



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

**Э.В. Денисова**

# **Информатика. Базовый курс**

Учебное пособие



**Санкт-Петербург**

**2013**

Э.В. Денисова, Информатика. Базовый курс: Учебное пособие. – СПб: СПбГУ ИТМО, 2013. – 70с.

В пособии изложено содержание семестрового учебного курса «Информатика». Учебный курс читается в 1 семестре первого года обучения для слушателей кафедры информатики и прикладной математики, кафедры оптических технологий, кафедры оптико-электронных приборов и систем, кафедры прикладной и компьютерной оптики, кафедры компьютеризации и проектирования оптических приборов.

Учебно-методическое пособие адресовано студентам обучающимся по специальностям:

231000.62 - Программная инженерия

231000.62.01 - Разработка программно-информационных систем

230100.62 - Информатика и вычислительная техника

230100.62.01 - Вычислительные машины, комплексы, системы и сети

200400.62 - Оплотехника

200400.62.05 - Оптико-электронные приборы и системы

200400.62.07 - Оптические технологии и материалы

200400.62.08 - Прикладная и компьютерная оптика

200400.62.10 - Проектирование и метрология оптико-электронных приборов специального назначения

Рекомендовано к печати Советом факультета Компьютерных Технологий и Управления, протокол № 3 от 12.03.2013г.



В 2009 году Университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена программа его развития на 2009–2018 годы. В 2011 году Университет получил наименование «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»

© Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, 2013

© Э.В. Денисова, 2013

## СОДЕРЖАНИЕ

Глава 1. Системы счисления	4
Глава 2. Определение объема информации	8
Глава 3. Операционная система	11
Глава 4. Информационные технологии	16
Глава 5. Технологии обработки графической информации	26
Глава 6. Расширенные возможности Word	31
Глава 7. Практическая работа с текстовым документом в процессоре Word	59

# Глава 1. Системы счисления

## Позиционные системы счисления

При переводе числа из одной системы счисления в другую используются различные приемы, позволяющие сделать это достаточно быстро.

1) Перед началом решения задач этого типа необходимо затратить несколько минут на составление таблицы соответствия чисел десятичной, двоичной, четверичной, восьмеричной и шестнадцатеричной систем счисления. Это поможет достаточно быстро осуществить перевод числа из двоичной системы счисления в соответствующую систему счисления.

- Для перевода из двоичной системы счисления в четверичную необходимо исходное двоичное число разбить на группы, начиная справа по два символа, затем каждую группу перевести в четверичное число.
- Для перевода из двоичной системы счисления в восьмеричную необходимо исходное двоичное число разбить на группы (триады), начиная справа по три символа, затем каждую группу перевести в восьмеричное число.
- Для перевода из двоичной системы счисления в шестнадцатеричную необходимо исходное двоичное число разбить на группы (тетрады), начиная справа по четыре символа, затем каждую группу перевести в шестнадцатеричное число.

*Примеры:*

$$537,1_8 = 101\ 011\ 111,001_2 ; 1A3, F_{16} = 1\ 1010\ 0011, 1111_2$$

↓	↓	↓	↓		↓	↓	↓	↓
5	3	7	1		1	A	3	F

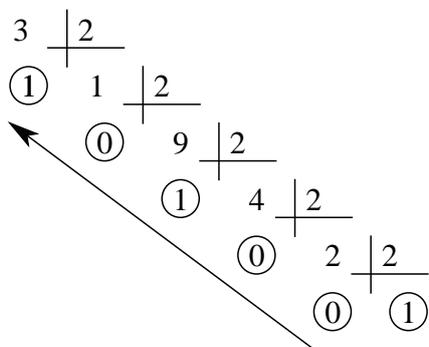
$$10101001,10111_2 = 10\ 101\ 001, 101\ 110_2 = 251,56_8$$

↓	↓	↓	↓	↓
2	5	1	5	6

$$10101001,10111_2 = 1010\ 1001, 1011\ 1000_2 = A9, B8_{16}$$

↓	↓	↓	↓
A	9	B	8

2) При переводе целого десятичного числа в систему с основанием  $q$  его необходимо последовательно делить на  $q$  до тех пор, пока не останется остаток, меньший или равный  $q-1$ . Число в системе с основанием  $q$  записывается как последовательность остатков от деления, записанных в обратном порядке, начиная с последнего.



$$A_{10} = 37 \quad A_2 = 100101$$

При переводе числа из десятичной системы счисления в двоичную систему можно так же пользоваться следующим способом записи:

Исходное число делят на 2, результат пишут под исходным числом, а справа от черты в строке с исходным числом ставят - 0, если деление без остатка, и 1, если остаток есть. Деление повторяют до тех пор, пока делимое не станет меньше делителя. Считывание результата производится снизу вверх.

3) При переводе числа из двоичной (восьмеричной, шестнадцатеричной) системы счисления в десятичную надо это число представить в виде суммы степеней основания его системы счисления.

*Примеры:*

$$1011,1_2 = 1 \cdot 2^3 + 1 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} = 11,5_{10}.$$

$$276,5_8 = 2 \cdot 8^2 + 7 \cdot 8^1 + 6 \cdot 8^0 + 5 \cdot 8^{-1} = 190,625_{10}.$$

$$1F3_{16} = 1 \cdot 16^2 + 15 \cdot 16^1 + 3 \cdot 16^0 = 499_{10}.$$

4) При переводе правильной десятичной дроби в систему счисления с основанием  $q$  необходимо сначала саму дробь, а затем дробные части всех последующих произведений последовательно умножать на  $q$ , отделяя после каждого умножения целую часть произведения. Число в новой системе счисления записывается как последовательность полученных целых частей произведения.

Умножение производится до тех пор, пока дробная часть произведения не станет равной нулю. Это значит, что сделан точный перевод. В противном случае перевод осуществляется до заданной точности. Достаточно того количества цифр в результате, которое поместится в ячейку.

*Пример:*

Перевести число 0,25 из десятичной системы в двоичную систему:

$$0,25 \cdot 2 = 0,5 \rightarrow 0$$

$$0,5 \cdot 2 = 1,0 \rightarrow 1$$

Ответ:  $0,25_{10} = 0,01_2$

Если дробь имеет целую и дробную часть, целую часть необходимо делить на основание  $q$ , а дробную умножать на это основание.

## **Арифметические операции в позиционных системах счисления**

Правила выполнения арифметических операций сложения, вычитания, умножения и деления в двоичной, восьмеричной и шестнадцатеричной системах счисления такие же, как и в десятичной системе. Иногда полезно пользоваться особыми для каждой системы таблицами сложения и умножения, которые могут быть составлены заранее.

Например, таблица для двоичной системы счисления имеет вид:

Сложение	Вычитание	Умножение
$0 + 0 = 0$	$0 - 0 = 0$	$0 \times 0 = 0$
$0 + 1 = 1$	$1 - 0 = 1$	$0 \times 1 = 0$
$1 + 0 = 1$	$1 - 1 = 0$	$1 \times 0 = 0$
$1 + 1 = 10$	$10 - 1 = 1$	$1 \times 1 = 1$

Единица – перенос в старший разряд

При выполнении всех операций необходимо помнить, действия выполнять можно в десятичной системе счисления, но перенос разряда необходимо

производить при достижении соответствующего максимума основания. Полезно так же, в случае появления сомнений, производить проверку результата проведением соответствующих действий в десятичной системе и переводом полученных чисел в необходимую систему счисления.

