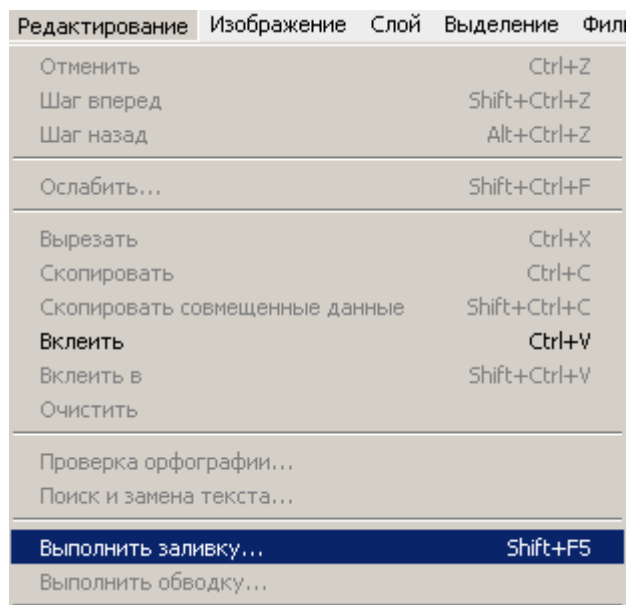
Вот результат, к которому мы будем стремиться:



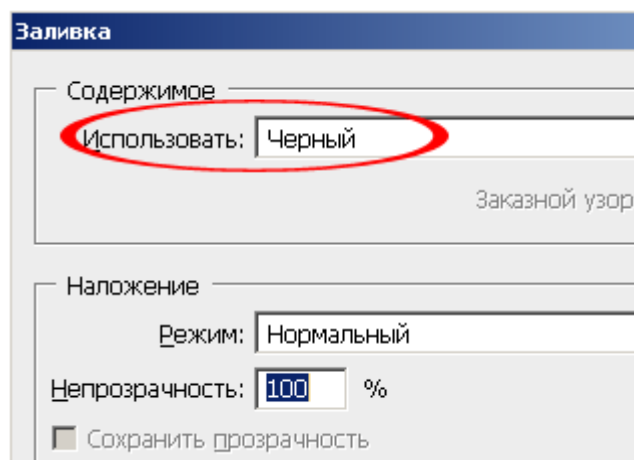
Шаг 1. Создайте новый документ (**Ctrl+N**). Размерами 3x3 дюйма, с разрешением 300 пикс./дюйм.



Залейте документ Черным цветом. Зайдите в **Редактирование – Выполнить заливку (Edit-Fill (Shift+F5))**.



В окне **Заливка (Fill)** установите **Содержимое (Contents) – Использовать (Use) – Черный (Black)**.



Нажмите **ДА**, ваш документ залейется черным цветом.

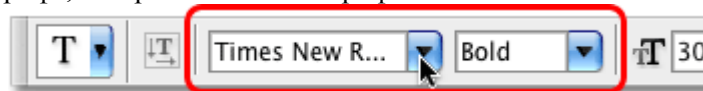


*так же можно использовать горячие клавиши: **(D, Alt+Backspace)***

Шаг 2. Теперь нужно написать текст. Выберите инструмент «Текст» **T**. (Type Tool (T)).



Установите любой шрифт, автор использовал шрифт **Times New Roman Bold**.



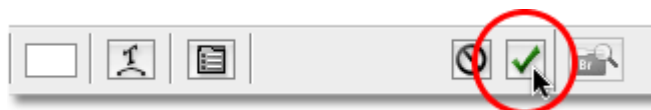
Установите белый цвет



И напишите слово – **FIRE**.

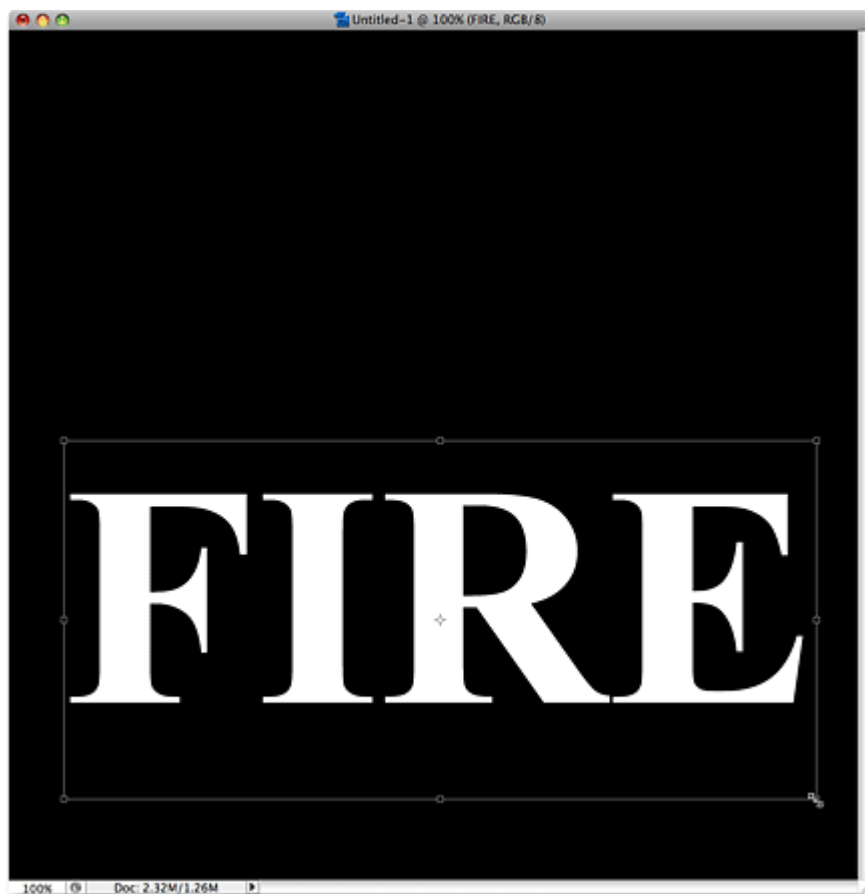


После того как напишите слова, примените текст. Нажав на галочку в правом верхнем углу Photoshop.

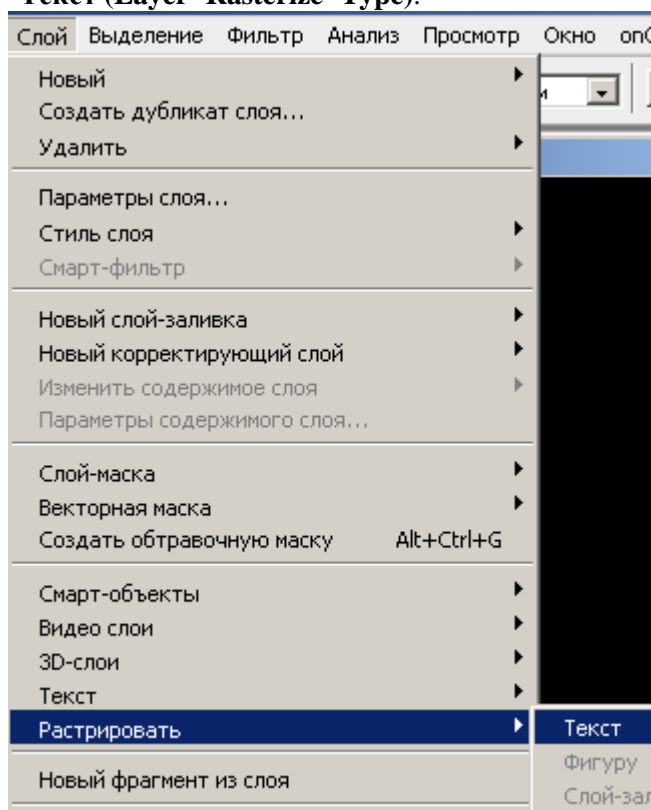


Или просто кликнув один раз на слое с текстом.

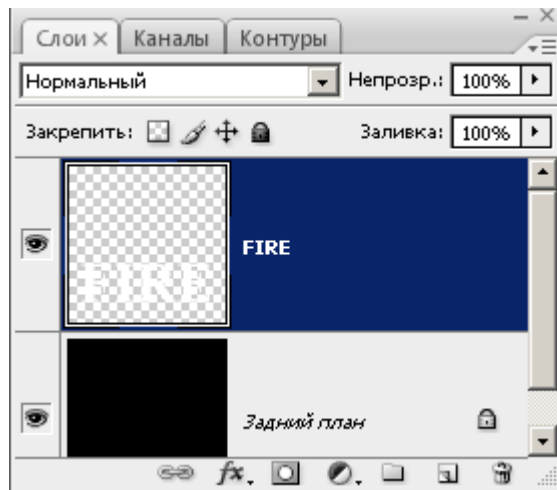
Теперь нужно увеличить текст в размере. Нажмите (**Ctrl+T**), чтобы вызвать **Свободную трансформацию (Free Transform)**. Удерживая клавишу **Shift**, потяните за край рамки трансформирования, увеличивая текст, почти на всю ширину документа. Поместите текст по центру, ближе к нижнему краю документа. Примените трансформацию (**Enter**).



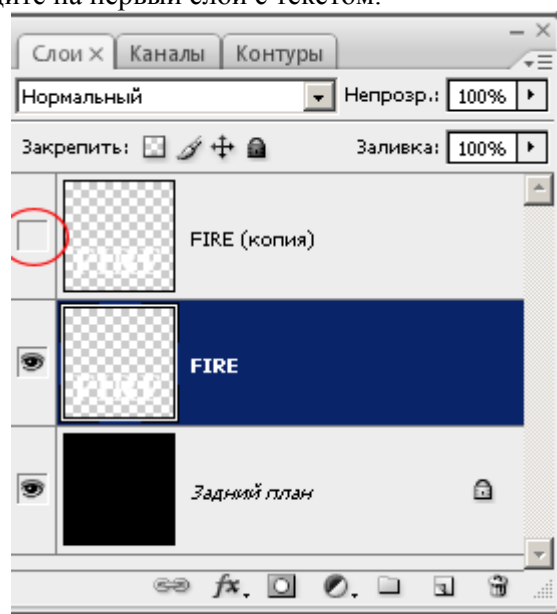
Шаг 3. Для дальнейшего шага, нам нужно **Растривать (Rasterize)** текст.
Слой – Растривать – Текст (Layer- Rasterize- Type).



Либо просто правой кнопкой на слое с текстом и в выпавшем меню, выберите - **Растривать текст**. Видимых изменений с текстом не произойдет, но в палитре слоев изменения будут заметны. Текстовый слой, превратиться, в обычный.

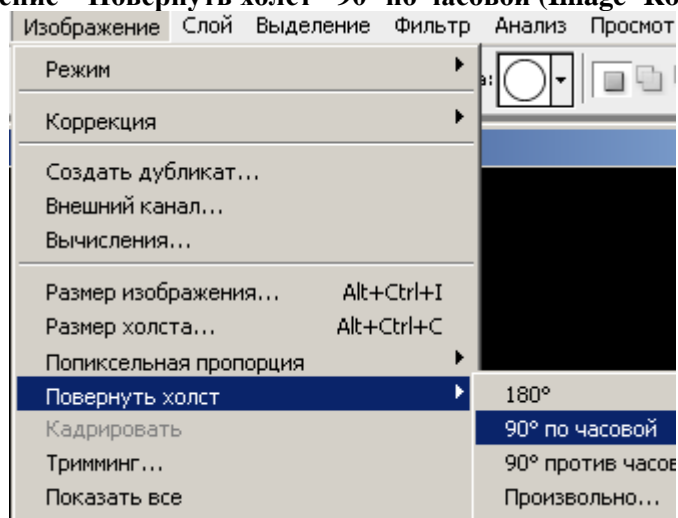


Для того чтобы создать эффект пламя, нам нужно будет воздействовать на дубликат слоя. Создайте копию слоя с текстом (**Ctrl+J**), уберите видимость копии слоя (убрав глазик возле миниатюры слоя) и перейдите на первый слой с текстом.

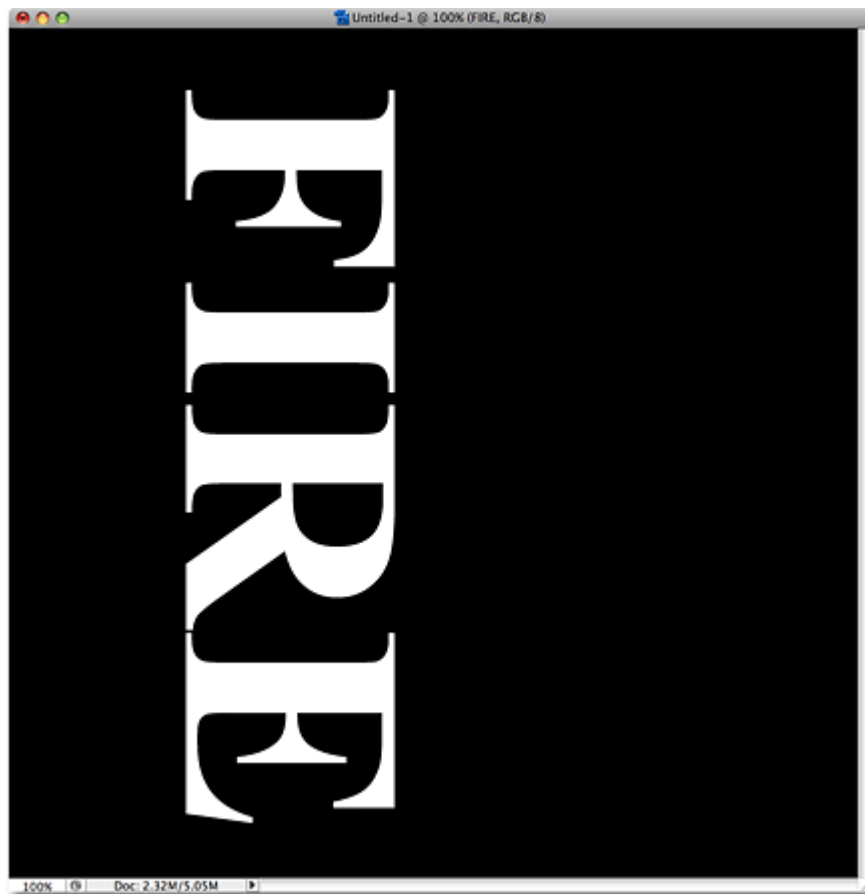


Шаг 4. Для создания эффекта пламя, мы будем использовать несколько фильтров. Первым будет **фильтр Ветер (Wind)**. Этот фильтр работает только слева направо (или справа налево), это означает, что нам нужно будет повернуть наш документ.

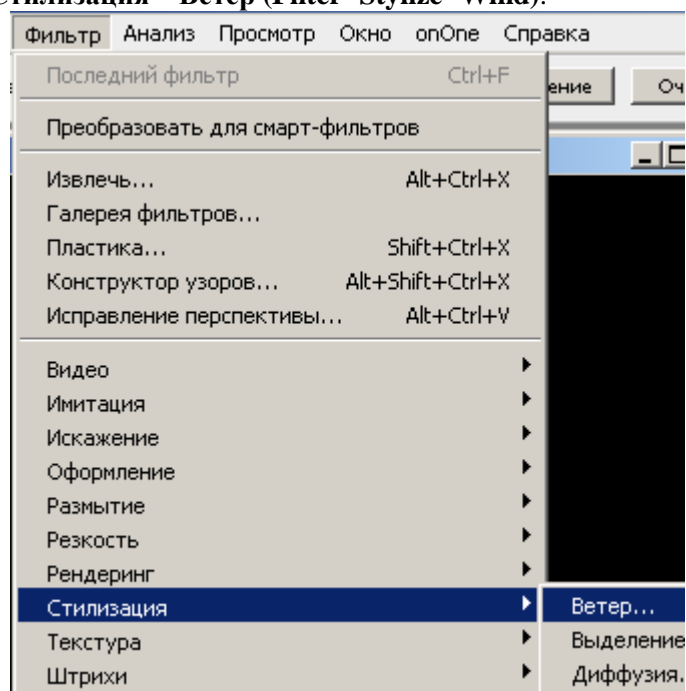
Перейдите в **Изображение – Повернуть холст - 90° по часовой (Image - Rotate Canvas - 90° CW)**.



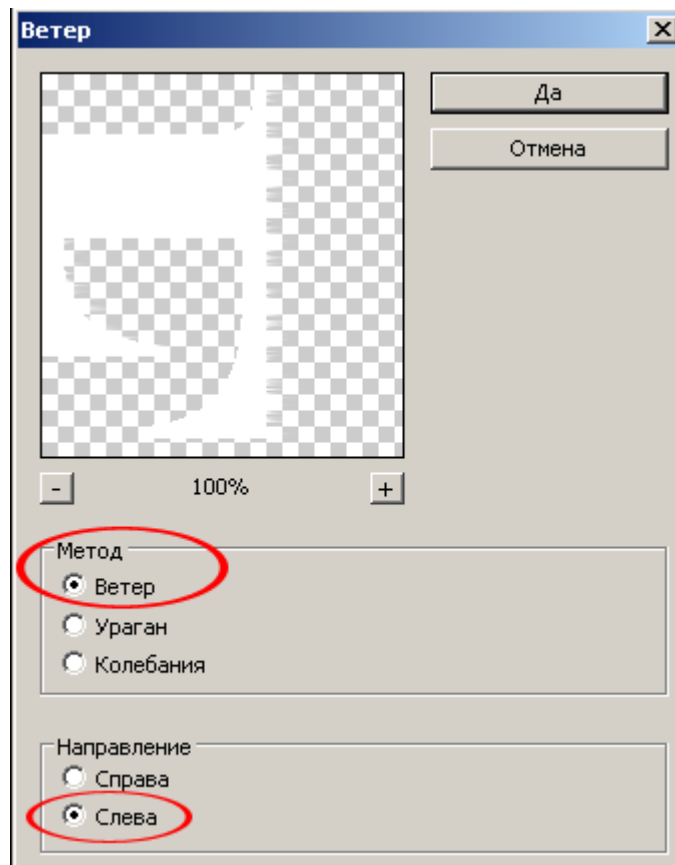
Текст примет вертикальное положение.



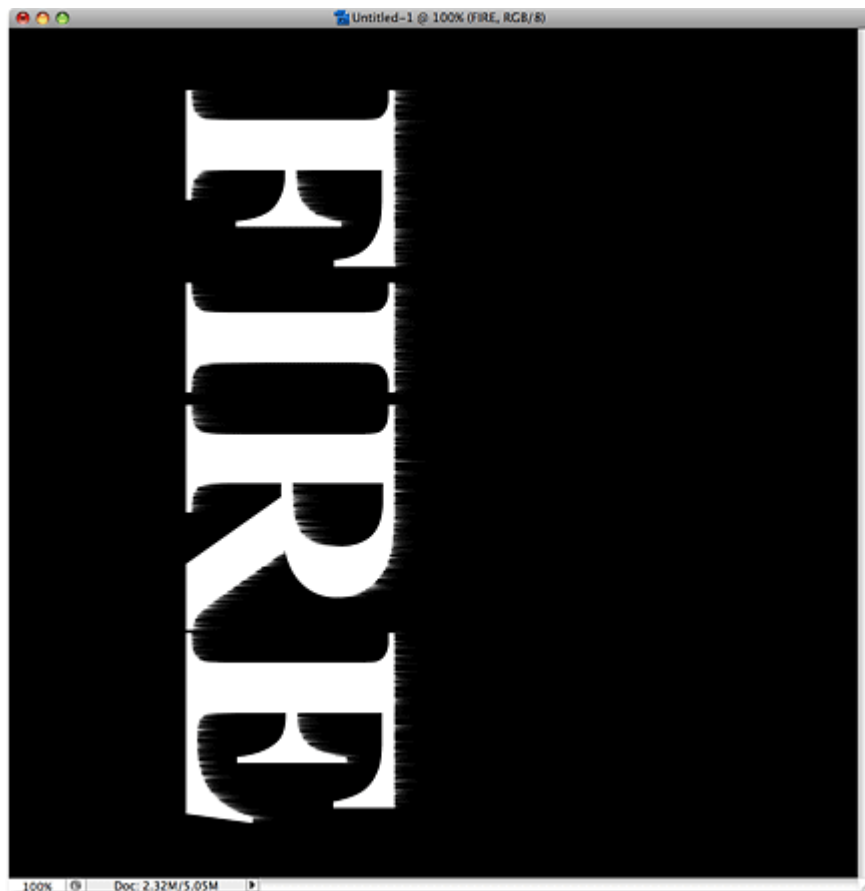
Зайдите в **Фильтр – Стилизация – Ветер (Filter- Stylize- Wind)**.



В появившемся окне установите **Метод (Method) – Ветер (Wind)**, а **Направление (Direction) – Слева (From the Left)**.

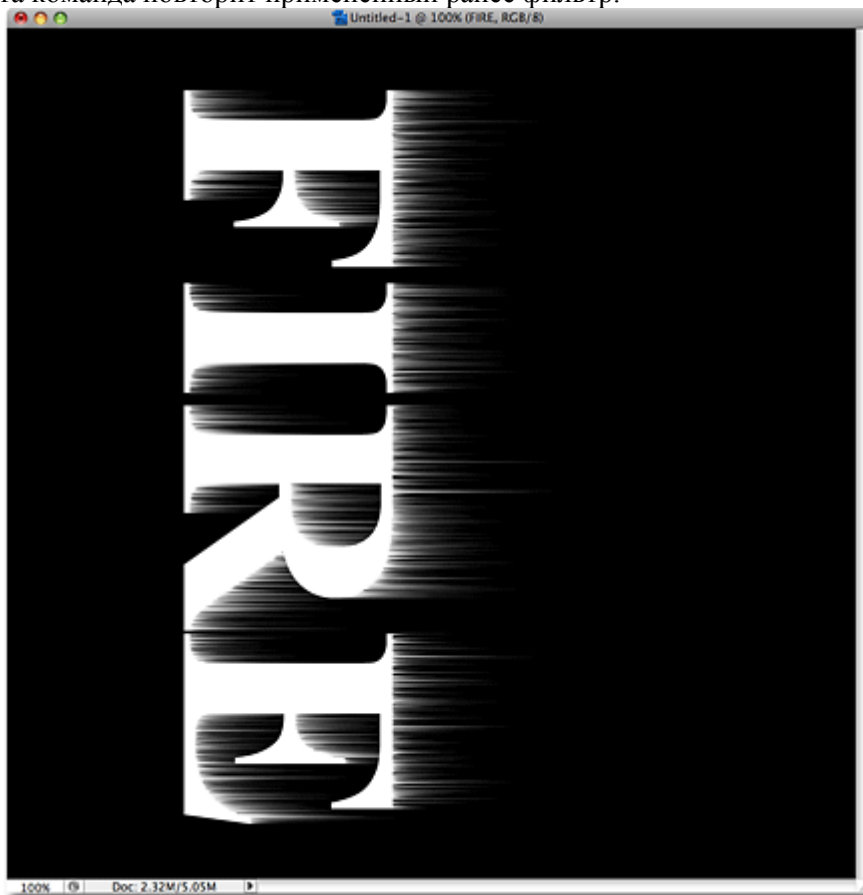


Нажмите ДА.



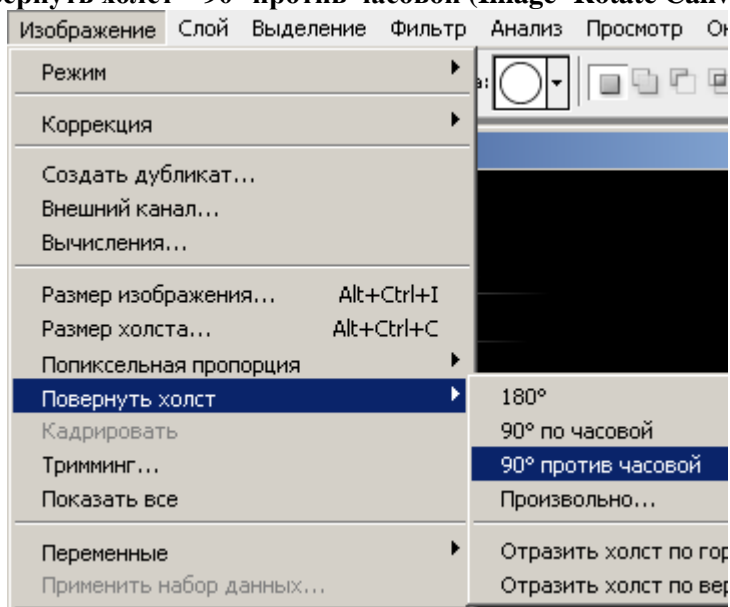
Однократное применение фильтра в данной ситуации вызывает незначительные изменения, для улучшения эффекта, нужно будет повторно применить фильтр.

Чтобы снова не заходить в Меню- Фильтры, просто нажмите **Ctrl+F**, несколько раз (двух будет достаточно), эта команда повторит примененный ранее фильтр.



Верните холст обратно в горизонтальное положение.

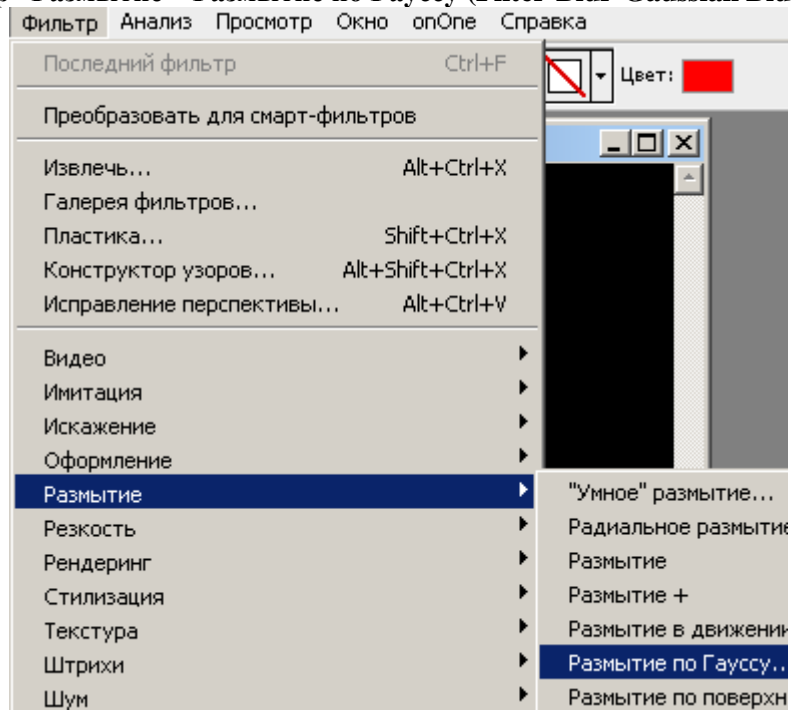
Изображение – Повернуть холст – 90° против часовой (Image- Rotate Canvas- 90° CCW).



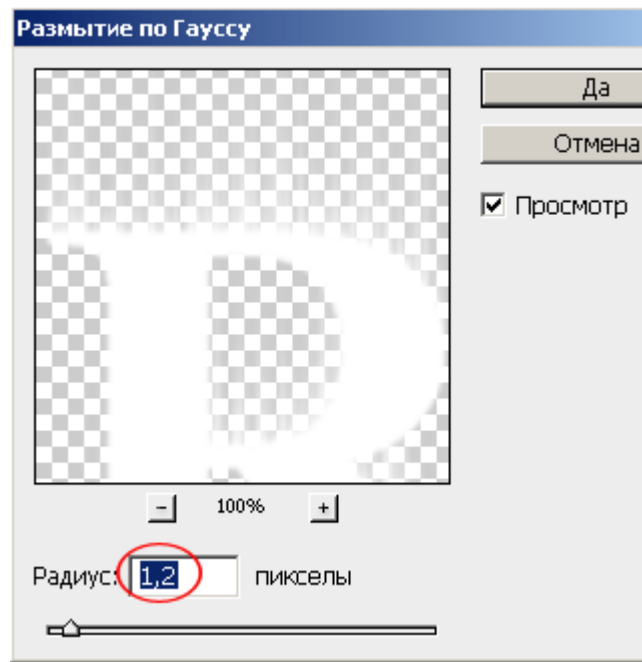


Чтобы линии были не такие резкие, мы их размоем, используя фильтр **Размытие по Гауссу (Gaussian Blur)**.

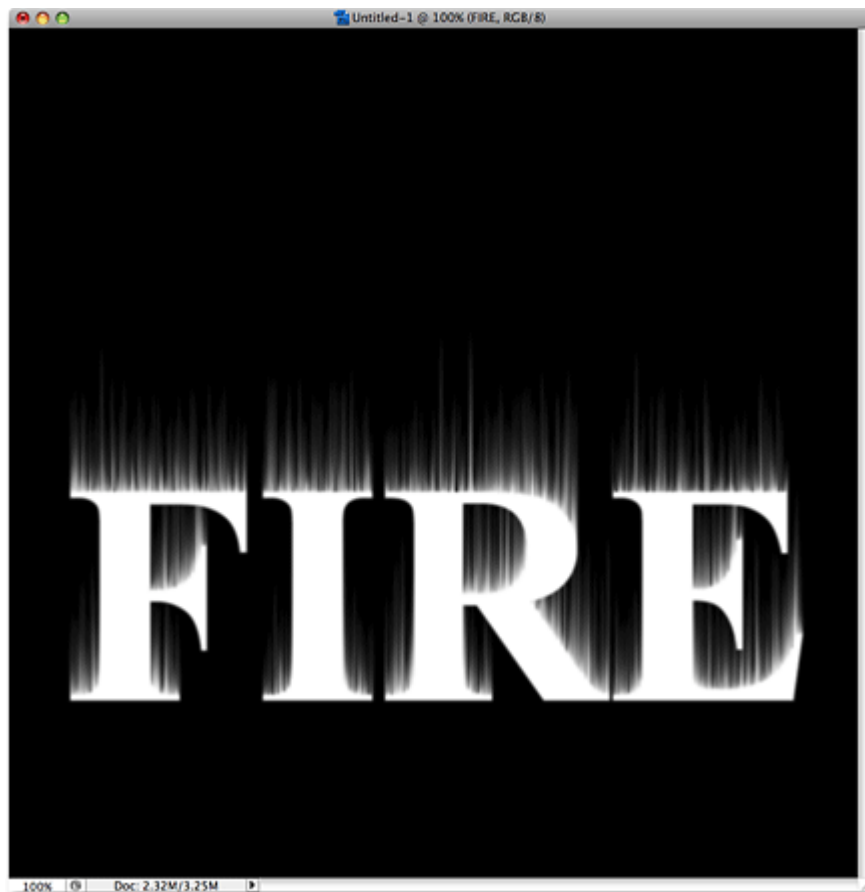
Зайдите в **Фильтр – Размытие – Размытие по Гауссу (Filter-Blur-Gaussian Blur)**.



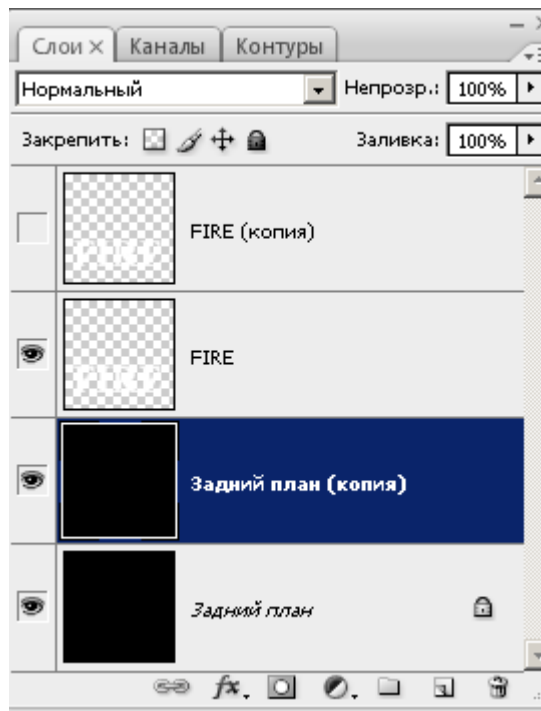
В окне Размытия по Гауссу, поставьте небольшой **Радиус (Radius) – 1,2 пикс.**



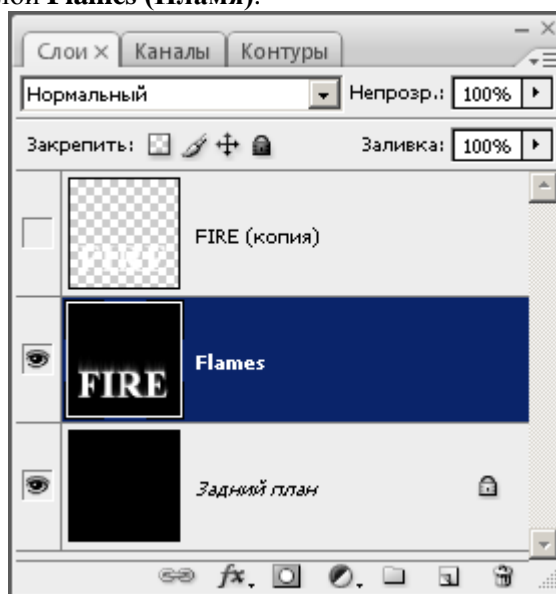
Нажмите ДА.




Шаг 5. Дублируйте первый слой «Задний план» (Background). Перейдите на слой «Задний план» и нажмите **Ctrl+J**.



Слейте слой с текстом, с копией слоя (Задний план), который мы только что создали. Для этого перейдите на слой **FIRE**, зайдите в **Слой – Объединить с предыдущим (Layer-Merge Down)**. Либо, просто нажмите **Ctrl+E**. Назовите получившийся слой **Flames (Пламя)**.



Шаг 6. Переходим к фильтру **Пластика (Liquify)**.

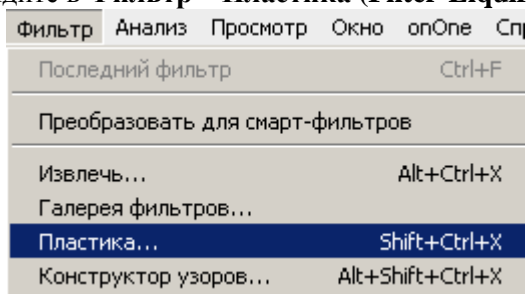
Для начала нам нужно выделить область, над которой мы будем работать. Для этого возьмите инструмент «**Прямоугольная область**»  (**Rectangular Marquee Tool (M)**).



Выделите текст и убедитесь, что оставили достаточно свободного места сверху текста.



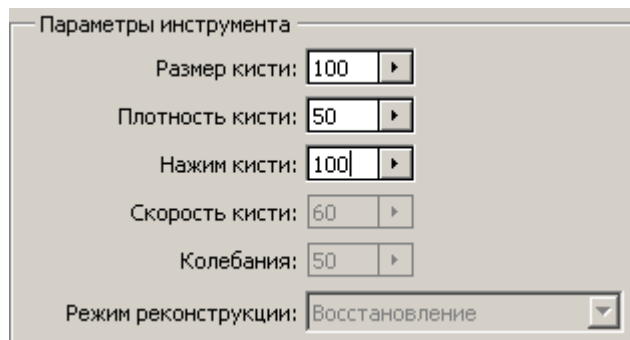
Не снимая выделения, перейдите в **Фильтр – Пластика (Filter-Liquify (Shift+Ctrl+X))**.



В окне фильтра Пластика, выберите инструмент «Деформация» (**Forward Warp Tool (W)**).



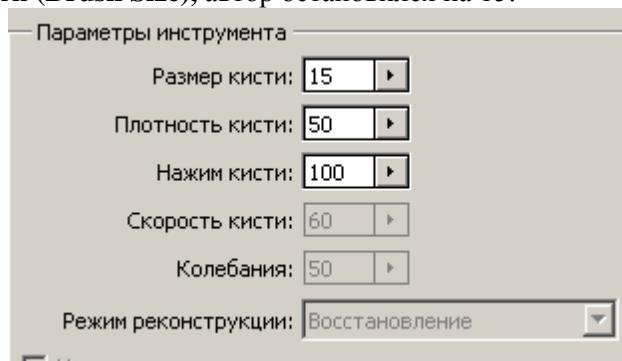
Установите такие настройки инструмента:



Короткими движениями влево и вправо, придайте волоскам некую волнистость, попытайтесь симитировать пламя. Постарайтесь пока не деформировать сам текст. У вас должно получиться что-то похожее.



Измените - **Размер Кисти (Brush Size)**, автор остановился на 15.



На это раз, мы будем вытягивать «языки» пламени, прямо из букв.

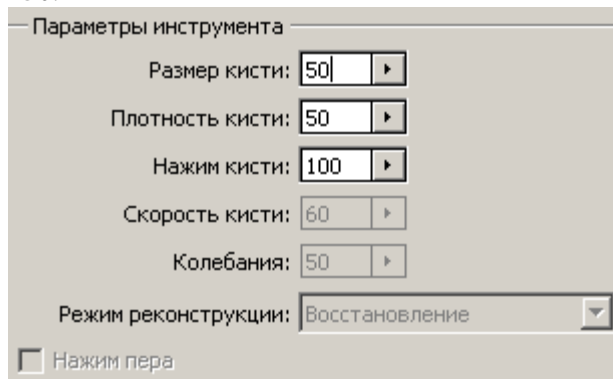
Щелкните на букве и тяните мышь к краю буквы, вытягивая тем самым «язычок».

Если у вас не получилось что-то и вы хотите отменить нарисованное, то нажмите **Ctrl+Z**, это отменить действие.

Если нужно отменить сразу несколько шагов, то жмите **Ctrl+Alt+Z** столько раз, сколько шагов вам нужно отменить. Попробуйте добиться, примерно такого результата.



И в завершении, нарисуем еще немного, больших пламенных «языков». Установите размер Кисти **50**.



Так же, как и в предыдущем шаге, вытягивая пламя из текста, добавьте еще немного «языков». Не забывайте о том, что нижняя граница буквы, должна остаться незатронутой, чтобы не потерять контур буквы.

Должно выглядеть примерно так.