*Пример.* Выполнить операцию арифметического сложения в двоичной системе счисления чисел 13 и 7.

$$\begin{array}{r}
 13_{10} = 1101_2 \\
 \text{Решение: } + \quad 13 \\
 \qquad \qquad \quad 7 \\
 \hline
 \qquad \qquad 20_{10}
 \end{array}
 \qquad
 \begin{array}{r}
 7_{10} = 0111_2 \\
 \qquad \qquad + \quad 01101 \\
 \qquad \qquad \quad \underline{00111} \\
 \qquad \qquad 10100_2
 \end{array}$$

При сложении двух единиц результат операции равен нулю и единица переносится в соседний разряд.

$$10100_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 20_{10}$$

*Пример.* Выполнить операцию арифметического вычитания в двоичной системе счисления чисел 13 и 7.

$$\begin{array}{r}
 \text{Решение: } \_ \quad 13 \\
 \qquad \qquad \quad 7 \\
 \hline
 \qquad \qquad 6_{10}
 \end{array}
 \qquad
 \begin{array}{r}
 \_ \quad 1101 \\
 \qquad \quad \underline{0111} \\
 \qquad \quad 0110_2
 \end{array}$$

$$0110_2 = 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 6_{10}$$

## Непозиционные системы счисления

Методы решения задач с римскими цифрами, которые так же относятся непозиционной системе счисления, как правило не вызывают затруднений. Особенностью их решения является внимательность и знание основных цифр.

Для решения задач с римскими числами необходимо знать, что запись чисел в этой системе счисления осуществляется по следующим правилам:

1) если цифра слева меньше, чем цифра справа, то левая цифра вычитается из правой (IV:  $1 < 5$ , следовательно,  $5 - 1 = 4$ , XL:  $10 < 50$ , следовательно,  $50 - 10 = 40$ );

2) если цифра справа меньше или равна цифре слева, то эти цифры складываются (VI:  $5 + 1 = 6$ , VIII:  $5 + 1 + 1 + 1 = 8$ , XX:  $10 + 10 = 20$ );

*Пример:*

1964 = MCMLXIV

M – 1000, CM – 900, LX – 60, IV – 4, здесь «девятьсот» получается посредством вычитания из «тысячи» числа «сто», «шестьдесят» — посредством сложения «пятидесяти» и «десяти», «четыре» — посредством вычитания из «пяти» «единицы».

## Глава 2. Определение объема информации

Данный тип задач содержит несложные задания в одно-два действия и нацелен на проверку умений подсчета информационного объема сообщения и количества информации.

Для решения задач на определение объема информации, необходимо учитывать, каким алфавитом закодирована информация, т.к. единица измерения информации зависит от конкретного алфавита. При подсчете количества информации особое внимание надо обратить на единицы измерения и применяемые таблицы кодировок (Unicode, ASCII, КОИ-8 и т.д.).

Обычно при решении задач такого вида делают две самые типичные ошибки: невнимательно читают задание и начинают считать не в нужной кодировке. Сосчитав количество символов в байтах и не найдя нужного ответа, решают, что задание содержит ошибку, и даже не пытаются перевести байты в биты или наоборот.

Необходимо помнить что: в 1 байте 8 бит, в 1 кб (килобайте) 1024 бит, в 1 Мб (Мегабайте) 1024 кб (килобайт).

Разберем несколько примеров решения задач такого типа.

*Пример 1:*

Автоматическое устройство осуществило перекодировку информационного сообщения на русском языке, первоначально записанного в 16-битном коде Unicode, в 8-битную кодировку КОИ-8. При этом информационное сообщение уменьшилось на 720 бит. Какова длина сообщения в символах?

Решение:

При переходе от 16-битной к 8-битной кодировке, размер сообщения в битах уменьшится вдвое. Раз оно уменьшилось на 720 бит, то длина сообщения в символах кодировки КОИ-8 составила 720 символов.

*Пример 2:*

В кодировке Unicode на каждый символ отводится два байта. Определите информационный объем слова из двадцати четырех символов в этой кодировке.

Решение:

$24 * 2\text{байта} = 48\text{ байтов} = 384\text{ бита}$

*Пример 4:*

В велокроссе участвуют 779 спортсменов. Специальное устройство регистрирует прохождение каждым из участников промежуточного финиша, записывая его номер с использованием минимально возможного количества бит, одинакового для каждого спортсмена. Каков информационный объем сообщения, записанного устройством, после того как промежуточный финиш прошли 280 велосипедистов?

Решение:

Справедлива оценка  $2^9 < 779 < 2^{10}$ , поэтому для равномерного двоичного кодирования номера каждого спортсмена требуется 10 бит информации. Тогда информационный объем сообщения, записанного устройством, после того как

промежуточный финиш прошли 280 велосипедистов, составляет 2800 бит=350 байт.

*Пример 5:*

Одна ячейка памяти «троичной ЭВМ» (компьютера, основанного на использовании троичной системы счисления) может принимать одно из трех возможных состояний. Для хранения некоторой величины отвели 6 ячеек памяти. Сколько различных значений может принимать эта величина?

Решение:

Мощность алфавита троичной ЭВМ  $N$  равно 3. Для 6 ячеек памяти количество различных значений оценивается величиной  $N^6 = 3^6 = 729$

*Пример 6:*

Скорость передачи данных через ADSL-соединение равна 128000 бит/с. Передача файла через это соединение заняла 2 минуты. Определите размер файла в килобайтах.

Решение:

$$128000 * 120 = 15360000(\text{бит}) = \frac{15360000}{8 * 1024} (\text{Килобайт}) = 1875(\text{Килобайт})$$

### Глава 3. Операционная система

*Операционная система (ОС)* - это упорядоченная последовательность системных управляющих программ совместно с необходимыми информационными массивами, предназначенная для планирования исполнения пользовательских программ и управления всеми ресурсами вычислительной машины (программами, данными, аппаратурой, и др.) с целью предоставления возможности пользователям эффективно (в некотором смысле) решать задачи, сформулированные в терминах вычислительной системы.

Вообще, ОС - это довольно расплывчатое понятие, опирающееся на определенные традиции, параметры оборудования, размеры программ, реализующих определенные функции, а также распределение

- обеспечивать загрузку пользовательских программ в оперативную память и их исполнение
- обеспечивать работу с устройствами долговременной памяти, такими как магнитные диски, ленты, оптические диски и т.д. Как правило, ОС управляет свободным пространством на этих носителях и структурирует пользовательские данные
- предоставлять более или менее стандартный доступ к различным устройствам ввода/вывода, таким как терминалы, модемы, печатающие устройства
- предоставлять некоторый пользовательский интерфейс. Слово некоторый здесь сказано не случайно - часть систем ограничивается командной строкой, в то время как другие на 90% состоят из средств интерфейса пользователя

Существуют ОС, функции которых этим и исчерпываются. Одна из хорошо известных систем такого типа - дисковая операционная система *MS DOS*.

Более развитые ОС предоставляют также следующие возможности:

- параллельное (точнее, псевдопараллельное, если машина имеет только один процессор) исполнение нескольких задач
- распределение ресурсов компьютера между задачами
- организация взаимодействия задач друг с другом
- взаимодействие пользовательских программ с нестандартными внешними устройствами
- организация межмашинного взаимодействия и разделения ресурсов
- защита системных ресурсов, данных и программ пользователя, исполняющихся процессов и самой себя от ошибочных и зловредных действий пользователей и их программ

## **Файловая система**

*Файловая система* - это часть операционной системы, назначение которой состоит в том, чтобы обеспечить пользователю удобный интерфейс при работе с данными, хранящимися на диске, и обеспечить совместное использование файлов несколькими пользователями и процессами.

В широком смысле понятие "файловая система" включает:

- совокупность всех файлов на диске,
- наборы структур данных, используемых для управления файлами, такие, например, как каталоги файлов, дескрипторы файлов, таблицы распределения свободного и занятого пространства на диске,
- комплекс системных программных средств, реализующих управление файлами, в частности: создание, уничтожение, чтение, запись, именование, поиск и другие операции над файлами.

## **Имена файлов**

Файлы идентифицируются именами. Пользователи дают файлам символьные имена, при этом учитываются ограничения ОС как на используемые символы, так и на длину имени. До недавнего времени эти границы были весьма узкими. Так в популярной файловой системе FAT длина имен ограничивается известной схемой 8.3 (8 символов - собственно имя, 3 символа - расширение имени), а в ОС UNIX System V имя не может содержать более 14 символов. Однако пользователю гораздо удобнее работать с длинными именами, поскольку они позволяют дать файлу действительно мнемоническое название, по которому даже через достаточно большой промежуток времени можно будет вспомнить, что содержит этот файл. Поэтому современные файловые системы, как правило, поддерживают длинные символьные имена файлов. Например, Windows NT в своей новой файловой системе NTFS устанавливает, что имя файла может содержать до 255 символов, не считая завершающего нулевого символа.

При переходе к длинным именам возникает проблема совместимости с ранее созданными приложениями, использующими короткие имена. Чтобы приложения могли обращаться к файлам в соответствии с принятыми ранее соглашениями, файловая система должна уметь предоставлять эквивалентные короткие имена (псевдонимы) файлам, имеющим длинные имена. Таким образом, одной из важных задач становится проблема генерации соответствующих коротких имен.

Длинные имена поддерживаются не только новыми файловыми системами, но и новыми версиями хорошо известных файловых систем. Например, в ОС Windows 95 используется файловая система VFAT, представляющая собой существенно измененный вариант FAT. Среди многих других усовершенствований одним из

главных достоинств VFAT является поддержка длинных имен. Кроме проблемы генерации эквивалентных коротких имен, при реализации нового варианта FAT важной задачей была задача хранения длинных имен при условии, что принципиально метод хранения и структура данных на диске не должны были измениться.

Обычно разные файлы могут иметь одинаковые символьные имена. В этом случае файл однозначно идентифицируется так называемым составным именем, представляющим собой последовательность символьных имен каталогов. В некоторых системах одному и тому же файлу не может быть дано несколько разных имен, а в других такое ограничение отсутствует. В последнем случае операционная система присваивает файлу дополнительно уникальное имя, так, чтобы можно было установить взаимно-однозначное соответствие между файлом и его уникальным именем. Уникальное имя представляет собой числовой идентификатор и используется программами операционной системы. Примером такого уникального имени файла является номер индексного дескриптора в системе UNIX.

## **Типы файлов**

Файлы бывают разных типов: обычные файлы, специальные файлы, файлы-каталоги.

Обычные файлы в свою очередь подразделяются на текстовые и двоичные. Текстовые файлы состоят из строк символов, представленных в ASCII-коде. Это могут быть документы, исходные тексты программ и т.п. Текстовые файлы можно прочитать на экране и распечатать на принтере. Двоичные файлы не используют ASCII-коды, они часто имеют сложную внутреннюю структуру, например, объектный код программы или архивный файл. Все операционные системы должны уметь распознавать хотя бы один тип файлов - их собственные исполняемые файлы.

*Специальные файлы* - это файлы, ассоциированные с устройствами ввода-вывода, которые позволяют пользователю выполнять операции ввода-вывода, используя обычные команды записи в файл или чтения из файла. Эти команды обрабатываются вначале программами файловой системы, а затем на некотором этапе выполнения запроса преобразуются ОС в команды управления соответствующим устройством. Специальные файлы, так же как и устройства ввода-вывода, делятся на блок-ориентированные и байт-ориентированные.

*Каталог* - это, с одной стороны, группа файлов, объединенных пользователем исходя из некоторых соображений (например, файлы, содержащие программы игр, или файлы, составляющие один программный пакет), а с другой стороны - это файл, содержащий системную информацию о группе файлов, его составляющих. В каталоге содержится список файлов, входящих в него, и

устанавливается соответствие между файлами и их характеристиками (атрибутами).

В разных файловых системах могут использоваться в качестве атрибутов разные характеристики, например:

- информация о разрешенном доступе,
- пароль для доступа к файлу,
- владелец файла,
- создатель файла,
- признак "только для чтения",
- признак "скрытый файл",
- признак "системный файл",
- признак "архивный файл",
- признак "двоичный/символьный",
- признак "временный" (удалить после завершения процесса),
- признак блокировки,
- длина записи,
- указатель на ключевое поле в записи,
- длина ключа,
- времена создания, последнего доступа и последнего изменения,
- текущий размер файла,
- максимальный размер файла.

Каталоги могут непосредственно содержать значения характеристик файлов, как это сделано в файловой системе MS-DOS, или ссылаться на таблицы, содержащие эти характеристики, как это реализовано в ОС UNIX. Каталоги могут образовывать иерархическую структуру за счет того, что каталог более низкого уровня может входить в каталог более высокого уровня (рисунок 1).

Иерархия каталогов может быть деревом или сетью. Каталоги образуют дерево, если файлу разрешено входить только в один каталог, и сеть - если файл может входить сразу в несколько каталогов. В MS-DOS каталоги образуют древовидную структуру, а в UNIX'e - сетевую. Как и любой другой файл, каталог имеет символьное имя и однозначно идентифицируется составным именем, содержащим цепочку символьных имен всех каталогов, через которые проходит путь от корня до данного каталога.