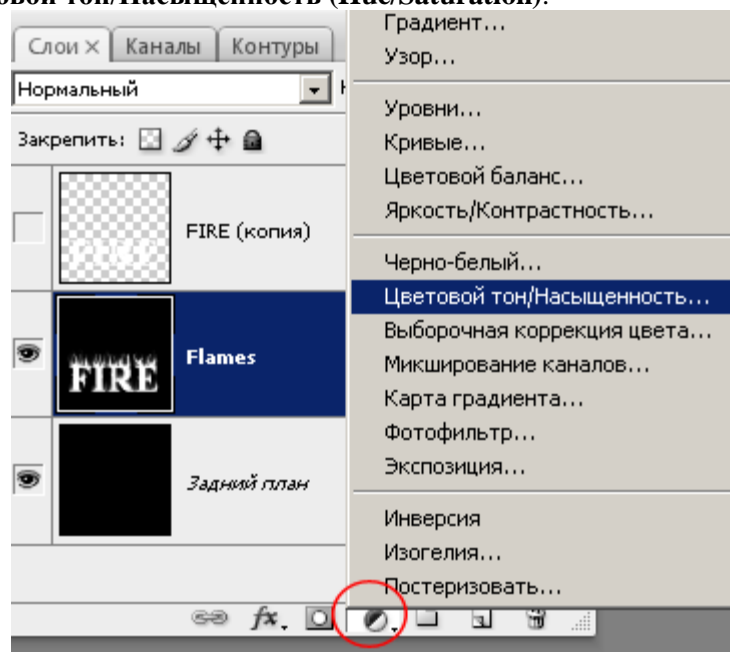
Когда закончите, нажмите ДА, чтобы применить фильтр.



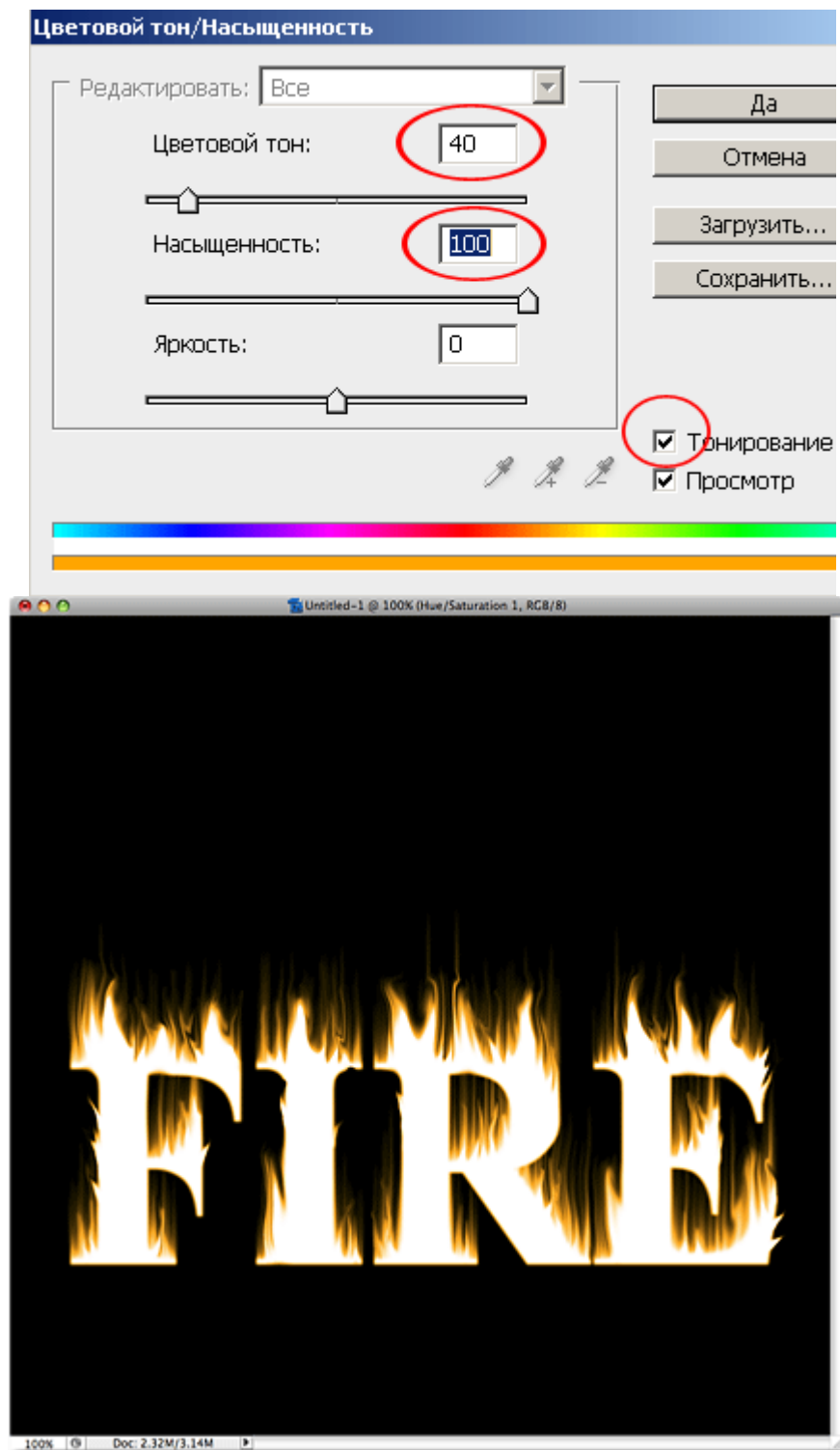
Ctrl+D, снимите выделение.

Шаг 6. Теперь добавим цвета нашему огню. Для этого мы будем использовать **Корректирующий слой – Цветовой тон/Насыщенность (Adjustment Layer-Hue/Saturation)**.

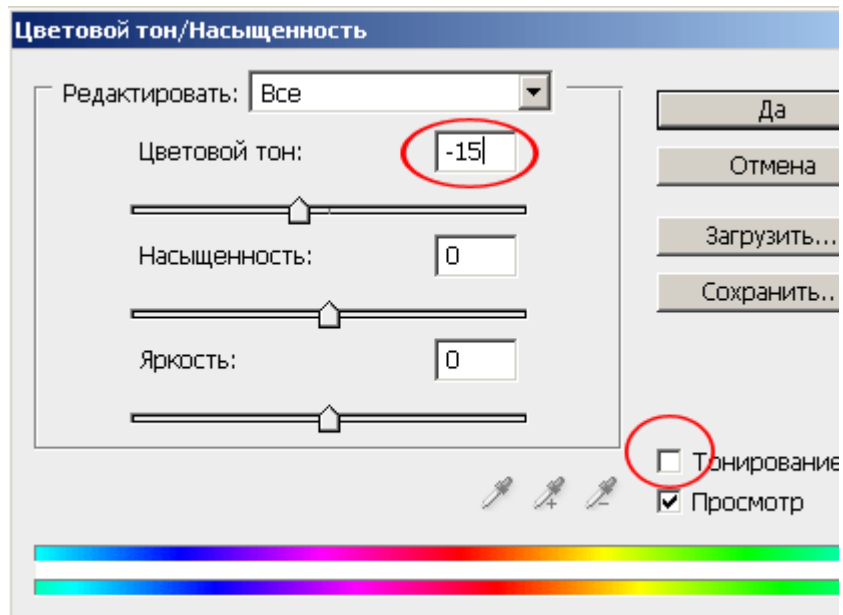
Нажмите на значок , **Создать новый корректирующий слой (New Adjustment Layer)** и выберите там **Цветовой тон/Насыщенность (Hue/Saturation)**.



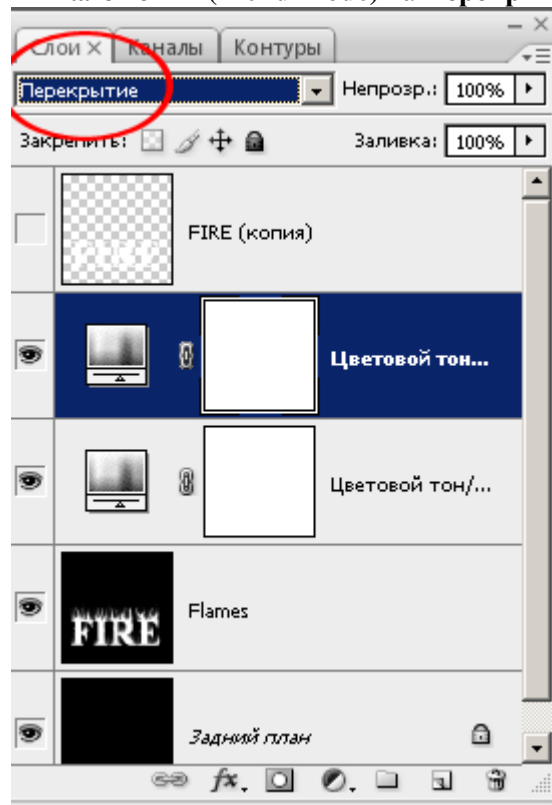
В окне Цветовой тон/Насыщенность, установите такие параметры:



Добавьте еще один **Корректирующий слой** – **Цветовой тон/Насыщенность** (**Adjustment Layer-Hue/Saturation**). И установите в нем такие параметры:



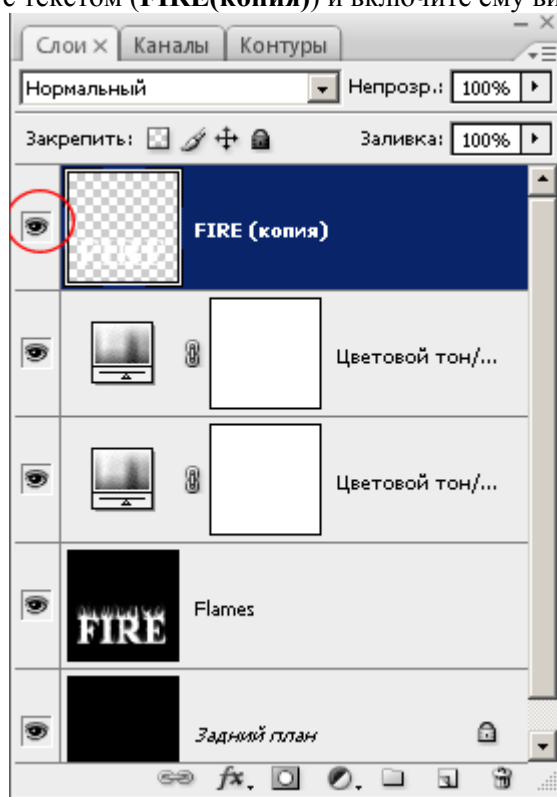
Измените этому слою, режим Наложения (Blend Mode) на **Перекрывание (Overlay)**.



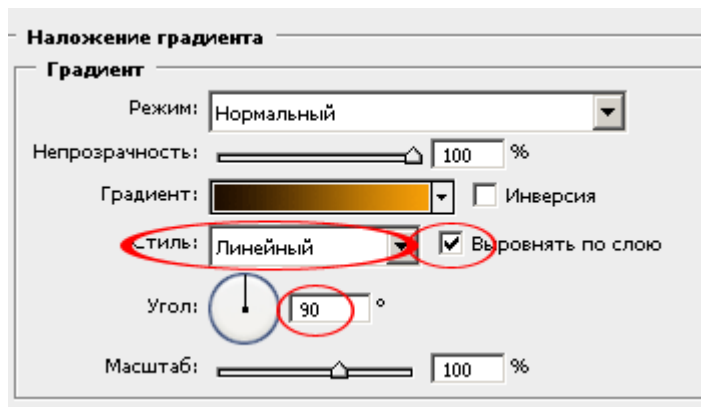
Это добавит, более реалистичный цвет нашему пламени: желтый оттенок, в светлых участках и глубокий, насыщенный, апельсиновый оттенок в темных.



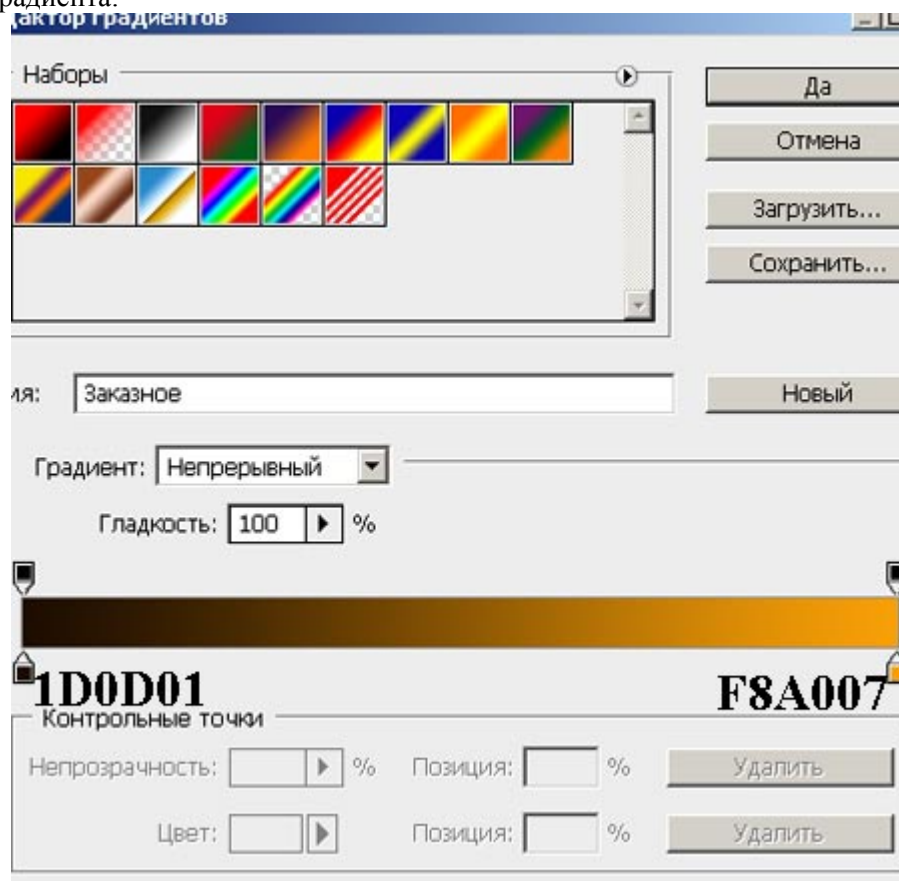
Шаг 7. С пламенем мы закончили, теперь осталось немного поработать с цветом текста. Перейдите на второй слой с текстом (**FIRE(копия)**) и включите ему видимость.



Добавьте этому слою **Стиль слоя(Layer Styles):**
Наложение Градиента (Gradient Overlay) :




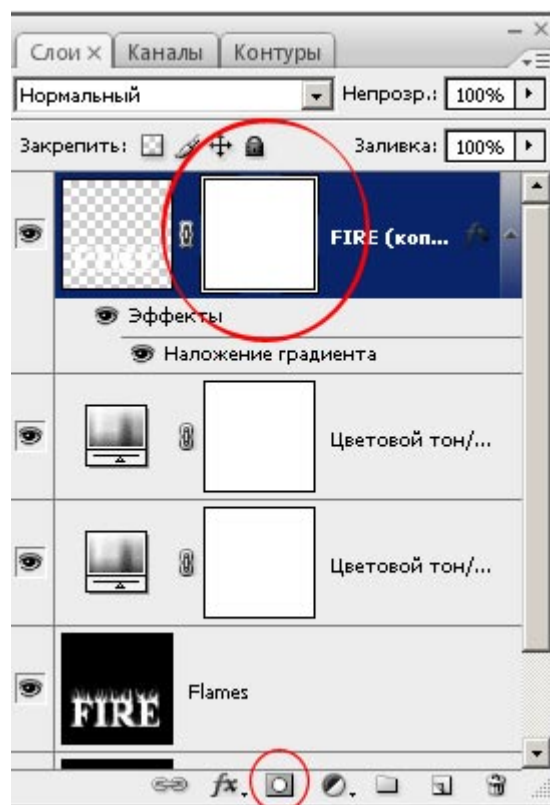
Параметры градиента:



Вот что получится.



Шаг 8. Так как мы хотим сделать текст, охваченный огнем, нам придется проделать еще кое-что. Добавьте **Слой-маску (Layer Mask)** слою **FIRE(копия)**, нажав на значок  внизу палитры слоев.



Возьмите инструмент «Кисть» (**Brush Tool (B)**).



Установите **Черный** цвет (**D**).



Жесткость 0%, **Размер** установите небольшой, относительно вашего текста. Убедитесь, что вы находитесь на **Слой-маске**.


Уменьшая и увеличивая, **Размер** и **Жесткость Кисти**, сотрите все резкие границы текста, которые соприкасаются с «языками» пламени. Тем самым, попытайтесь добиться иллюзии того, как будто текст находится в огне.



Если вы случайно стерли лишнее, то просто переключитесь на белый цвет (**X**) и верните стертое (закрасив нужный участок белым).



Слой с текстом получился немного выше, чем слой с огнем. Из за этого, появилась оранжевая окантовка у основания текстом.

Чтобы это исправить, возьмите инструмент «Перемещение»  (Move Tool (V)) и нажмите на вашей клавиатуре стрелочку вниз, несколько раз, это сместит слой с текстом немного вниз и сровняет его со слоем с огнем.

Вот и наш конечный результат.



Лабораторная работа № 2

Объёмный текст с мелким узором

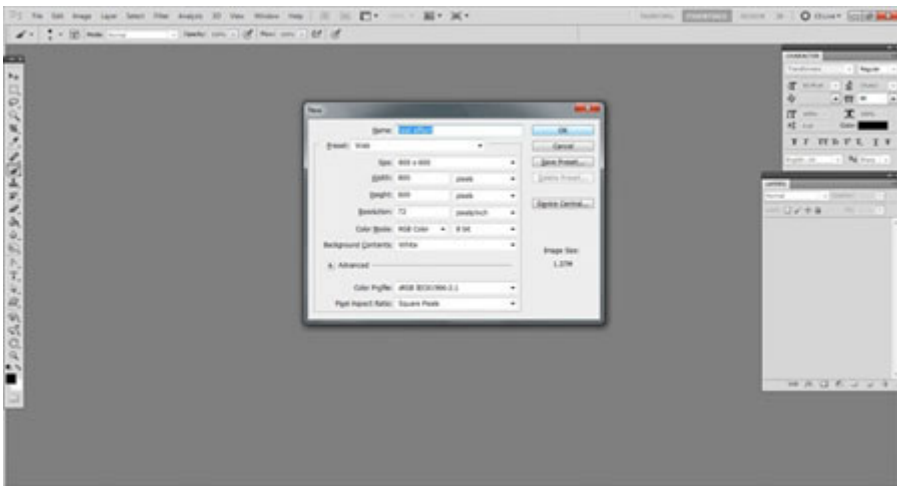
В этом уроке Вы узнаете, как создать текст с мелким узором.

Конечный результат:



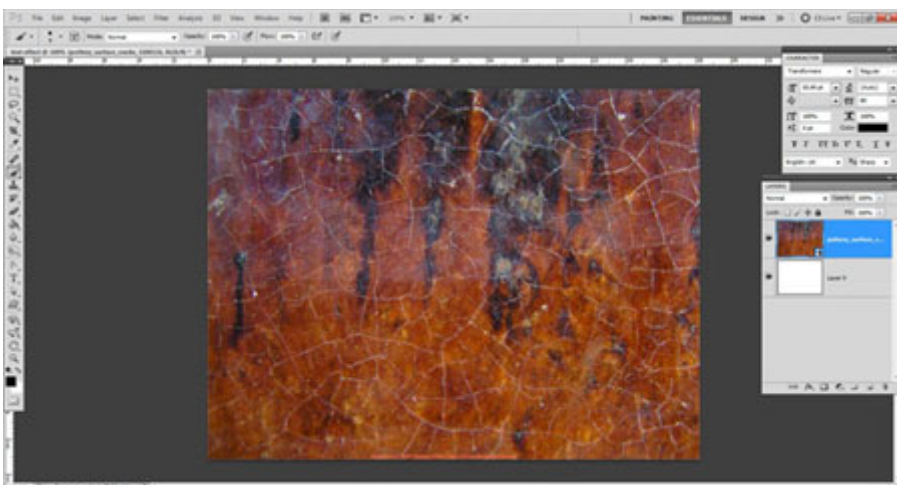
Шаг 1. Создание документа

Создайте новый документ в Фотошопе размером 800x600 пикселей в режиме RGB с разрешением 72 пиксель/дюйм.



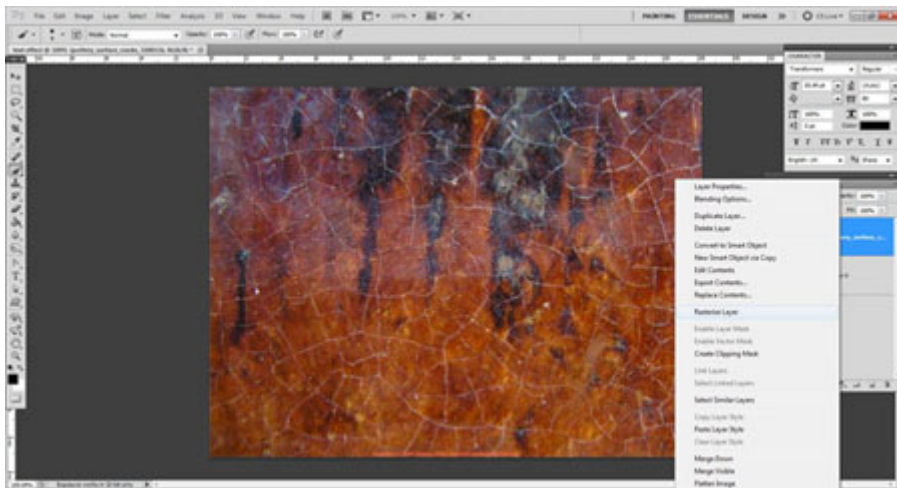
Шаг 2. Текстура

Вставьте текстуру с трещинами в наш документ. Вам не обязательно использовать такую же текстуру, как у меня.



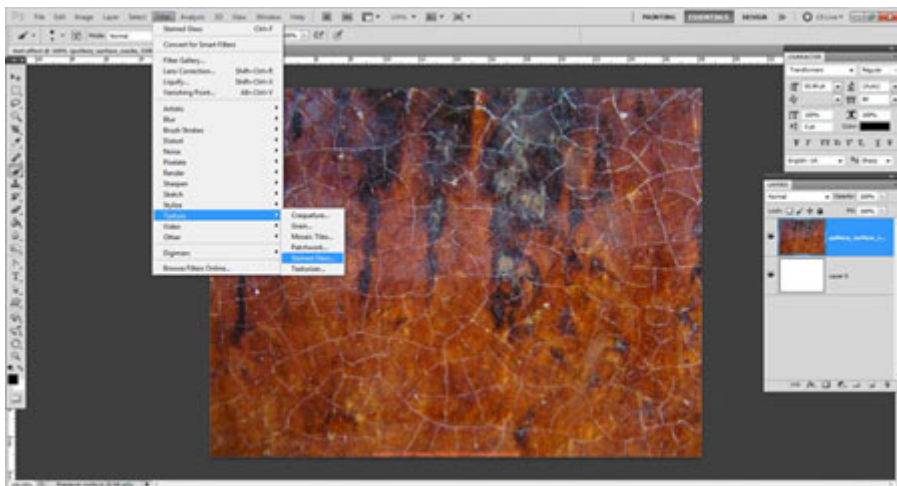
Шаг 3. Растрирование слоя

Кликните правой кнопкой мыши на слое текстуры и выберите пункт **Растрировать слой** (Rasterize Layer).



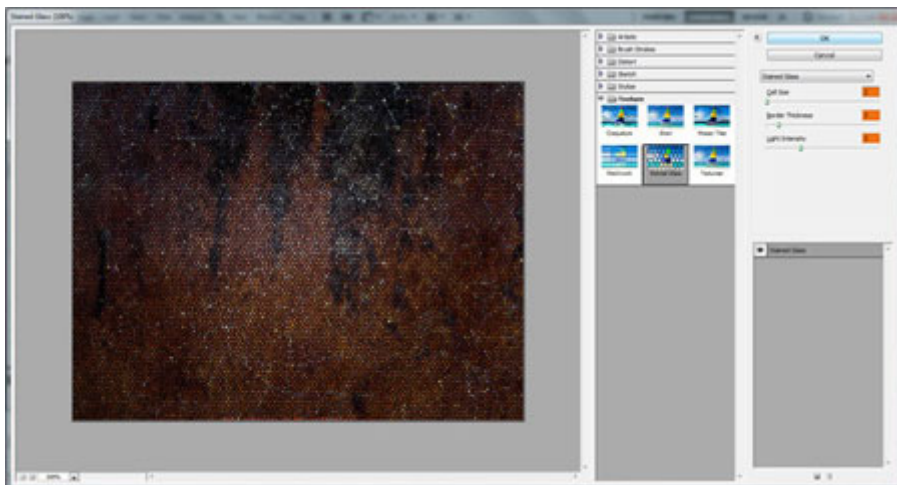
Шаг 4. Использование фильтра

К слою с текстурой примените **фильтр Витраж** (Фильтр > Текстура > Витраж) (Filter > Texture > Stained Glass).




Шаг 5. Настройки фильтра

Размер ячейки (Cell Size): 2
 Толщина границ (Border Thickness): 3
 Интенсивность света (Light Intensity): 3



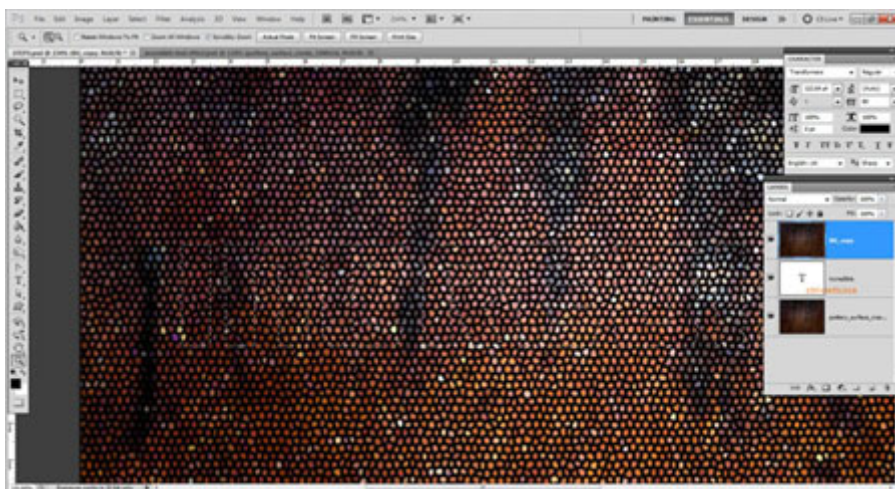
Шаг 6. Текст

Установите шрифт «Transformers» и напишите слово INCREDIBLE при помощи инструмента **Горизонтальный текст**  (Horizontal Type Tool) (T).



Шаг 7. Дублирование текстуры и выделение текста

Создайте копию слоя с текстурой (Ctrl + J) и поднимите её над текстовым слоем. Назовите копию текстуры «BG copy». Затем выделите текстовый слой (удерживая Ctrl, кликните на миниатюре слоя).



Шаг 8. Наложение текстуры на текст

Инвертируйте выделение (Ctrl + Shift + I), перейдите на слой «BG copy» и нажмите клавишу Delete, чтобы оставить текстуру только на тексте.



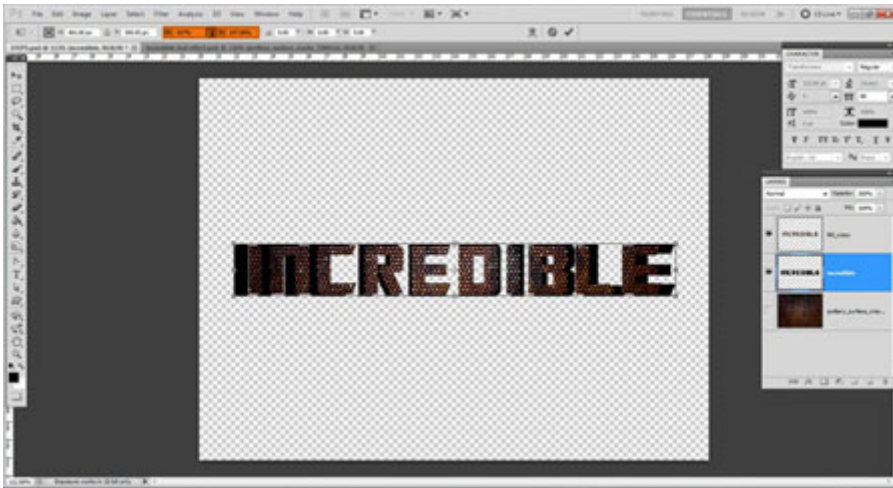
Шаг 9. Цвет текста

Выключите все слои, кроме текстового. Чтобы быстро это сделать, зажмите клавишу Alt и кликните на иконке глаза текстового слоя. Выберите текстовый слой и кликните на цветовой иконке на панели инструментов, чтобы открыть Редактор цвета. Выберите чёрный цвет и нажмите ОК. Затем нажмите Alt + Backspace, чтобы залить текст чёрным цветом.



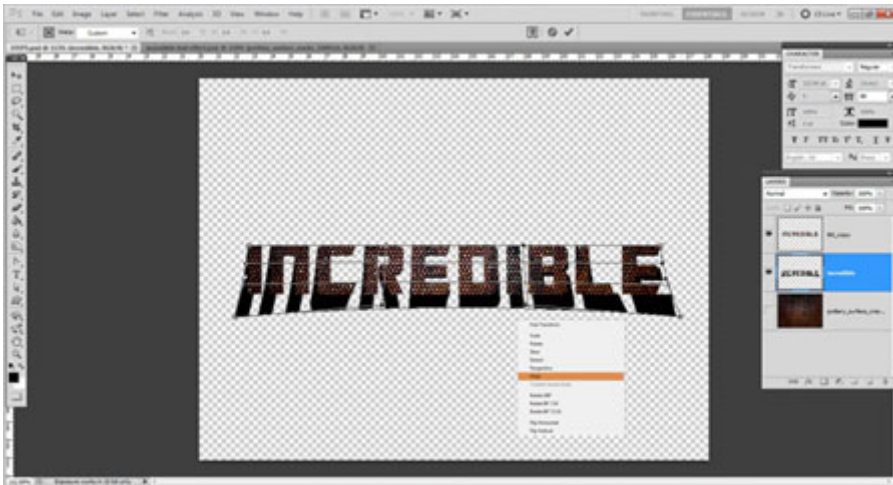
Шаг 10. Растрирование текстового слоя

Растрируйте текстовый слой через контекстное меню и в режиме **Свободное трансформирование** (Ctrl + T) увеличьте текст до 107%.



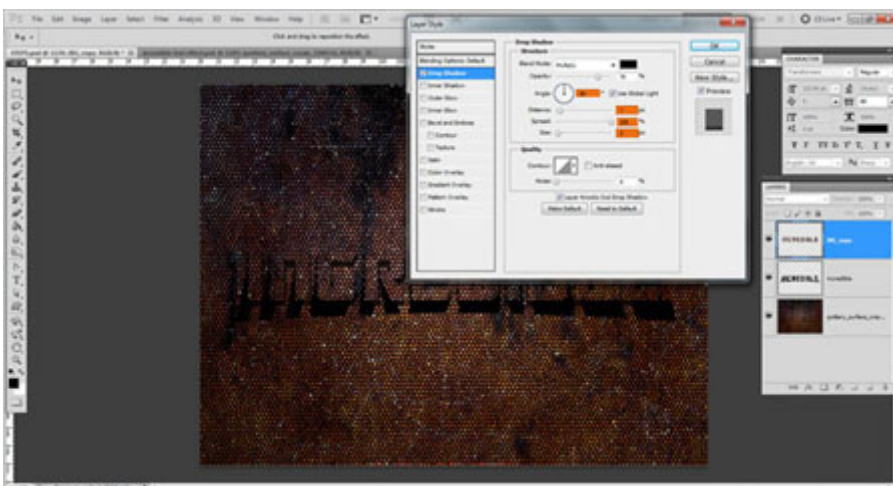
Шаг 11. Форма текста

В режиме свободного трансформирования кликните правой кнопкой мыши на холсте и выберите пункт **Деформация (Warp)**. Измените форму текста, как показано на скриншоте.



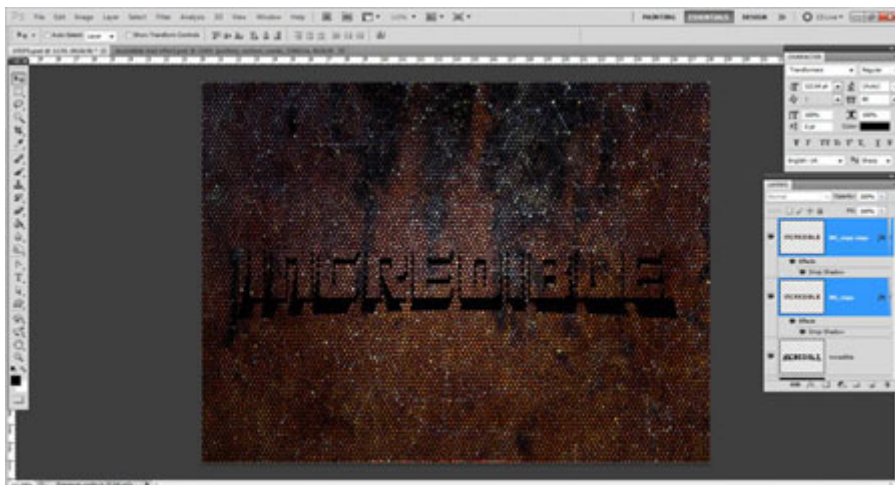
Шаг 12. Тень от текста

Кликните дважды на слое «BG сору», чтобы открыть окно стилей. Примените стиль **Отбрасывание тени (Drop Shadow)** и настройте, как показано ниже.



Шаг 13. Дублирования текстуры текста

Создайте копию слоя «BG сору» (Ctrl + J).



Шаг 14. Рисование тени

Часть тени от текста можно нарисовать и чёрной кистью.



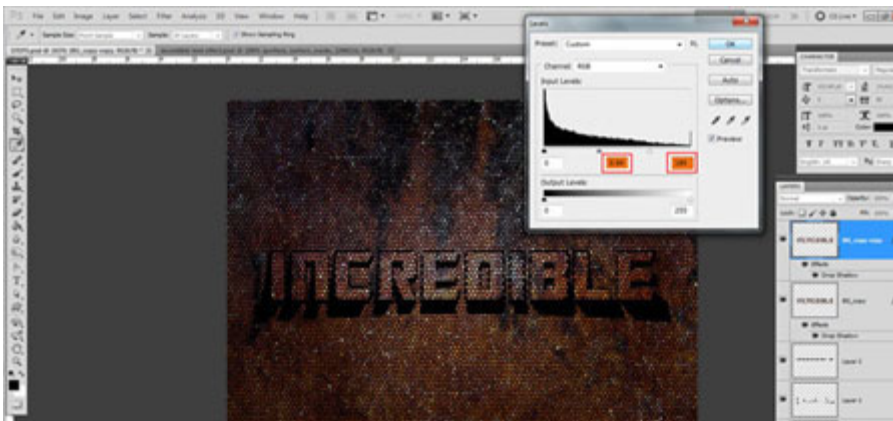
Шаг 15. Размер текста

Активируйте режим Свободное трансформирование (Ctrl + T), зажмите Ctrl + Shift и сожмите текст по горизонтали.



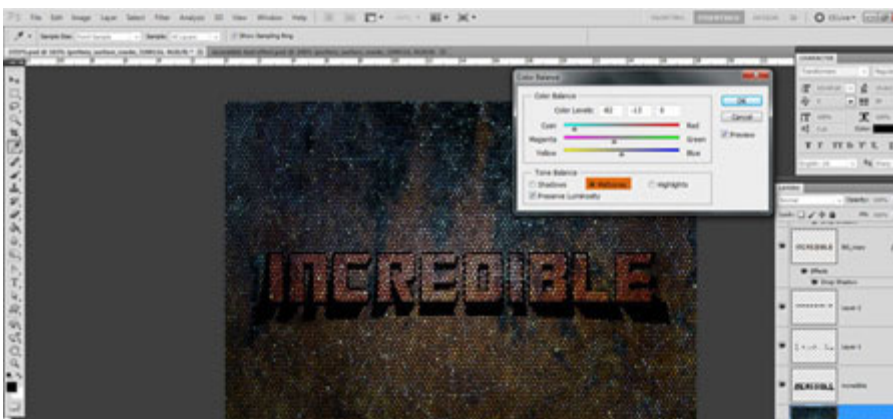
Шаг 16. Коррекция уровней

Выберите самый верхний слой и перейдите к коррекции Уровни (Ctrl + L).



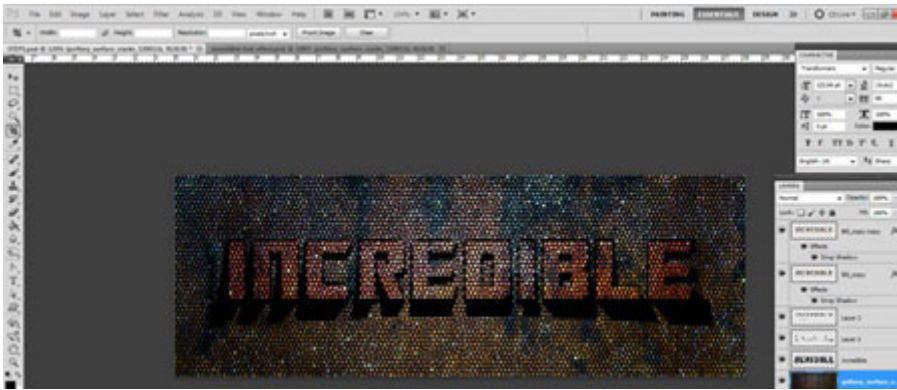
Шаг 17. Цветовой баланс

Выберите самый нижний фоновый слой с текстурой и перейдите к коррекции Цветовой баланс (Ctrl + B), настройте Средние тона (Midtones).



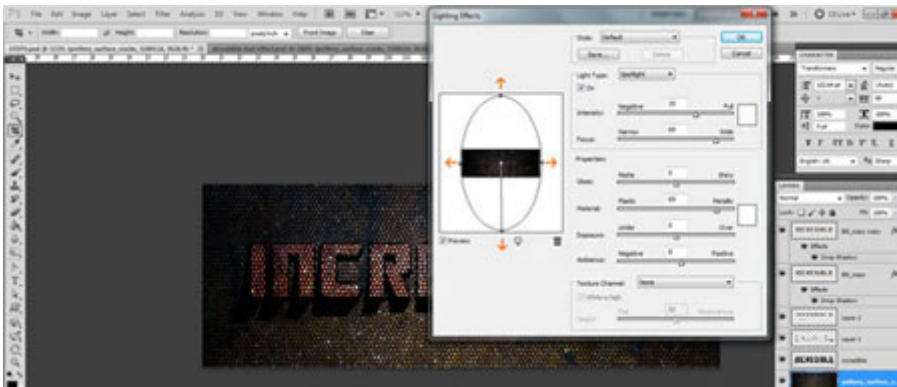
Шаг 18. Обрезка изображения

Выберите инструмент **Кадрирование** (Crop Tool) (C) и отсеките часть изображения.