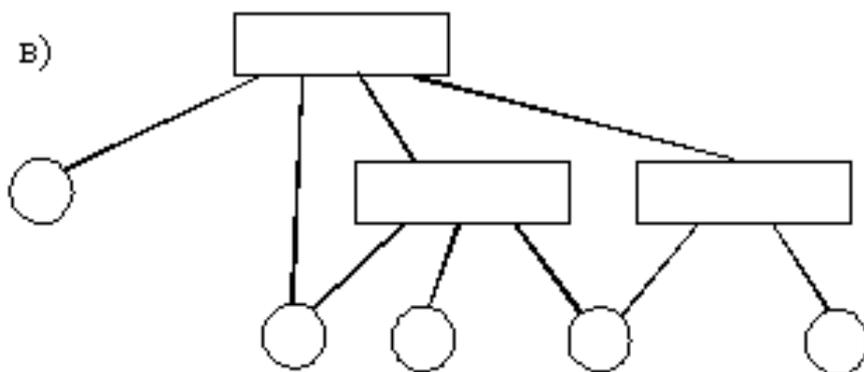
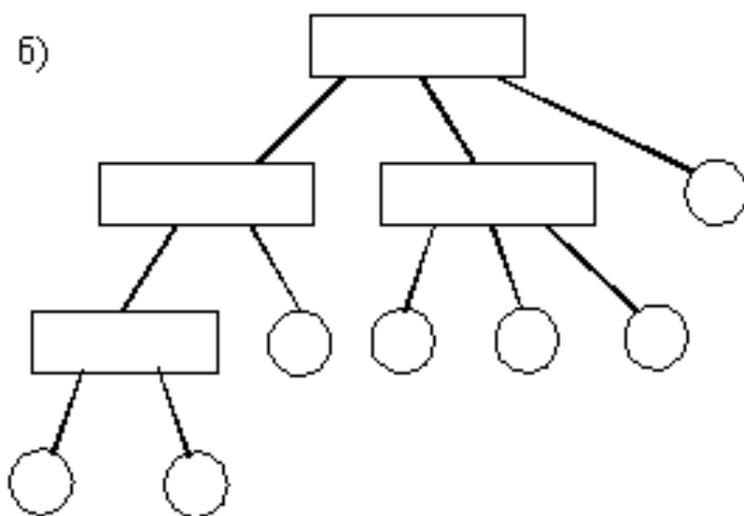
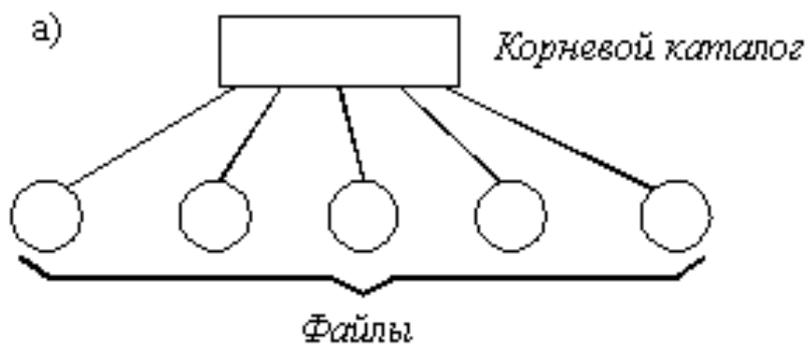


Рис. 1. Логическая организация файловой системы  
 а - одноуровневая;  
 б - иерархическая (дерево);  
 в - иерархическая (сеть)

## Глава 4. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Традиционно под словом “Технология” понимают процедуры и средства, позволяющие получить определенный результат в какой-то области человеческой деятельности. **Информационные технологии** в этом смысле не являются исключением — под ними понимается вся совокупность программного и аппаратного обеспечения, методов и средств организации автоматизированной обработки информации, представленной в цифровой форме.

Существует большое количество оснований для классификации информационных технологий. Их делят на индивидуальные и коллективные, на локальные и сетевые, технологии управления данными и процессами, защиты информации, разработки программного обеспечения и т.д.

Рассмотрим пять основных блоков:

1. Технологии обработки текста.
2. Технологии обработки графической информации.
3. Мультимедиа-технологии.
4. Электронные таблицы.
5. Базы данных.

Постоянный рост мощности вычислительной техники увеличивает спектр решений, доступных пользователю современного компьютера. Решение многих типовых на сегодняшний день задач еще 10 лет назад представляло серьезную, часто неразрешимую проблему. Многие технологии были недоступны за пределами крупных организаций. Постоянное и быстрое развитие техники и программного обеспечения приблизило их к пользователю. Следует отметить, что одним из магистральных направлений развития современных информационных технологий является разработка подходов и методов интеграции данных различного вида, смешивания функций различных систем, активное использование методов коммуникации для обмена данными.

## Технологии обработки текста.

### Текст: представление, хранение, ввод.

#### Представление текста

Энциклопедический словарь дает такое определение понятию “текст”:

*“Текст — это упорядоченный набор слов, предназначенный для того, чтобы выразить некий смысл. В лингвистике термин используется в широком значении, включая в себя и устную речь”.*

Представление информации в виде текста при обработке с помощью вычислительной техники близко к этому определению. Под “текстовым” понимают такое представление информации, в котором она представлена в виде записи слов (логических элементов) некоторого языка и доступна для чтения человеком.

Язык для такого представления характеризуется некоторым алфавитом — т.е. допустимым набором символов. Поскольку компьютер работает только с двоичным кодом, то для записи и обработки требуется взаимно-однозначно сопоставить символы и двоичные коды. *Правило сопоставления кодов и символов, входящих в алфавит, называется кодировкой.*

Первый широко распространенный стандарт кодирования — таблица (т.е. прямое сопоставление кодов символам) кодировки ASCII (*American Standard Code for Information Interchange*, американский стандартный код для обмена информацией) — был разработан в 1963 году. Стандарт предполагал использование не только в вычислительной технике, но и в телеграфии (он стал заменой 5-битного кода Бодо). В нем для кодирования каждого символа отводилось 7 бит. Восьмой бит использовался для служебных целей — контроля четности при передаче.

Эта часть таблицы кодировки содержит символы латинского алфавита, цифры, некоторые знаки препинания и набор управляющих символов (возврат каретки, перевод строки, конец файла, сигнал и т.п.).

Позже восьмой бит стали использовать для представления символов национальных алфавитов: первая часть таблицы — US-ASCII — использовалась по-прежнему, а содержание второй менялось в зависимости от исходного естественного языка. Каждый вариант этой второй половины (расширенной таблицы) исходной таблицы получил название “кодовой страницы” языка (code page).

Для русского языка таких расширений несколько (разрабатывались они в разное время). Наиболее известны: CP866 (DOS), KOI-8R (UNIX), CP1251 (Windows) и MacCyr.

Применение такого способа кодирования сильно затрудняет передачу текстовых сообщений между разными странами, объединение в сообщении текста на нескольких языках, а в случае с русским языком — и обмен файлами между разными ОС (для русского языка до сих пор активно применяется 4 разных кодовых таблицы). Для решения этих проблем в 1991 году некоммерческим объединением был предложен стандарт кодирования Юникод (Unicode).

Стандарт состоит из двух частей: универсального набора символов (Universal Character Set) и правил трансформации (Unicode Transformation Format). Универсальный набор символов предполагает описание всех возможных при записи текстов символов в виде общей таблицы кодов. Правила трансформации определяют способ записи этих кодов.

Первая версия стандарта предполагала использование двух байтов для кодирования каждого символа. В дальнейшем это кодовое пространство было расширено.

Сейчас чаще всего применяется способ трансформации UTF-8, обеспечивающий совместимость с предыдущими реализациями и стандартами. В частности, коды менее 128 записываются одним байтом, что автоматически превращает их в коды ASCII.