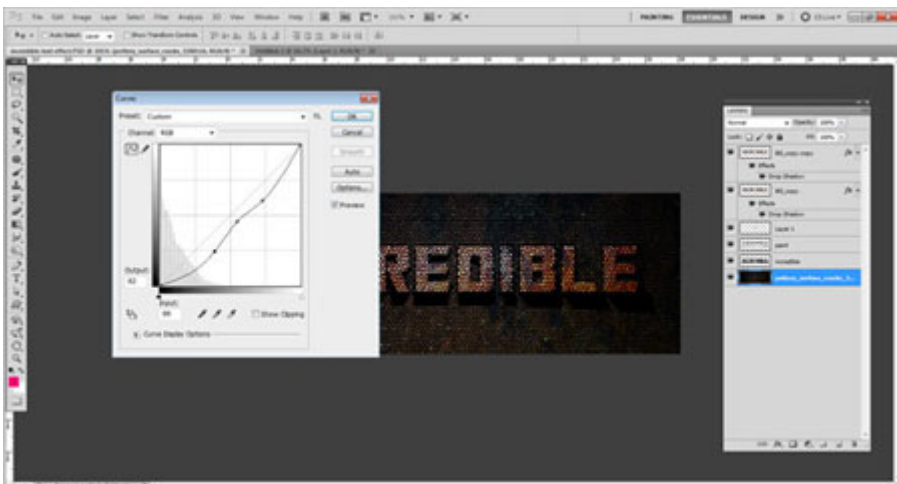
Шаг 19. Освещение

К фоновому слою примените фильтр **Эффекты освещения** (Фильтр > Рендеринг > Эффекты освещения) (Filter > Render > Lighting Effects). Используйте настройки, показанные на скриншоте, или используйте свои.



Шаг 20. Коррекция Кривые

Чтобы осветлить текст ещё сильнее, примените коррекцию **Кривые** (Ctrl + M).



Конечный результат:



Лабораторная работа №3.

Создание теней


В этом уроке показано, как создать тени в Photoshop CC, используя 3D возможности программы.

Конечный результат:



Шаг 1

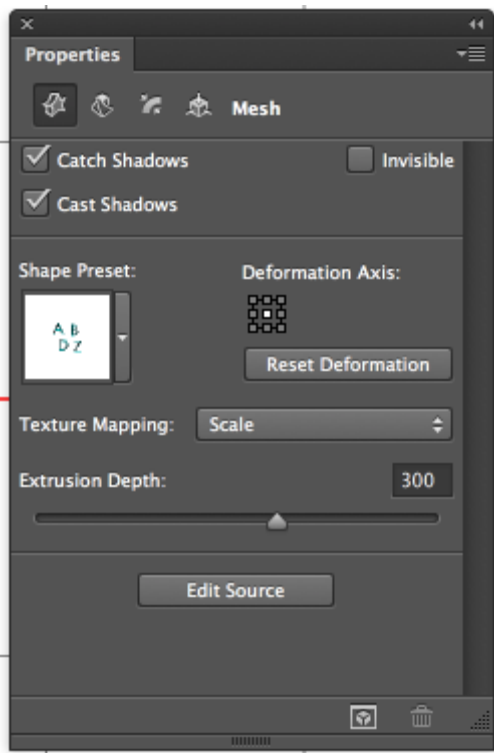
Создайте новый документ в Фотошопе (Ctrl + N) размером 2880x1800 пикселей.

Инструментом **Горизонтальный текст**  (Horizontal Type Tool) (T) напишите «ABDZ» при помощи шрифта «Futura Medium».

A B
D Z

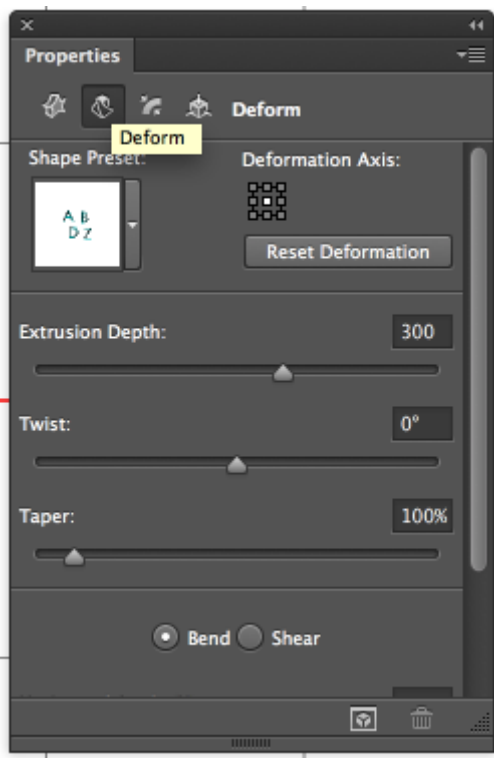
Шаг 2

Перейдите в меню **3D > Новая 3D-экструзия из выделенного слоя** (3D > New 3D Extrusion from Selected Layer). Настройте свойства 3D-объекта так:



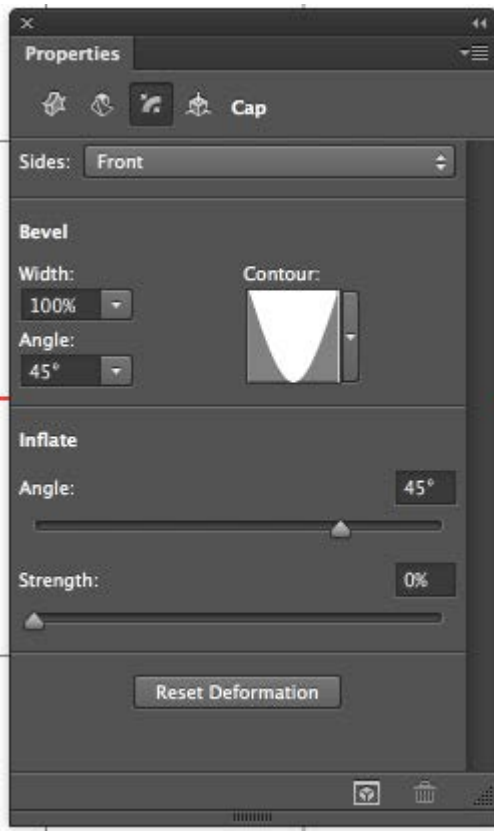
Шаг 3

Перейдите во вкладку деформации и измените глубину экструзии. Затем поверните текстовый объект так, чтобы Вы могли работать с ним сверху.



Шаг 4

Настройте вкладку Капитель (Cap):



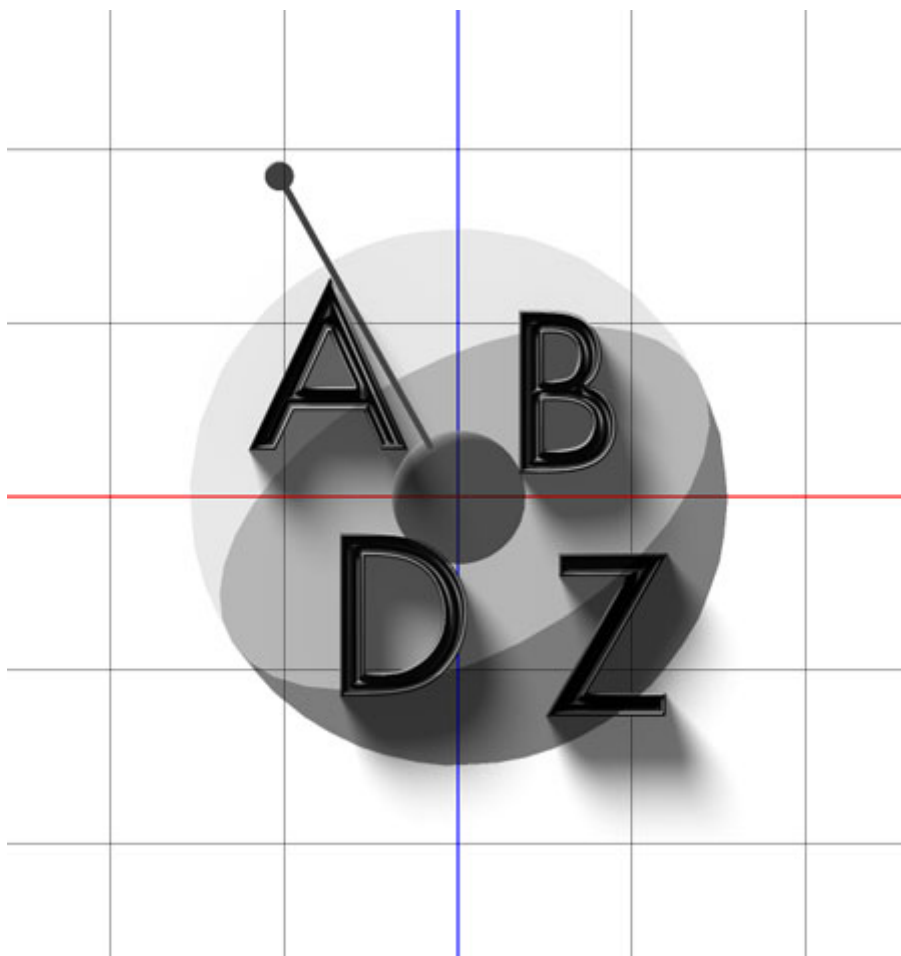
Шаг 5

Выберите **Бесконечный свет** (Infinite Light) на панели 3D, поверните его немного и измените свойства так:

Интенсивность (Intensity): 1000%

Мягкость (Softness): 75%

После этого поставьте сцену на рендеринг (3D > Рендеринг) (3D > Render).



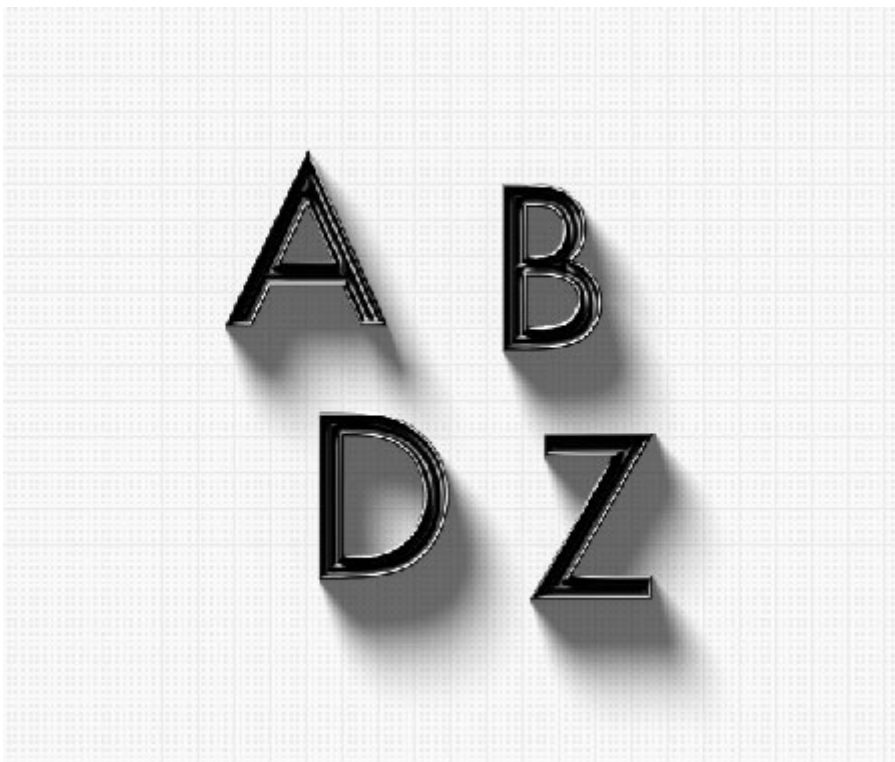
Шаг 6

Вы получите результат, как на скриншоте ниже. Если Вам не понравится результат, измените экструзию и источник света. Важно, чтобы объект находился прямо на плоскости основания.



Шаг 7

Создайте новый слой под текстовым и вставьте на него какой-нибудь узор.



Шаг 8

Добавьте ещё какие-нибудь элементы, чтобы окружающая область не была пустой.



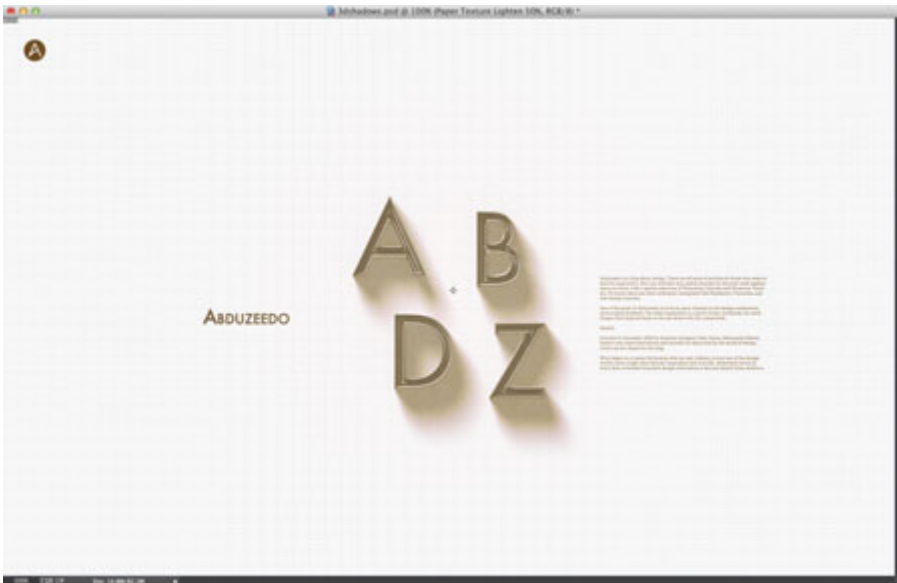
Шаг 9

Вставьте текстуру бумаги поверх всех слоёв.



Шаг 10

Установите режим наложения бумаги на **Замену светлым** (Lighten) и уменьшите непрозрачность до 50%.



Заключение

Сведите все слои на один отдельный (Ctrl + Shift + Alt + E) и установите для него режим наложения **Умножение** (Multiply).

На этом мы заканчиваем урок. Как видите, в Photoshop CC был улучшен режим 3D, и теперь можно создавать разные эффекты более удобным и быстрым способом. Не бойтесь экспериментировать и создавать свои уникальные 3D-объекты и эффекты.



Лабораторная работа №4.

Рисуем фрукты в программе Photoshop

В этом уроке, я покажу вам простую технику создания фруктов, используя программу Photoshop. Приятного урока!

Фрукты

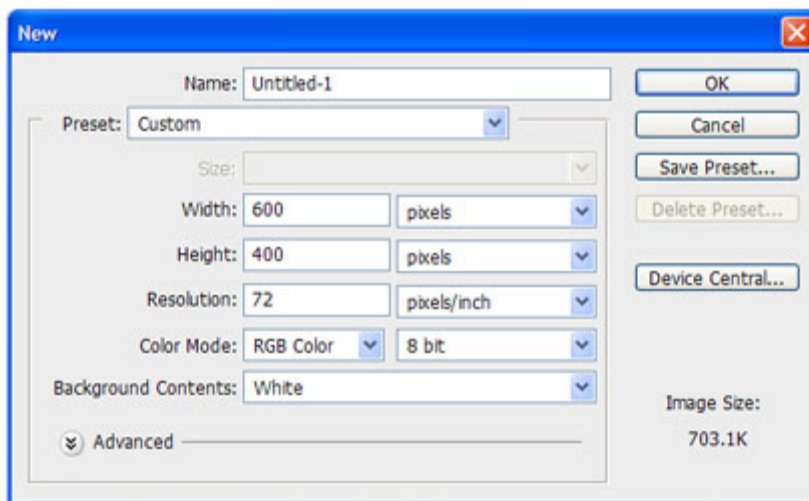
Мы создадим апельсин, банан и виноград.




Как нарисовать Апельсин

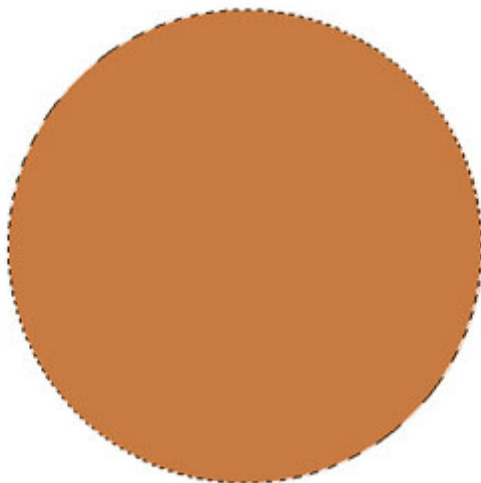
Шаг 1

Создайте новый документ в программе Photoshop, для этого идём **Файл – Новый** (File>New). Для нашего урока, я применил следующие значения 600px x 400px (72 dpi).



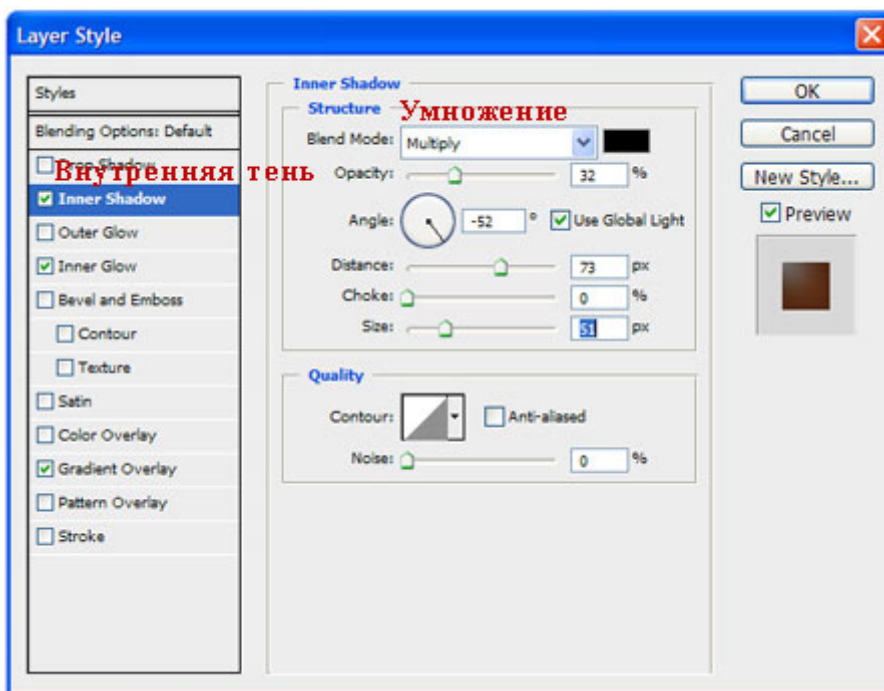
Шаг 2

Создайте новый слой. С помощью инструмента **Овальная область**  (Elliptical Marquee Tool), нарисуйте круг. Залейте круг оранжевым цветом.