Применение этого стандарта кодирования позволяет объединять в одном тексте слова на различных языках (без ограничений на их количество), использовать устаревшие языки, дополнительные символы.

Наиболее переносимым и легко используемым с технической точки зрения способом хранения и передачи текста являются текстовые файлы. По сути, эти файлы представляют собой последовательности символов, разбитых на абзацы или строки.

## **Текстовые файлы**

Понятие “текстового файла” не предусматривает строго заданного формата или расширения. Тем не менее, помимо характерной для той или иной ОС таблицы кодировки, в текстовых файлах могут применяться три основных способа деления текста на строки (абзацы):

1. Windows (DOS) — символы “Возврат каретки” + “Перевод строки” (CR+LF).
2. Unix — символ “Перевод строки” (LF).
3. MacOS — символ “Возврат каретки” (CR).

Текстовые файлы применяются для самых различных целей и часто оказываются формой хранения данных, описанных более сложными формальными языками.

Эти файлы часто используются для записи конфигурации ПО, документирования, переноса данных, описания HTML- или XML-кода.

## Правила машинописного набора текста

Для облегчения анализа и последующего преобразования текста при его наборе в самых различных случаях рекомендуется соблюдать общие правила машинописного набора:

1. Все слова разделяются пробелом, и только одним пробелом.
2. Знаки препинания примыкают к предыдущему слову.
3. Скобки и кавычки всех видов примыкают к первому и последнему слову заключенного в них текста.
4. Текст разрывается только в конце абзаца.
5. Большие форматированные пробелы делаются вставкой символа табуляции, а не несколькими пробелами подряд.

Соблюдение этих правил позволяет легко использовать текст при подготовке более сложных документов, в которые он входит как важнейший элемент, или при организации автоматической обработки.

Возможность исправлять ошибки и набирать текст постепенно стала одной из существенных причин, по которым подготовка текстовой информации практически повсеместно была переведена с бумажной на компьютерную основу.

Текстовые редакторы с развитыми возможностями предоставляют пользователям возможность протоколировать и сохранять наборы действий, создавать макрокоманды, или **макросы**. Использование макросов позволяет ускорить выполнение частых простых задач обработки.

## Шрифты

Чаще всего текстовая информация используется при подготовке различных печатных материалов. Конечная цель подготовки такого материала — его печать или точное изображение печатной страницы на экране. В отличие от простой подготовки текстового файла, при подготовке печатного материала важно, как отображается текст. Практически все основные элементы и приемы оформления текстовых материалов заимствованы у давно существующих технологий — печатных, отсюда же пришла и основная часть терминологии.

Основным и наиболее важным средством определения внешнего вида текста является **шрифт** (*schreiben*, от нем. — “писать”). Шрифт — это графический

рисунок букв, цифр и символов, обладающий общими для всех символов стилистическими особенностями изображения.

Шрифт характеризуется рядом параметров:

1. Рисунок шрифта — графические особенности, определяющие общность шрифта и его отличие от всех других.
2. Кегль (кегель) — размер шрифта — предельная высота большой буквы и окружающих ее пробелов (термин введен для описания высоты площадки литеры при наборе с помощью типографской кассы). Чаще всего задается в типографских пунктах (1 пункт = 1/72 дюйма = 0,375 мм). По историческим причинам некоторые размеры имеют собственные названия: 8 пт — “петит”, 9 пт — “боргес”, 10 пт — “корпус”, 12 пт — “цицера”.
3. Начертание — шрифт с общим рисунком, но какими-либо отличительными признаками: более жирный, наклонный, разреженный. Иногда параметр плотности шрифта (светлый, полужирный, жирный) отделяют от начертания.
4. Часто как параметр задается подчеркивание или зачеркивание шрифта, или его написание как индекса — с уменьшением размера и подъемом/спуском относительно текущей строки.

Совокупность всех возможных размеров и вариантов написания шрифта называется **гарнитурой**. Гарнитур имеют имена, по которым часто называют и конкретный шрифт.

По общим чертам рисунка различают три основных вида шрифтов:

1. Рубленые шрифты. Для этих шрифтов характерно угловое соединение штрихов. Чаще всего у таких шрифтов нет засечек. Такими шрифтами часто набирают заголовки.
2. Антиквенные шрифты. Происходят от созданного Альбрехтом Дюрером шрифта “Антиква”. В этом шрифте соединения между штрихами сглажены, обязательны засечки. Это наиболее популярная “книжная” группа шрифтов для набора больших объемов текстов. Пример одного символа такого шрифта приведен на рисунке.
3. Акцидентные (оформительские) шрифты. Шрифты с самым разным рисунком, применяемые для оформительских целей, часто — стилизованные под рукописные буквы. Большие объемы текста такими шрифтами набирать не рекомендуется, он начинает утомлять взгляд.

Шрифт задается для набранного текста и не изменяет самих символов — он только определяет написание каждого символа, исходя из эталонного изображения. Библиотека таких изображений называется просто “шрифтом”.

Существует несколько основных способов описания шрифтов (точнее — гарнитуры шрифта):

1. Растровые шрифты. При таком способе каждая буква описывается отдельно, как некоторая матрица точек. Способ позволяет максимально ускорить обработку, но сильно затрудняет изменение размеров или начертаний. Для достижения качества каждый символ такой гарнитуры должен быть отредактирован вручную и должен храниться отдельно.

2. Векторные шрифты. При таком способе описания шрифт задается с помощью некоторых математических кривых, совокупность которых и составляет рисунок символов. Такой шрифт может изменять размеры без потери качества, но с помощью примитивов трудно добиться прорисовывания заполняемых элементов.

3. Контурные шрифты. Аналогично векторным, описываются с помощью некоторых математических кривых, но они определяют не символ, а его контур, который заполняется по определенным правилам. Именно этот тип шрифтов и является наиболее популярным.

Для использования векторных и контурных шрифтов необходимо выполнение операции, “создающей” шрифт (заданного рисунка, размера и начертания), годного для отображения. Такая операция называется “растеризацией”. В состав графических оболочек современных операционных систем входят программы — растеризаторы шрифтов определенного формата.

Наиболее популярные форматы шрифтов — это TrueTypeFonts (TTF, поддерживается ОС Windows и MacOS) и PostScript (разработан фирмой Adobe, для использования необходима программа Adobe TypeManager). Сейчас на смену этим форматам приходит совместно разработанный этими компаниями формат OpenType.

Растеризация шрифта — достаточно ресурсоемкая операция, поэтому контурные шрифты получили распространение только с началом массового применения достаточно мощных компьютеров.

## Структурирование текста

Помимо внешнего вида букв, важное значение имеет пространственное расположение текста. Единицей пространственного размещения служит **абзац**. Как и в литературе, в компьютерном тексте абзацем называется выделенный по смыслу участок.

Для оформления абзаца используют несколько параметров:

1. **Выравнивание** — *правило расположения букв в строке абзаца*. Видов выравнивания четыре: по левому краю, центральное, по правому краю и по ширине полосы набора.
2. **Отступы** от краев полосы набора.
3. **Абзацный отступ** (красная строка) — *положение первой строки абзаца*.
4. **Интервалы**. Различают **межстрочное расстояние** — задается множителем размера шрифта (одинарный, полуторный, двойной интервал) — и **промежутки** до и после абзаца.
5. **Буквица** — *крупная выступающая первая буква абзаца*. Часто задается не просто более крупным размером буквы, но и буквой другого рисунка.