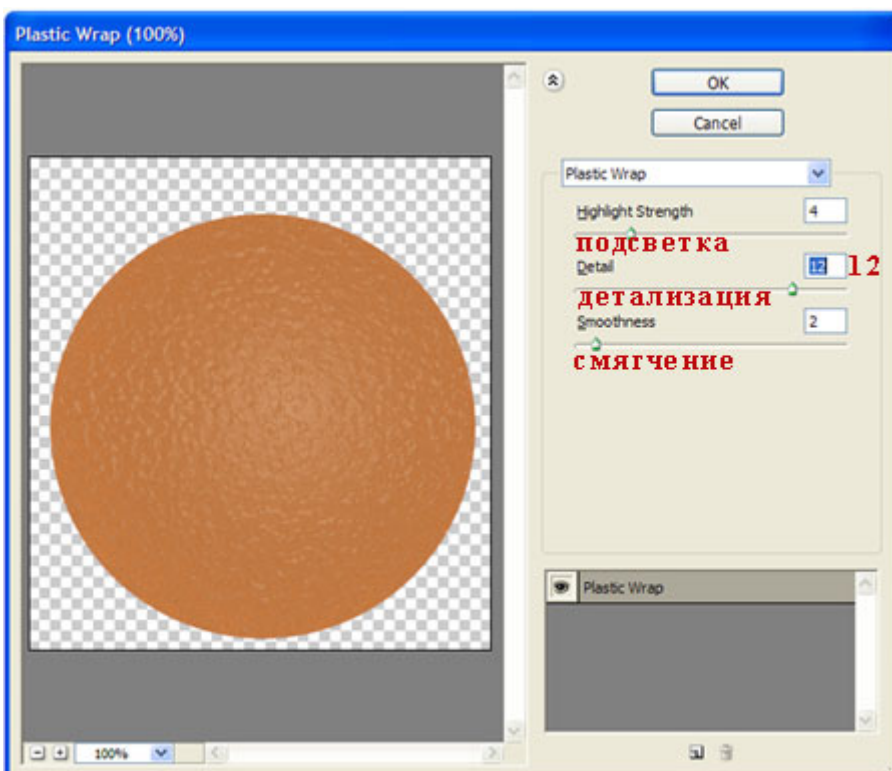
Шаг 3

Добавьте к данному слою немного шума **Фильтр – Шум – Добавить шум** (Filter>Noise>Add Noise). За счёт данного фильтра мы создадим некую текстуру, которая нам понадобится для дальнейшей обработки в следующем шаге.



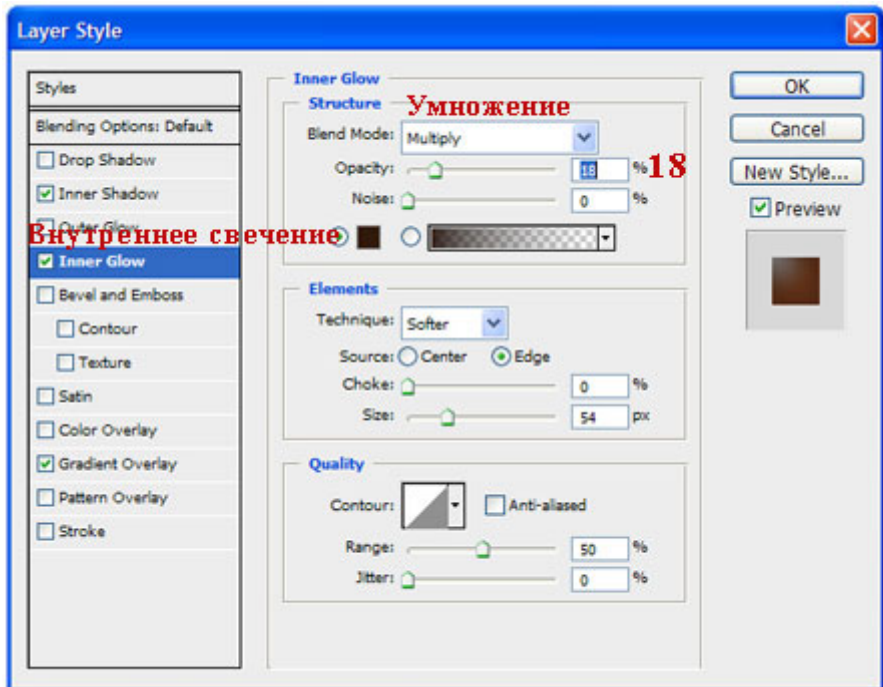
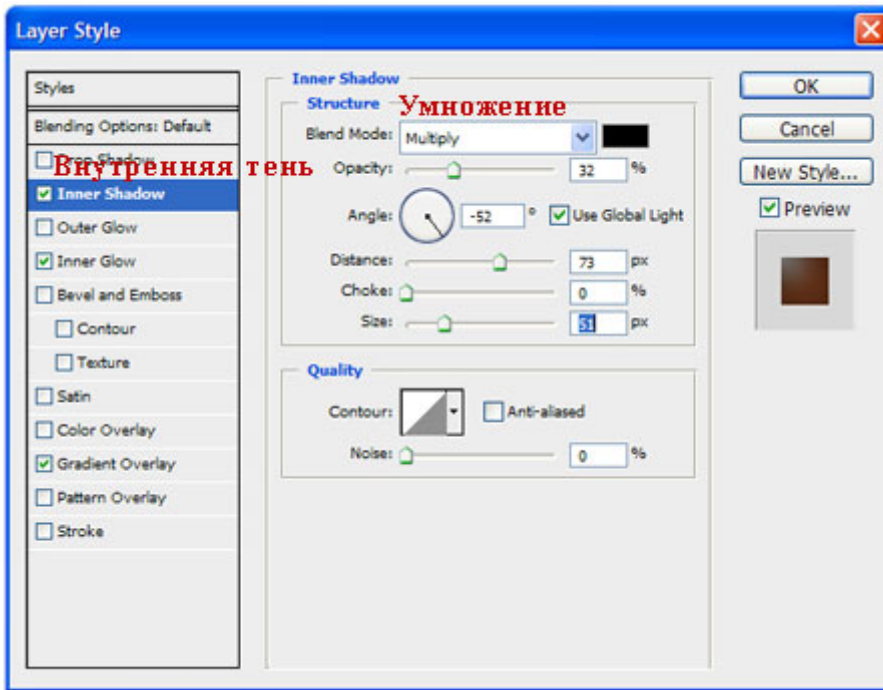
Шаг 4

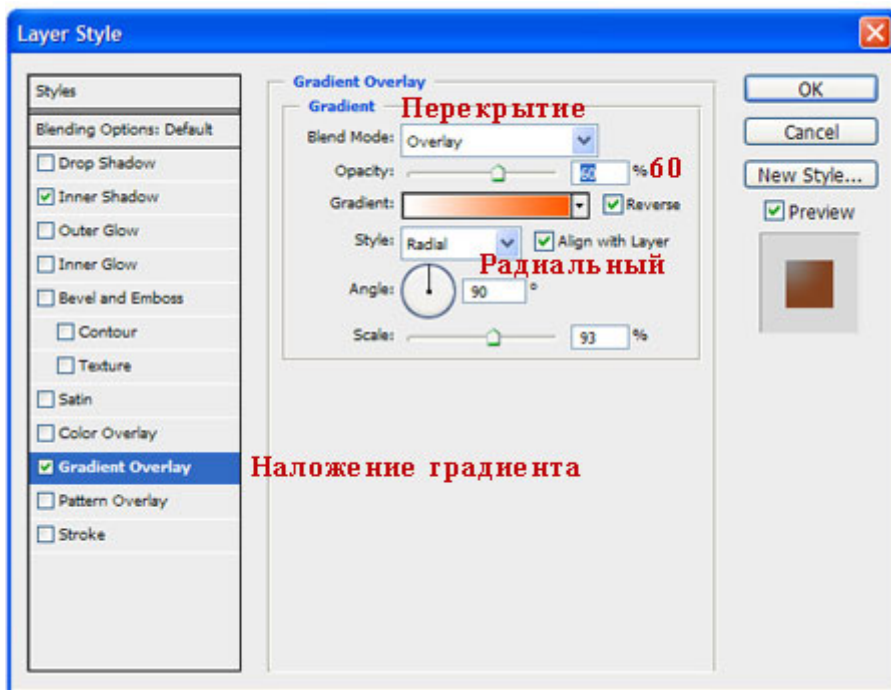
Далее, идём, **Фильтр- Имитация – Целлофановая упаковка** (Filter>Artistic>Plastic Wrap) и применяем следующие значения:



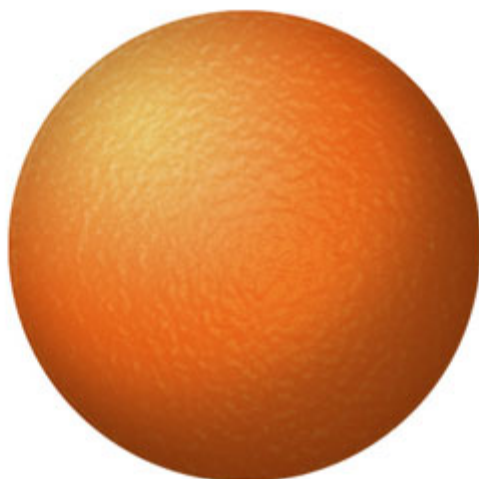
Шаг 5

Дважды щёлкните по слою, чтобы появилось окно **Стили слоя** (Layer Styles), далее, примените следующие опции:




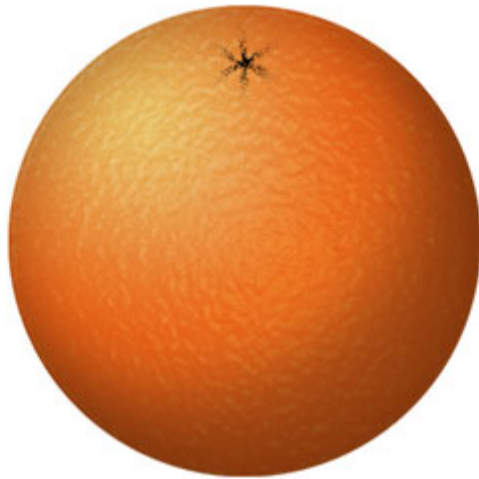


Результат должен быть, как скриншоте ниже.



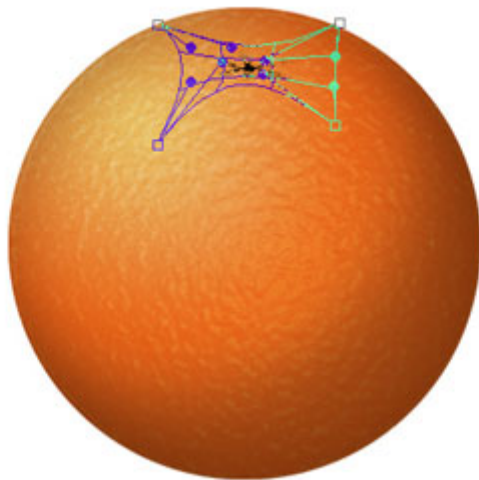
Шаг 6

Создайте новый слой. Выберите инструмент **Кисть**  (Paint Brush Tool), в настройках кисти установите кисть **'star 70 pixels'** (кисть 'звёздочка 70 пикселей'). Не перемещая кисть, просто щёлкните 3-4 раза, чтобы получилась красивая тёмная звёздочка;

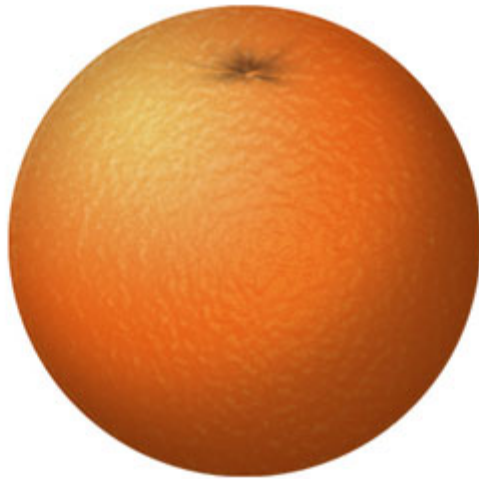




Шаг 7

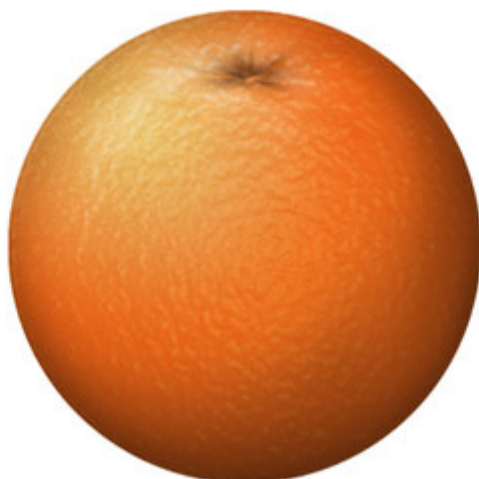
Трансформируйте звёздочку с помощью опции **Деформация (Warp Mode)**, для этого идём **Редактирование – Трансформация – Деформация (Edit>Transform>Warp)** и далее, деформируйте звёздочку, пока у вас не получится желаемый «апельсиновый пупок».



Поменяйте режим наложения для данного слоя на **Перекрытие (Overlay)**. Результат должен быть, как на скриншоте ниже



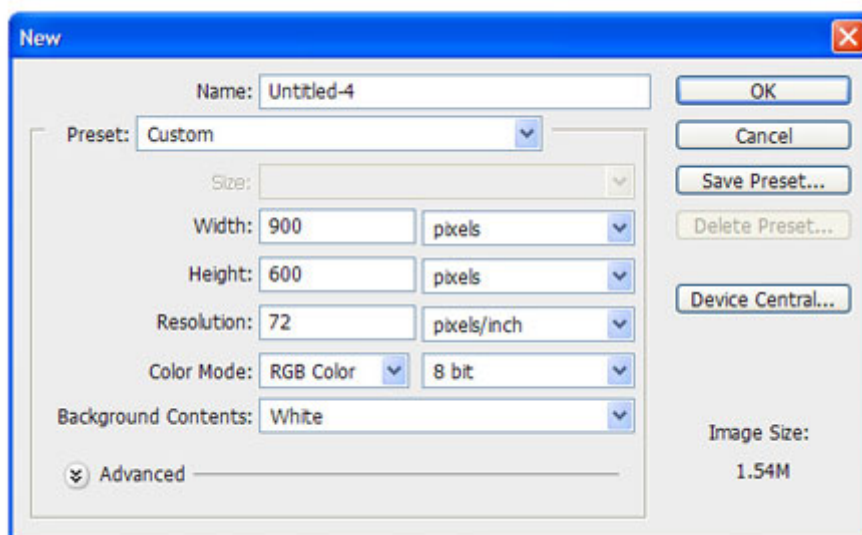
Для добавления некоторых деталей, используйте инструмент **Осветлитель**  и **Затемнитель**  (Dodge and Burn Tools) для создания светотеней.
Например:




Как нарисовать Банан

Шаг 1

Создайте новый документ в программе Photoshop, идём **Файл – Новый** (File>New). Для нашего урока, я применил следующие значения 900px x 600px (72 dpi).



Шаг 2

Создайте новый слой. С помощью инструмента **Прямоугольная область**  (Rectangular Marquee Tool), создайте прямоугольник. Залейте его жёлтым цветом.



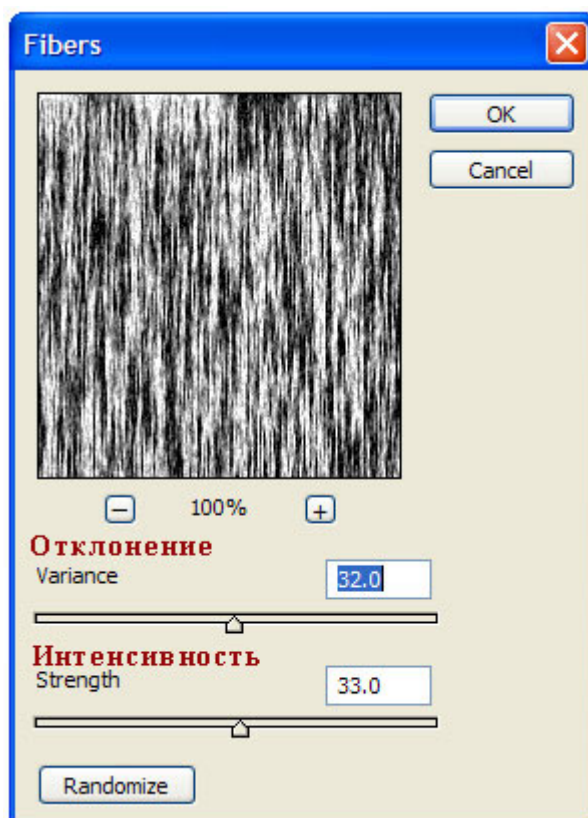
Шаг 3

Ещё раз создайте новый слой. Создайте второй прямоугольник, как показано на скриншоте ниже. Залейте его чёрным цветом.

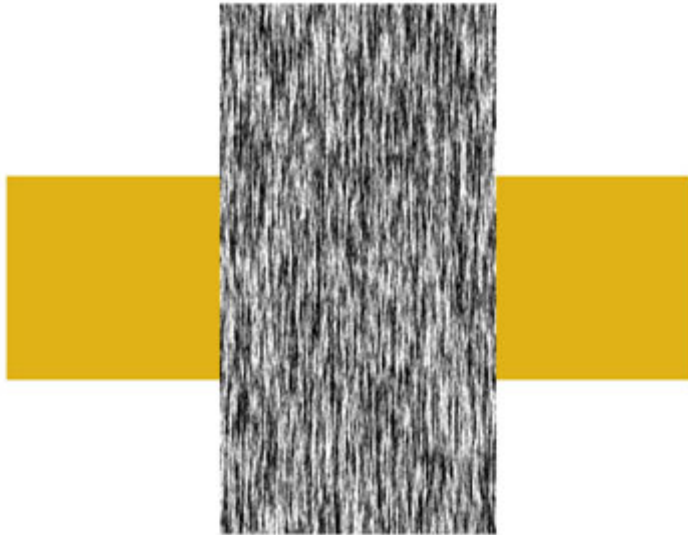


Шаг 4

Далее, идём **Фильтр – Рендеринг – Волокна** (Filter>Render>Fibers), а затем применяем следующие значения (не забудьте установить белый и чёрный цвет в качестве цветовых оттенков переднего и заднего плана):

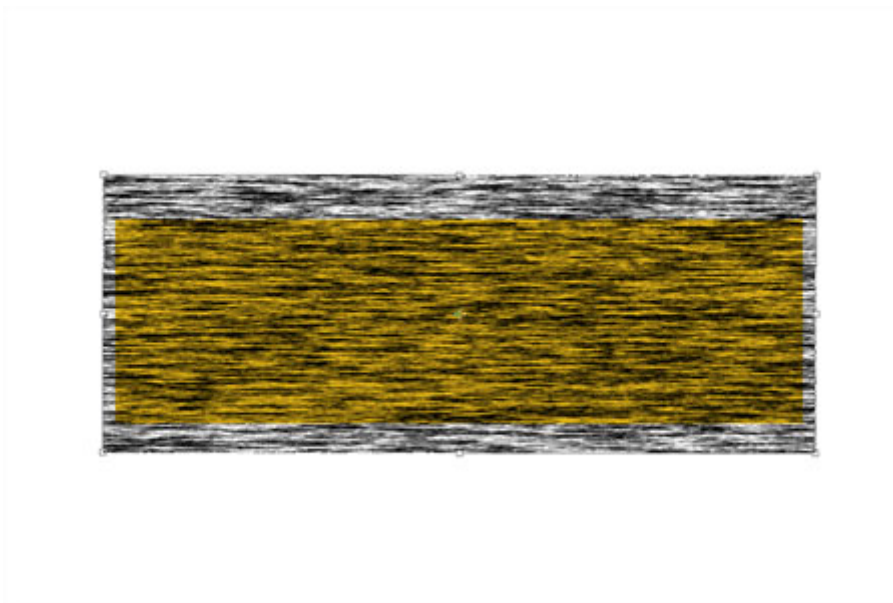


Результат должен быть, как скриншоте ниже.



Шаг 5

Поверните, а также примените трансформацию (Ctrl + T) к слою с текстурой волокна в соответствии с размерами желтого прямоугольника. Поменяйте режим наложения для слоя с текстурой волокна на **Умножение** (Multiply).



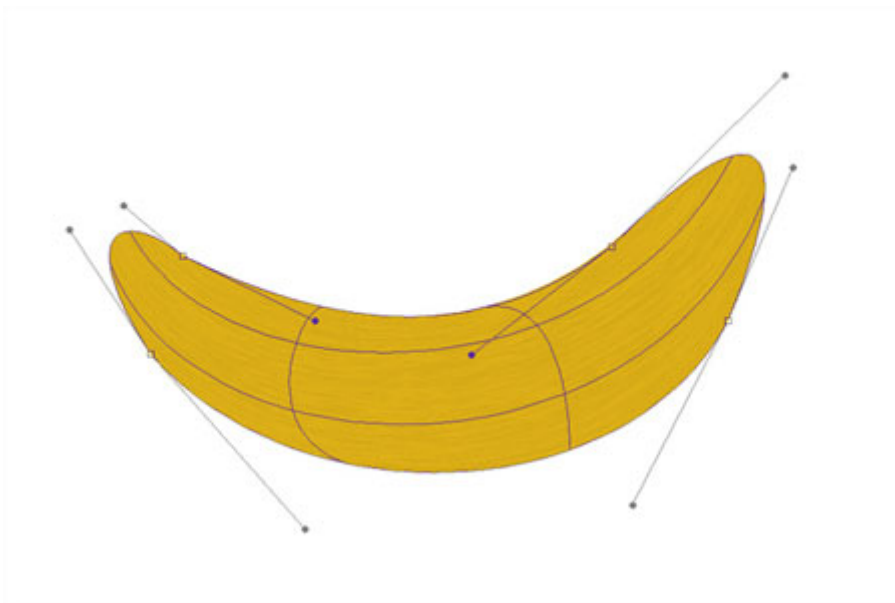
Шаг 6

Удерживая клавишу Alt + щёлкните между слоем с текстурой волокна и слоем с жёлтым прямоугольником, чтобы преобразовать слой с текстурой волокна в оброчную маску. Уменьшите непрозрачность слоя с текстурой волокна до, примерно, 6%.




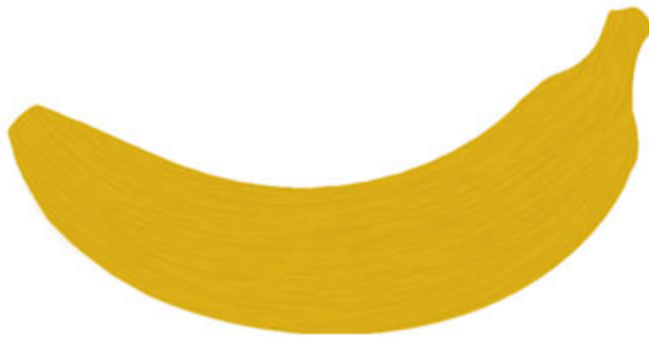
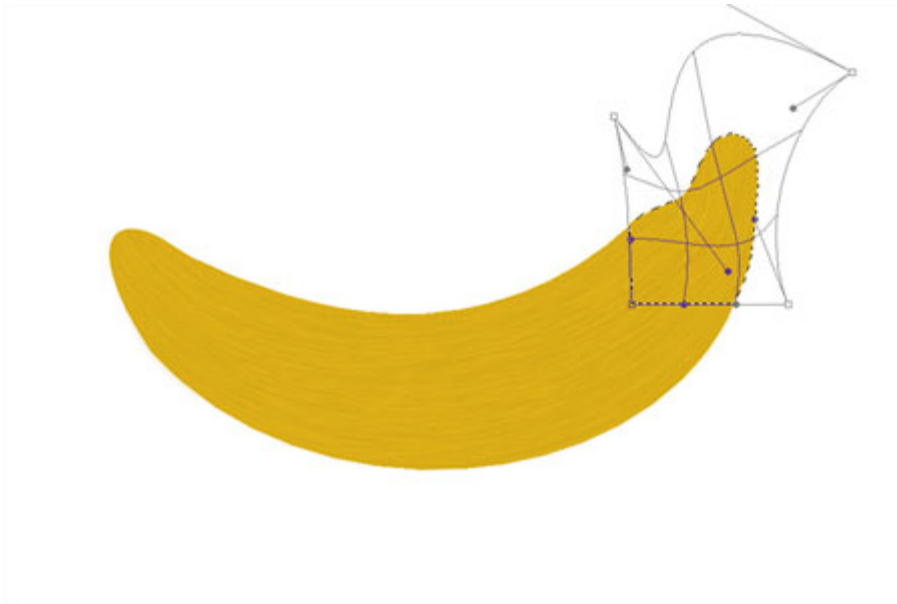
Шаг 7

Выделите оба слоя, а затем слейте их вместе (Ctrl + E). С помощью **Деформации** (Warp Mode), придайте жёлтому прямоугольнику форму банана, для этого идём **Редактирование – Трансформация – Деформация** (Edit>Transform>Warp):





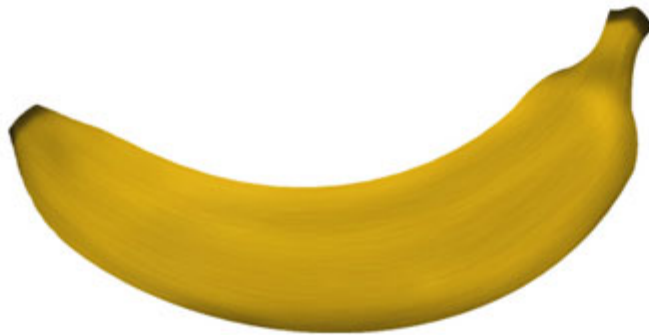
Шаг 8

Доработайте форму с помощью опции **Деформации** (Warp Mode), на кончике банана по мере необходимости. Также, с помощью инструмента **Ластик**  (Eraser tool), можно придать необходимые грани:




Шаг 9

С помощью инструментов **Осветлитель**  и **Затемнитель**  (Dodge and Burn Tools), создайте светотени на нашем банане. Сфокусируйтесь на кончиках банана, они обычно намного темнее.



Шаг 10

Создайте новый слой. Выберите инструмент **Овальная область**  (Elliptical Marquee Tool), чтобы создать овальную область. Залейте созданную овальную область белым цветом. Убедитесь, чтобы белый овал закрывал чуть меньше, чем на половину поверхность банана, как показано на скриншоте ниже:



Шаг 11

Преобразуйте данный слой с белым овалом в обтравочную маску к слою с бананом (Alt+щёлкните между двумя слоями в палитре слоёв). Также, поменяйте режим наложения для данного слоя на **Освещение** (Screen), а также уменьшите непрозрачность слоя до 25%.



Шаг 12

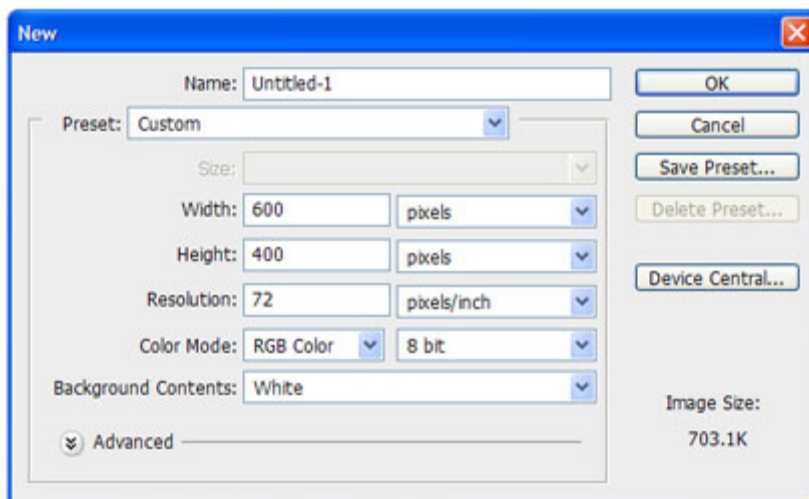
При необходимости, вы можете применить размытие или инструмент **Палец** (smudge) к кромке овала, чтобы смягчить кромку для естественности. Также, для детализации, вы можете добавить к изображению банана оттенки зелёного цвета (создайте новый слой и на новом слое, с помощью мягкой кисти бледно-зелёного цвета, прокрасьте штрихами по поверхности банана. Не забудьте преобразовать данный слой в обтравочную маску и поменять режим наложения на **Цветность** (color). А затем, примените **Размытие по Гауссу** (Gaussian Blur), чтобы смягчить цвет).




Как нарисовать Виноград

Шаг 1

Создайте новый документ в программе Photoshop, идём **Файл – Новый** (File>New). Для нашего урока, я применил следующие значения 600px x 400px (72 dpi).



Шаг 2

Создайте новый слой. С помощью инструмента **Овальная область**  (Elliptical Marquee Tool), нарисуйте овал. Залейте нарисованный овал серым цветом средних тонов (например: R=91 G=91 B=91)

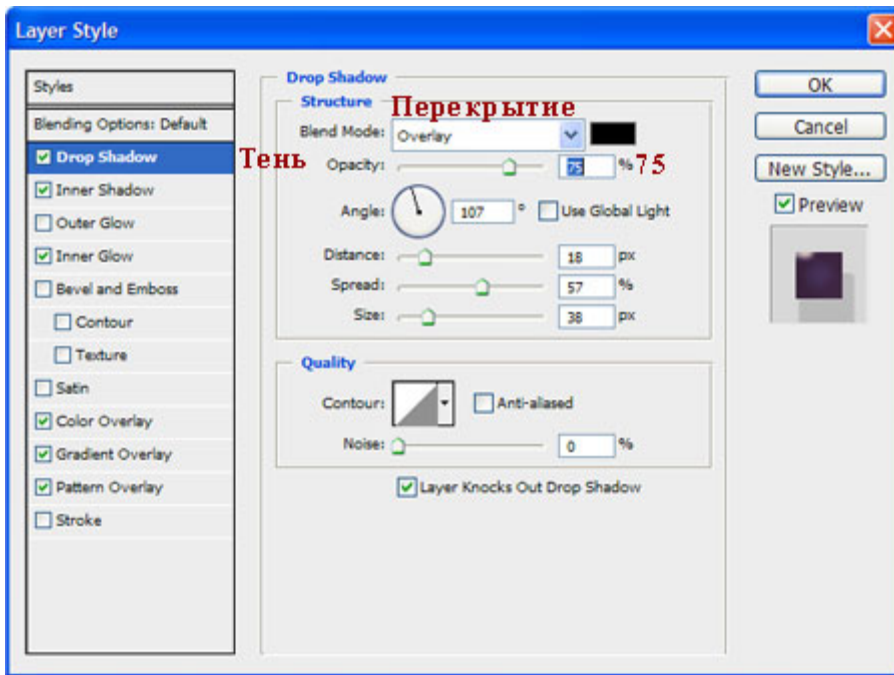


Шаг 3

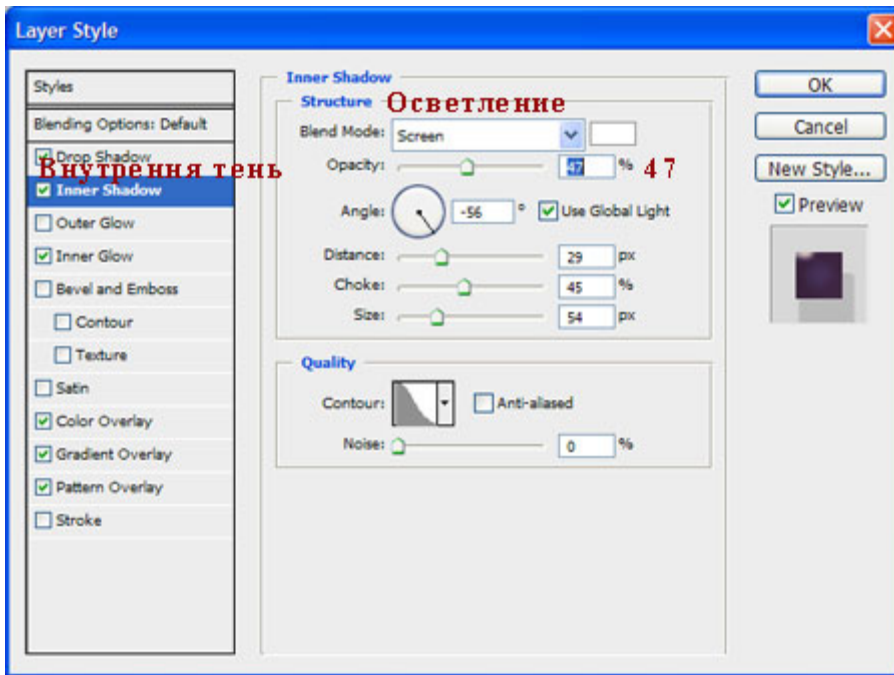
Дважды щёлкните по слою, чтобы появилось окно **Стили слоя** (Layer Styles), далее, примените следующие опции:

Глобальное Освещение (Global Lights)!

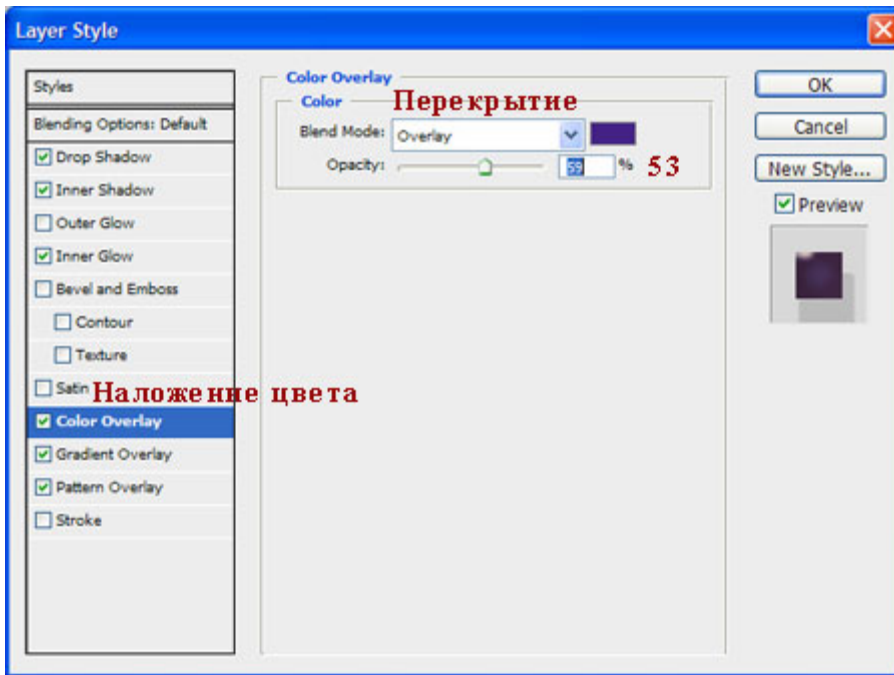
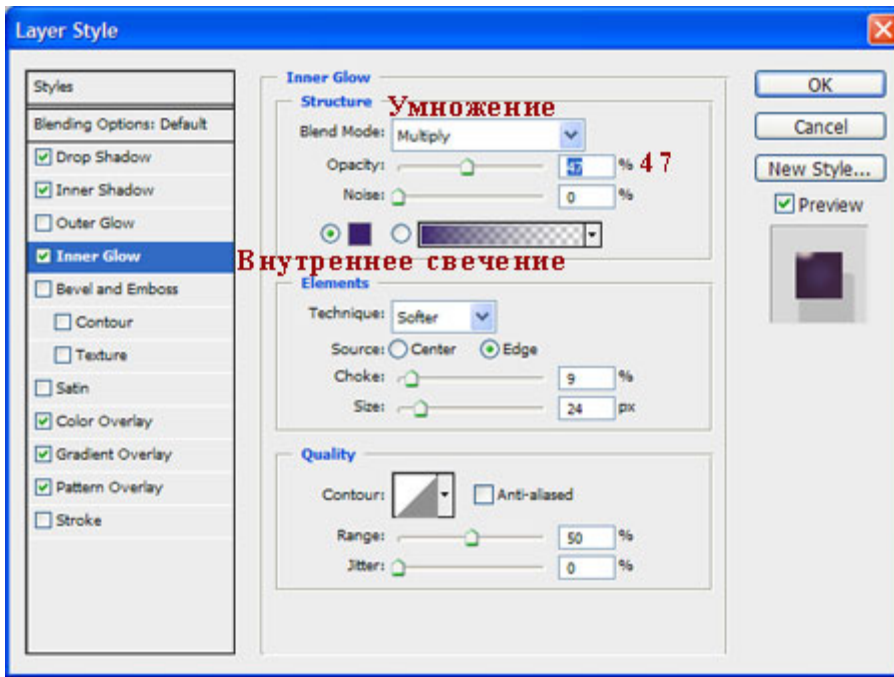
Очень важно, чтобы вы обратили внимание, когда ставить галочку в окошке **Глобальное Освещение** (Global Lights), а когда нет.