Абзацы размещаются в рамках **полосы** — *выделенного участка страницы, как правило, прямоугольной формы, в котором размещаются текст и иллюстрации*. На листе может быть либо одно такое место (**одна колонка**), либо несколько — тогда говорят о многоколоночном тексте.

Как правило, текстовые процессоры не дают появляться **висячим строкам** — *отдельным строкам абзацев в начале или конце полосы*.

Важным элементом оформления текста на странице являются **поля** — пробелы вдоль края страницы и интервалы между колонками. Для удобочитаемости, в силу особенностей восприятия, такие пробелы должны быть обязательно.

Как правило, в достаточно большом (больше нескольких страниц) тексте выделяется несколько смысловых блоков (разделов) и видов содержательного текста — обычный текст, примечания, ссылки и т.п.

Для оформления таких типовых элементов создаются **стили** — *определенные наборы параметров оформления шрифта и абзацев*. Применение стилей позволяет ускорить набор, автоматизировать оформление (например, автоматически создавать оглавления) и изменять внешний вид различных элементов, не разыскивая их по всему тексту. Практически все современные текстовые процессоры опираются на стили, даже если пользователь не использует их. Единство оформления — одно из условий удобочитаемости и красоты печатного издания.

Для решения некоторых типовых задач оформления текстов существующие текстовые процессоры предусматривают два мощных средства автоматизации.

1. **Списки.** При оформлении текста это *набор визуально выделенных элементов перечисления*. Элементы выделяют с помощью символа-маркера (**маркированные списки**) либо номером — в упорядоченных списках. При оформлении списка чаще всего также предусматривают форматирование абзацев — так, чтобы они не выступали за маркер. Автоматизация оформления позволяет автоматически маркировать и выделять новые элементы списков.

2. **Таблицы.** Современные текстовые процессоры предусматривают средства для создания двухмерной структуры размещения информации. Применение таких средств позволяет редактировать структуру и содержание таблицы, добавлять строки и столбцы, изменять их линейные размеры, выделять их с помощью сетки или фона. Фактически каждая ячейка таблицы становится листом в миниатюре.

Стоит отметить, что файлы текстового процессора содержат массу дополнительных (по отношению к тексту) данных об оформлении и текстовыми очень часто не являются.

## **Автоматизированная обработка текста**

Текстовое представление информации — одно из наиболее удобных для организации автоматической обработки. Связано это с тем, что в этой форме информация представляется в виде близком к исходному языку, что позволяет выполнять преобразования, связанные со смыслом текста.

Существует несколько наиболее распространенных автоматизированных операций, связанных с текстовым представлением.

## **Поиск**

Задача поиска необходимой информации чаще всего формулируется как поиск фрагментов, содержащих некоторые понятия, в достаточно большом массиве. Большое значение этот вид автоматической обработки получил с ростом популярности межсетевой среды Интернет. Существует несколько подходов к организации такого поиска.

Первый подход опирается на поиск фрагмента текста, соответствующего некоторому образцу. Наиболее популярная форма задания этого образца — так называемые **регулярные выражения**. По сути, это *описание фрагмента текста, удовлетворяющего некоторым условиям*, по тем частям, которые в нем содержатся, и их порядку. Таким способом в большом текстовом массиве можно находить упоминания тех или иных слов, адреса, номера телефонов и т.п. шаблонные элементы.

Достоинство этого подхода — возможность применять его к массиву текста без предварительной обработки. Например, сразу при посимвольном получении текста.

Второй подход предусматривает предварительное создание *специального вида базы для ускорения поиска* — **индекса**. Такой способ применяется для ускорения поиска, если некоторые типовые поисковые запросы повторяются часто и нет возможности формировать/хранить весь массив текста. Например, при организации поисковой машины в среде Интернет.

## **Расшифровка или уточнение значений слова**

Для решения такой задачи в самых разных видах применяют **словари** — *базы информационных фрагментов, связанных с некоторыми ключевыми словами или словосочетаниями*. Примером таких баз могут быть словари различных языков: англо-русский, русско-английский, толковый и другие виды словарей. Одно из самых распространенных применений словарей — проверка правописания слов при наборе.

Особым видом словарей являются **тезаурусы** — *словари, в которых слова связываются на основе каких-либо лексических отношений*. Например: слова являются синонимами (смысловыми аналогами), антонимами (противоположны по смыслу) и т.п.

Этот вид словарей важен не только потому, что может помочь при подготовке текстов, но и потому, что это отразит смысл слов — для систем, моделирующих отдельные аспекты мышления человека.

## **Системы автоматизированной обработки текста**

Используя закономерности естественного языка и описанные выше средства выполнения некоторых операций и выявления зависимостей, с помощью ЭВМ автоматизируют (хотя и не полностью) некоторые операции по смысловому преобразованию текста. Современные системы обработки позволяют создавать краткие обзоры текстов (рефераты) или готовить перевод с одного естественного языка на другой. Приходится отметить, что точного решения эти задачи не имеют, поскольку зачастую трудно подобрать адекватное слово или выражение, учитывая не только формальный перевод, но и грамматические особенности, и культурные. Тем не менее с применением специализированных по областям знания словарей современные системы автоматизированного перевода создают подстрочник, который может дать представление о смысле текста и в дальнейшем помочь переводчику в переводе документа.

## **История компьютерной графики**

Первоначально практически все ЭВМ работали с числовыми и символьными данными. В качестве устройств ввода и вывода использовались специальные пульты, перфокарты и телетайпы — печатающие машины.

В декабре 1951 года инженер Массачусеттского технологического института Джей У. Форрестер продемонстрировал новый компьютер “Вихрь”, принципиальным отличием которого было устройство вывода, формировавшее изображение на экране электронно-лучевой трубки. Изображение формировалось из отдельных светящихся точек. Позднее для оперативного управления компьютером во время управления комплексами ПВО было разработано первое интерактивное устройство ввода — световой пистолет.

Таким образом, было положено начало одному из самых применяемых направлений информационных технологий — компьютерной графике.

Основной принцип формирования изображения, примененный в этом компьютере, формирование изображения на экране из *отдельных точек, расположенных в узлах прямоугольной сетки (растра)*, получил название **растровой графики**.

Следующим шагом в развитии этого направления стала разработка в 1961–1962 году Айвеном Сазерлендом первой интерактивной программы для выполнения чертежей — Sketchpad (Блокнот). Программа впервые реализовала принцип интерактивного рисования отдельных графических примитивов (отрезков и дуг) из отдельных точек и последующих операций с ними. Интерактивность достигалась применением светового пера для указания необходимых координат. Примерно в то же время была разработана первая система автоматизированного проектирования (DAS-1), но она требовала ввода координат примитивов с клавиатуры.

Наиболее существенной трудностью при работе с графикой была высокая загрузка центрального процессора и памяти, поскольку изображение полностью формировалось с помощью ЦП. Для преодоления этого затруднения были разработаны системы с памятью регенерации (позднее — видеопамятью), снимавшие с центрального процессора нагрузку. В таких системах каждая точка изображения описывалась некоторым числом. Практически все мониторы были монохромными, позволяли работать только с одним цветом (не обязательно белым).