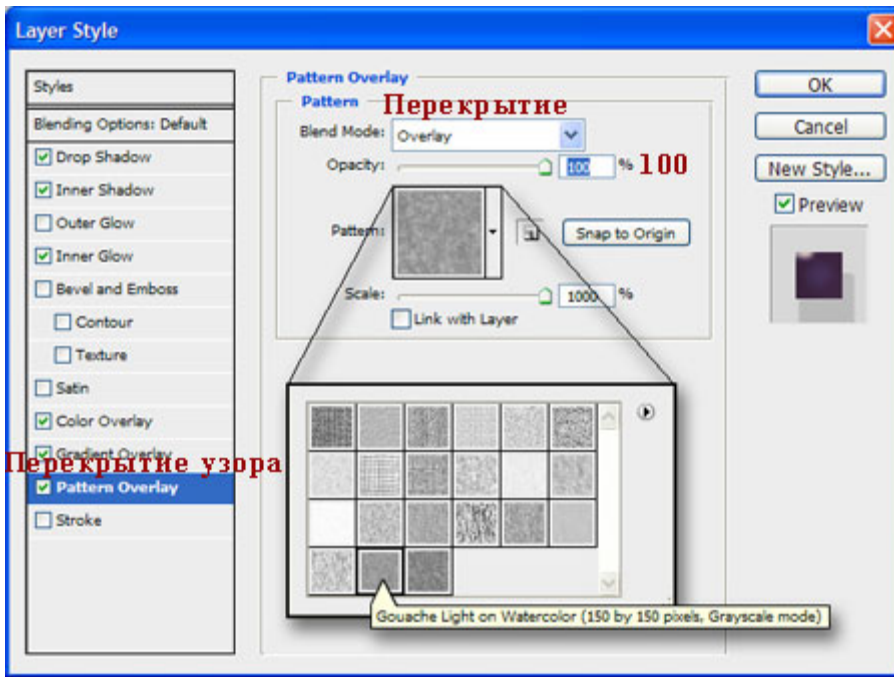
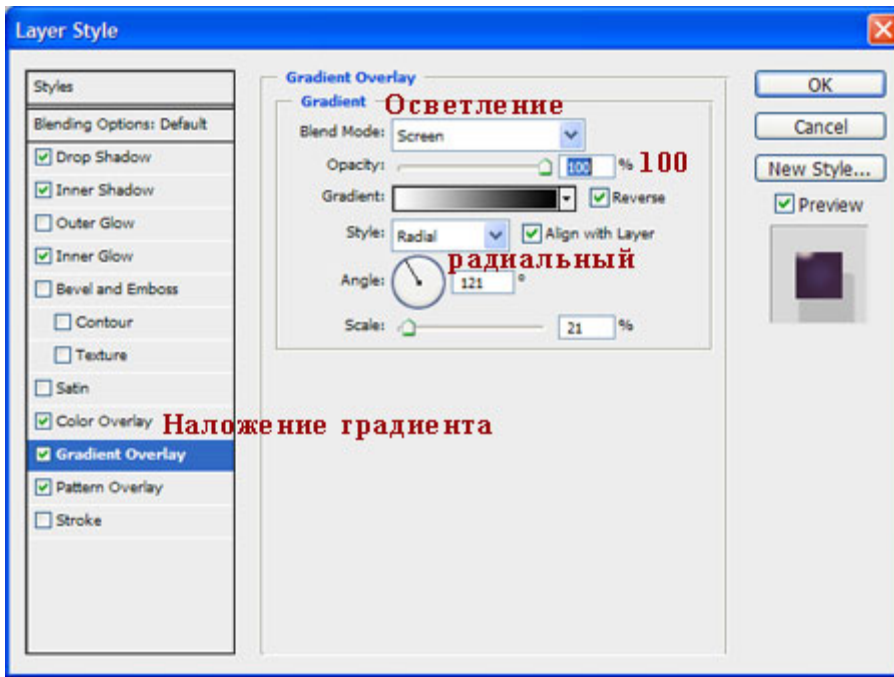


Тень



Внутренняя тень



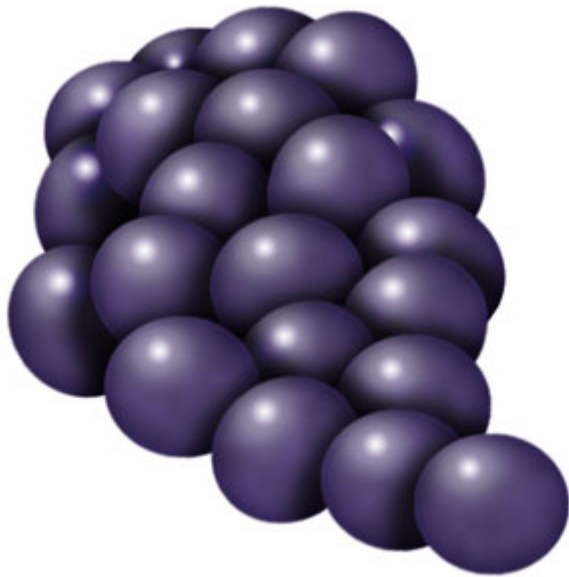


Ваша виноградка должна выглядеть, как на скриншоте ниже:



Шаг 4

Продублируйте слой с виноградинкой столько раз, сколько вы пожелаете (для дублирования изображения, просто удерживайте клавишу Alt + перемещайте мышкой копию изображения виноградинки). Также для разнообразия, вы можете развернуть (Ctrl+ T) виноградинки в различных направлениях.



Итоговый результат



Лабораторная работа №5.

Быстрый эффект карандашного рисунка

Ищите способ, как превратить фотографию в карандашный рисунок? Урок для начинающих покажет вам, как с помощью трёх слоёв вы сможете создать данный эффект.

Итоговый результат





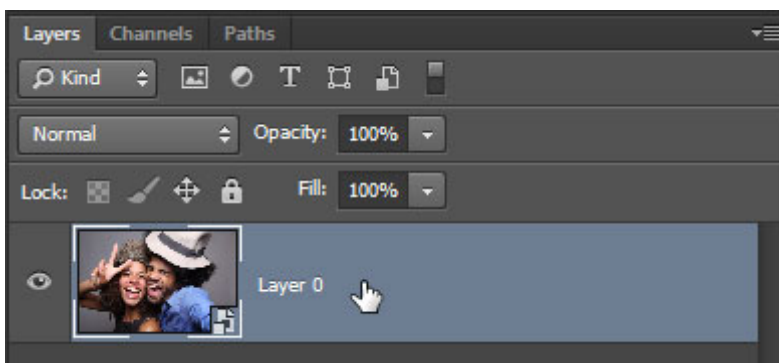
Шаг 1

На скриншоте ниже фотография, которую мы используем в этом уроке. Для получения наилучших результатов, вы должны использовать фотографию с разрешением примерно 6 мега пикселей или ниже (3000×2000 px).



Шаг 2

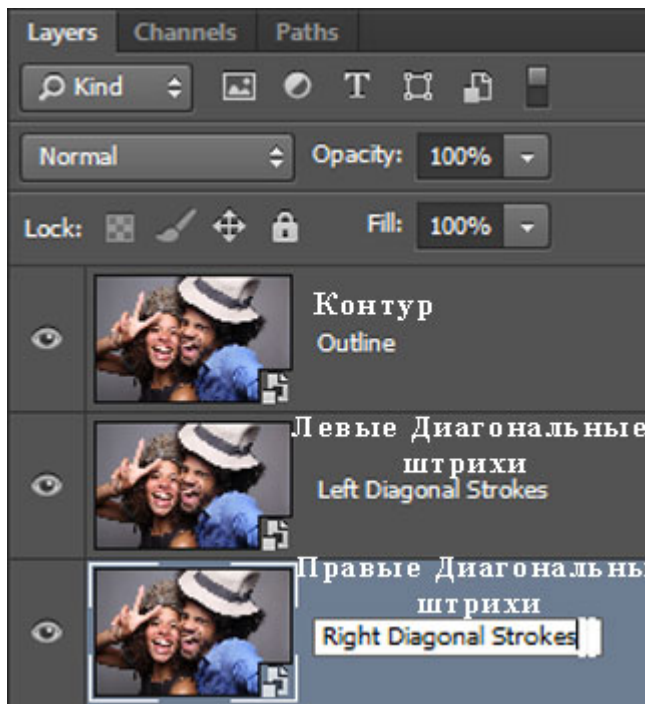
Мы создадим эффект, используя неразрушающий метод, при котором используется минимальное количество слоёв и который позволяет проводить дополнительную коррекцию в любое время. Вначале преобразуйте наше исходное изображение в **Смарт-объект (Smart Object)**, для этого идём **Слой- Смарт-Объект – Преобразовать в смарт-объект (Layer > Smart Objects > Convert to Smart Object)**. Преобразовав слой с изображением в смарт-объект, позволит вам применять **Смарт Фильтры (Smart Object)**, настройки которых вы можете поменять в любое время.



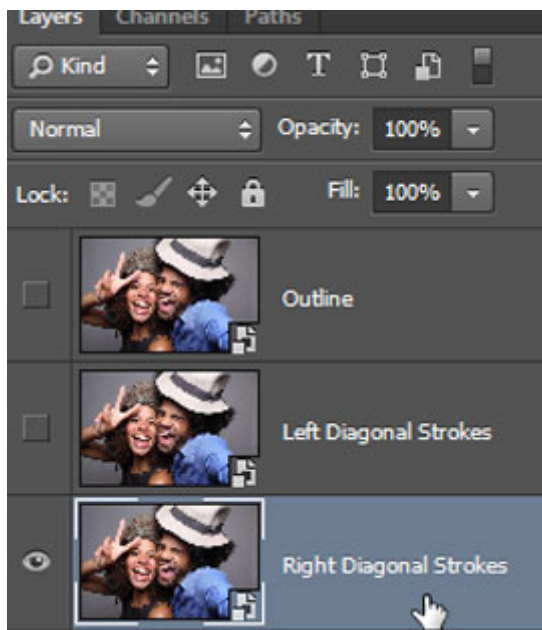
Смарт-объект (Smart Objects) можно определить по появившемуся значку в нижнем правом углу миниатюры слоя.

Шаг 3

Дважды продублируйте слой, далее, переименуйте слои **Контур (Outline)**, **Левые Диагональные штрихи (Left Diagonal Strokes)**, и **Правые Диагональные Штрихи (Right Diagonal Strokes)**, назвав слои сверху вниз соответственно.



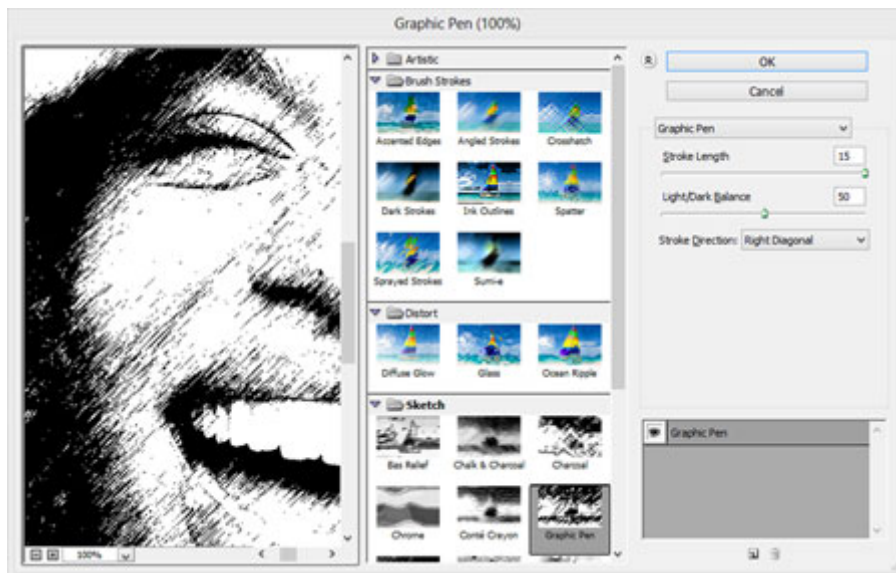
Скройте видимость двух верхних слоёв, перейдите на нижний слой.



Шаг 4

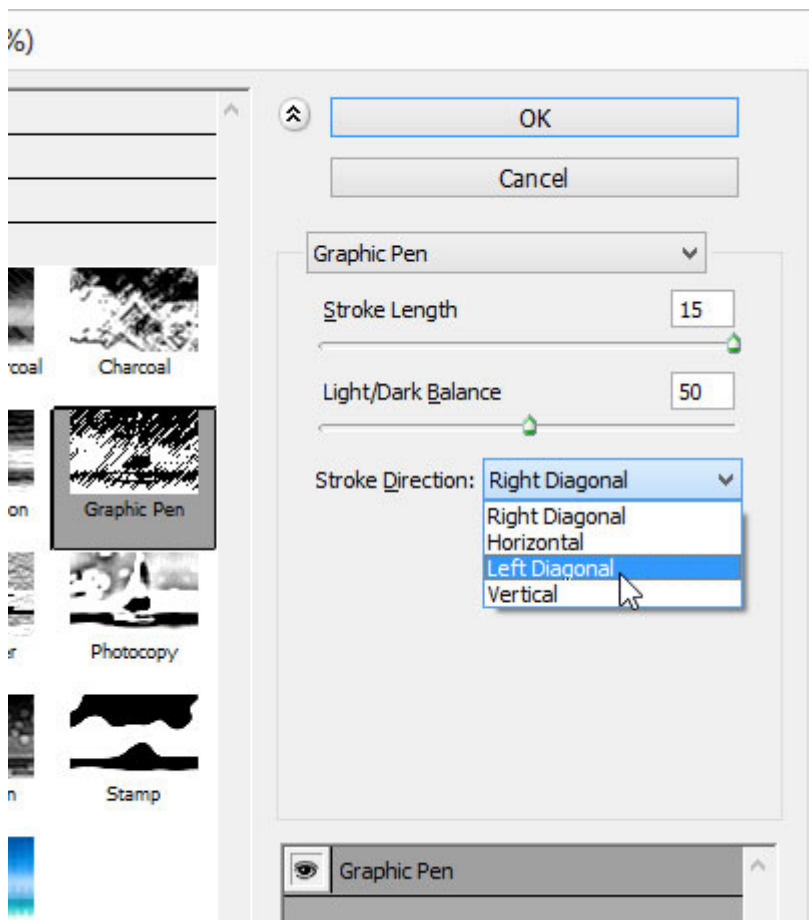
Находясь на самом нижнем слое, идём **Фильтр- Галерея фильтров** (Filter > Filter Gallery). В папке **Эскиз** (Sketch), выберите фильтр **Тушь** (Graphic Pen), в настройках **Направление штрихов** (Stroke Direction), установите опцию **По диагонали вправо** (Right Diagonal), далее отрегулируйте настройки **Длины штриха** (stroke length) и **Тоновый баланс** (light/dark balance). Нажмите ОК, чтобы применить выставленные значения.

*Примечание переводчика: настройки автора **Длины штриха** (stroke length) 15, **Тоновый баланс** (light/dark balance) 50.*



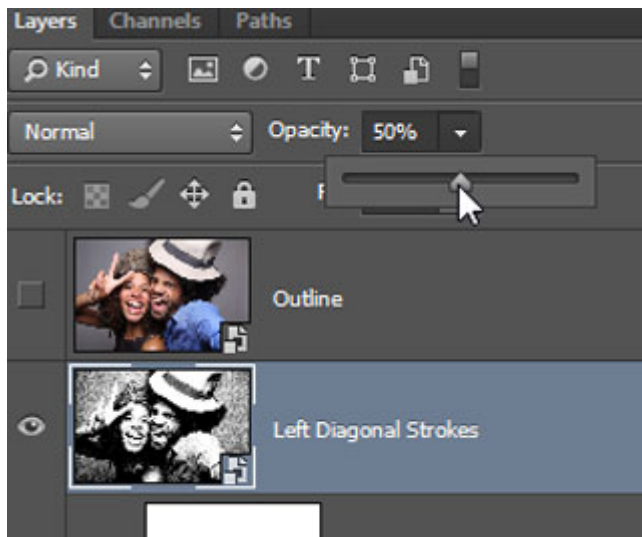
Шаг 5

Включите видимость слоя **Левые Диагональные штрихи (Left Diagonal Strokes)**, находясь на данном слое, повторите предыдущий шаг, но на этот раз установите опцию **По диагонали влево (Left Diagonal)** в настройках **Направление штрихов (stroke direction)**.



Шаг 6

Уменьшите непрозрачность текущего слоя **Левые Диагональные штрихи (Left Diagonal Strokes)** до 50%.



Результат на данный момент.



Шаг 7

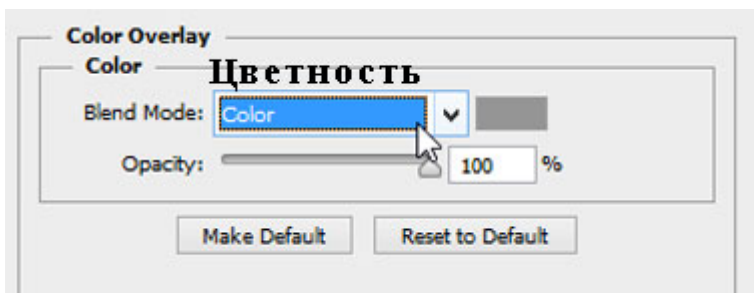
Теперь мы добавим контур. Включите видимость третьего слоя **Контур (Outline)**, находясь на данном слое, идём **Фильтр – Стилизация – Выделение краёв (Filter > Stylize > Find Edges)**. Поменяйте режим наложения для данного слоя на **Умножение (Multiply)**. На изображении появились отдельные цветовые оттенки, которые не очень привлекательны. Поэтому, нам необходимо избавиться от цветовых оттенков в следующем шаге.



Шаг 8

Вы не можете добавить фильтр **Обесцветить** (Desaturate) или **Цветовой тон / Насыщенность** (Hue/Saturation) прямо на **Смарт-фильтр**(Smart Object), тем не менее, вы можете добавить их в качестве корректирующих слоёв через **Слой – Новый корректирующий слой** (Layers > New Adjustment Layer). Существует ещё один интересный способ – это использование **Стилей слоя** (layer styles), чтобы обесцветить слой. Лично я предпочитаю данный метод, т.к. он не требует добавления дополнительного слоя.

Для обесцвечивания используем **Стили слоя** (layer styles), применяем опцию **Наложение цвета** (Color Overlay), идём **Слой – Стили слоя – Наложение цвета** (Layer > Layer Style > Color Overlay). Установите цвет белый, серый или чёрный (без разницы какой цвет), далее, поменяйте режим наложения на **Цветность** (Color). Нажмите ОК, чтобы применить выставленные значения, и цвета исчезнут!

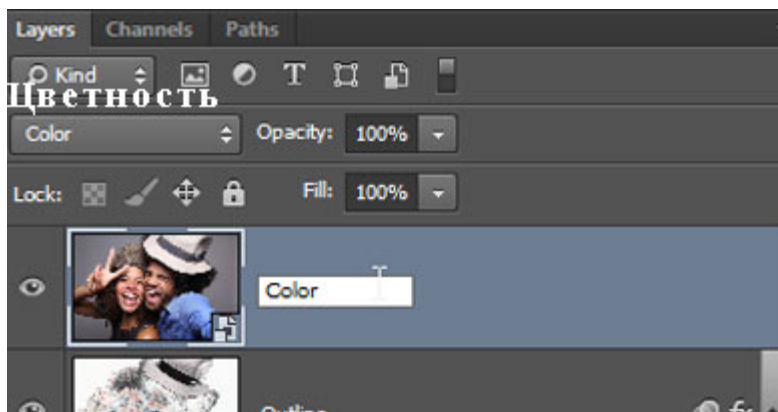


Мы завершили урок! Изображение будет выглядеть, как на скриншоте ниже. Следующий шаг является выборочным, но вы должны попробовать его выполнить, если вы хотите создать эффект карандашного рисунка в цвете.



Шаг 9

Чтобы цвета вновь появились, продублируйте слой **Контур** (Outline), далее, удалите стили слоя и смарт-фильтр, для этого вначале идём **Слой– Стили слоя – Очистить стили слоя** (Layer > Layer Style > Clear Layer Styles), а затем идём **Слой- - Смарт-фильтры - Снять смарт-фильтр** (Layer > Smart Filters > Clear Smart Filters). Поменяйте режим наложения для данного дубликата слоя на **Цветность** (Color), переименуйте слой в “Цветность”, и мы закончили урок!



Итоговый результат



3. Советы по подготовке к текущему, промежуточному и итоговому контролю по дисциплине

Изучение дисциплины заканчивается определенными методами контроля, к которым относятся: сдача всех лабораторных работ, получение зачета.

При подготовке к зачету у студента должен быть учебник, лекции, методические рекомендации.

Первоначально следует просмотреть весь материал по сдаваемой дисциплине, отметить для себя трудные вопросы. Обязательно в них разобраться. Выполнить все лабораторные работы

по темам. В заключение еще раз целесообразно повторить основные положения, используя при этом опорные конспекты лекций.

Систематическая подготовка к занятиям в течение семестра, регулярная выполнение лабораторных работ позволит использовать время зачетной недели для систематизации знаний. Если в процессе самостоятельной работы над изучением теоретического материала или при решении задач у студента возникают вопросы, разрешить которые самостоятельно не удастся, необходимо обратиться к преподавателю для получения у него разъяснений или указаний. В своих вопросах студент должен четко выразить, в чем он испытывает затруднения, характер этого затруднения. За консультацией следует обращаться и в случае, если возникнут сомнения в правильности выполнения лабораторных работ.