Поскольку наличие большой памяти (для каждой точки — не менее бита) делало такие системы крайне дорогими, вместо них долгое время применялись системы с запоминающей электронно-лучевой трубкой, удерживавшей изображение около часа. Применение таких систем не давало возможности работать с изображением интерактивно, но сильно удешевляло производство.

Появление в начале 80-х годов высокоскоростных и дешевых запоминающих устройств на основе микросхем позволило активно развивать это направление.

## Глава 5. Технологии обработки графической информации

### Основные характеристики растровой графики

Основными параметрами изображения в растровой форме является **разрешение**, *возможное количество градаций*. Различают разрешение линейное — количество столбцов по горизонтали и линий по вертикали, и цветное/оттеночное — количество оттенков или цветов у каждой точки. Линейное разрешение описывают как количество точек, а цветное — в виде количества битов, отводимых на описание каждой отдельной точки (эту величину еще называют битовой глубиной цвета).

Чем выше количество точек на единицу площади, чем выше количество цветов каждой точки, тем выше возможное качество изображения, но тем больше объем памяти, необходимый для хранения и обработки изображения. Например, при использовании 24 бит для представления цвета каждой точки может быть использовано 16 777 216 (224) оттенков, и изображение размером 1280 г 1024 точки (разрешение современного монитора) будет занимать как минимум 3840 Кб памяти.

Растровая графика — универсальное средство для формирования и обработки любых плоских изображений. С помощью цветов и оттенков отдельных точек на плоском изображении могут быть показаны и пространственные (объемные) сцены.

Растровая графика — основное средство представления и обработки фотографических изображений, стилизованных художественных рисунков, всевозможных диаграмм, текста. С помощью именно этого способа представления информации строятся современные человеко-машинные интерфейсы.

Несмотря на универсальность, этот способ представления информации имеет целый ряд недостатков. К ним относятся: зависимость (причем квадратичная) качества изображения от его объема, трудность выделения и манипуляции отдельными осмысленными элементами, существенное падение качества изображения в результате геометрических преобразований.

Для преодоления этих недостатков программы работы с растровой графикой предусматривают средства создания составных изображений с помощью механизма **слоев** (*layers*) — накладывающихся друг на друга плоскостей, в каждой из которых используется только часть точек, механизма **фильтров** — преобразующих цвета пикселей с учетом некоторых параметров (выполняя, например, размытие или внесение геометрических искажений), управления цветовыми каналами и способом взаимодействия отдельных слоев.

Программы работы с растровой графикой имеют в своем составе большой набор способов изменения цвета пикселей, для этого используется метафора “инструмента” — модели кисти или карандаша с изменяемыми параметрами. Пользователь может создавать библиотеки таких инструментов.

## **Цветовые модели**

Один из самых важных вопросов при организации обработки графических данных — это представление и кодирование цвета.

В простейшем случае, когда на устройстве назначения всего два цвета, используется всего один бит, состояние которого и задает цвет. Если же цветов становится больше, то такой подход уже не может решить задачу.

Существует несколько способов кодирования цвета, применяемых при обработке как растровой, так и векторной графики.

Для описания градации одного цвета применяется обычное кодирование, в котором номер обозначает **градацию**. Чем больше значение, тем сильнее проявляется цвет. Для устройств-мониторов (в которых точка самостоятельно излучает свет) 0 обычно соответствует отсутствию цвета, а максимальное значение — максимальной светимости точки. Таким образом, появляется возможность задавать **оттенок** на монохромном мониторе.

В случае, когда используется печатающее устройство, на котором чернильная точка либо есть, либо нет, оттенок задается некоторой матрицей (например — 4 x 4 точки), количество чернильных точек в матрице точек будет образовывать оттенок.

В более сложных случаях, когда речь идет о кодировании сложного цвета с большим количеством оттенков, рассматривают разложение цвета на несколько отдельных компонентов, которые, смешиваясь (т.е. действуя в одной точке), образуют заданный цвет.

Компоненты цвета и способ образования из них видимого оттенка образуют **цветовую модель**.

Цветовые модели разрабатывались задолго до появления вычислительной техники, в психологии восприятия. Существует большое количество цветовых моделей, которые создавались и вводились разными авторами для описания и исследования зрения человека. С появлением проекционной и печатающей аппаратуры, с учетом технических требований были разработаны новые модели, учитывающие в первую очередь физические и технические аспекты формирования конкретного цвета.

Для каждого конкретного изображения все, что передается одним из компонентов цвета, также называется “каналом”.

Наиболее популярны сейчас следующие модели:

## **Модель восприятия HLS**

Модель подразумевает образование цвета из трех основных компонентов:

**Hue** — оттенок цвета;

**Lights** — яркость;

**Saturation** — насыщенность.

При ее использовании считается, что все оттенки заданы на едином цветовом круге. Поэтому первый параметр задает градус поворота от эталонного оттенка (0 — белый). Остальные параметры задаются в процентах как положение между максимальными и минимальными доступными значениями.

Модель также известна под названиями HSL, HIS и другими.

Эта модель наиболее приближена к человеческому восприятию и описанию цвета. Она применяется в основном для описания цвета при анализе его восприятия человеком.

## **Аддитивная модель цвета RGB**

В этой модели цвет образуется смешиванием трех компонентов:

**Red** — красный;

**Green** — зеленый;

**Blue** — голубой.

В данном случае цвет образуется из света нескольких источников (в том случае, когда источники излучают свет и он может быть “просуммирован”); модель является **аддитивной**.

Самыми популярными “потребителями” модели RGB являются мониторы, в которых цвет каждого пикселя раstra складывается из трех компонентов, проекторы и сканеры, которые чаще всего регистрируют отраженный свет.

Именно цветовая модель RGB используется и при описании возможностей различных графических устройств. Цветовое пространство в этом случае характеризуют количеством битов, отводимых на сохранение цвета. Чаще всего

используются режимы HighColor (16 бит, в соотношении 5:6:5 или 5:5:5) и TrueColor (24 бита, в соотношении 8:8:8).

Профессиональные программы обработки графической информации позволяют работать с расширенным представлением, когда на одну компоненту отводится не 8, а 16 бит.

Каждый компонент задается силой светимости, 0 соответствует отсутствию света. Таким образом, цвет 0-0-0 — это черный, цвет из равных долей каждого компонента — один из оттенков серого, а цвет с максимальными значениями компонентов — белый.

## **Векторная графика**

Существенным недостатком растровой графики является трудность манипуляции отдельными объектами и выполнение геометрических преобразований. Одно из следствий этого — трудности в организации качественного вывода на различных устройствах и затруднения при изображении новых, не сфотографированных объектов.

Кроме того, очень быстро растет объем изображения (и необходимой для него памяти) при увеличении линейных размеров.

Для преодоления этих трудностей применяется подход, подразумевающий хранение и обработку изображения *не в виде растра, а в виде некоторых отдельных элементов (графических примитивов)*. Элементами обычно являются математические объекты с заданными конкретными параметрами. Параметры позволяют выполнить визуализацию элементов на устройстве вывода (**растеризацию**), исходя из его характеристики и заданного “окна” просмотра.

Поскольку пространственное положение примитивов и способ отображения задаются с помощью координат, способ хранения и обработки получил название **векторной графики**.

Одним из наиболее существенных достоинств векторной формы представления изображения является ее компактность и малая зависимость объема от размеров изображения.

К минусам этой формы представления относится отсутствие общих стандартов (практически у каждого редактора есть свои собственные форматы и особенности) и высокие требования к системным ресурсам, особенно — ресурсам вычислительным.

В программах подготовки векторных изображений работа строится вокруг **объектов** (примитивов), обладающих некоторыми свойствами.

С объектами редактор векторной графики может выполнять большое количество разнообразных операций. К таким операциям относятся: **повороты, масштабирование, геометрические искажения** всевозможных видов, тиражирование готовых объектов. Специфика формы представления такова, что операции выполняются без искажений.

Современные редакторы векторной графики могут импортировать и использовать как готовые объекты изображения растровой графики.

Редакторы векторной графики позволяют **группировать** объекты и создавать таким образом сложные объекты для выполнения операций над ними как над единым целым.

Объекты могут быть упорядочены друг относительно друга, распределены на плоскости как “на поверхности”, так и “по вертикали”.

Как и программы растровой графики, программы векторной графики поддерживают работу со слоями.

Векторная графика применяется в программах автоматизированного проектирования, подготовки графических печатных материалов (плакатов, например), для подготовки анимационных роликов к публикации в сети Интернет, презентаций.

## Глава 6. РАСШИРЕННЫЕ ВОЗМОЖНОСТИ WORD

### Методы обработки информации

Существует множество методов обработки информации, но в большинстве случаев они сводятся к обработке текстовых и числовых данных.

Текстовая информация может возникать из различных источников и иметь различную степень сложности по форме представления. В зависимости от формы представления для обработки текстовых сообщений используют разнообразные информационные технологии. Чаще всего в качестве инструментального средства обработки текстовой электронной информации применяют текстовые редакторы или процессоры. Они представляют программный продукт, обеспечивающий пользователя специальными средствами, предназначенными для создания, обработки и хранения текстовой информации. Текстовые редакторы и процессоры используются для составления, редактирования и обработки различных видов информации. Отличие текстовых редакторов от процессоров заключается в том, что редакторы, как правило, предназначены для работы только с текстами, а процессоры позволяют использовать и другие виды информации.

Редакторы, предназначенные для подготовки текстов условно можно разделить на обычные (подготовка писем и других простых документов) и сложные (оформление документов с разными шрифтами, включающие графики, рисунки и др.). Редакторы, используемые для автоматизированной работы с текстом, можно разделить на несколько типов: простейшие, интегрированные, гипертекстовые редакторы, распознаватели текстов, редакторы научных текстов, издательские системы.

В простейших редакторах-формateraх (например, “Блокнот”) для внутреннего представления текста дополнительные коды не используются, тексты же обычно формируются на основе знаков кодовой таблицы ASCII.

Текстовые процессоры представляют систему подготовки текстов (Word Processor).

Большинство текстовых процессоров поддерживает концепцию составного документа – контейнера, включающего различные объекты. Она позволяет вставлять в текст документа рисунки, таблицы, графические изображения, подготовленные в других программных средах. Используемая при этом *технология связи и внедрения объектов* называется *OLE* (Object Linking and Embedding – связь и внедрение объектов).

Для автоматизации выполнения часто повторяемых действий в текстовых процессорах используют макрокоманды. Самый простой макрос – записанная последовательность нажатия клавиш, перемещений и щелчков мышью. Она может воспроизводиться, как магнитофонная запись. Её можно обработать и изменить, добавив стандартные макрокоманды.

Перенос текстов из одного текстового редактора в другой осуществляется программой-*конвертером*. Она создаёт выходной файл в соответствующем формате. Обычно программы текстовой обработки имеют встроенные модули конвертирования популярных файловых форматов.

Разновидностью текстовых процессоров являются *настольные издательские системы*. В них можно готовить материалы по правилам полиграфии. Программы настольных издательских систем (например, Publishing, PageMaker) являются инструментом верстальщика, дизайнера, технического редактора. С их помощью можно легко менять форматы и нумерацию страниц, размер отступов, комбинировать различными шрифтами и т.п. В большей степени они предназначены для издания полиграфической продукции.

## Назначение тестового процессора Word

MS Word – наиболее популярный текстовый процессор из известных на сегодняшний день. Это очень удобный и достаточно простой в использовании инструмент для создания профессионально оформленных документов.

При запуске текстового процессора Word лента (область, расположенная в верхней части окна приложения Word) выносит на передний план наиболее распространенные команды, так что теперь нет необходимости искать часто используемые команды в разных частях приложения.

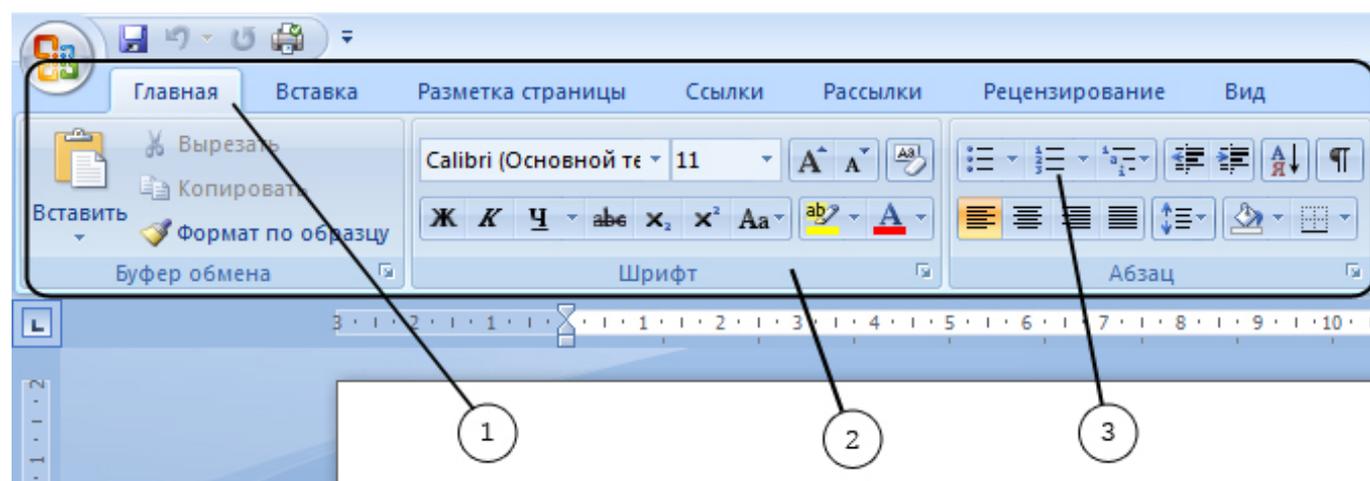


Рисунок 1. Лента

Лента состоит из трех основных элементов:

1. **Вкладки.** В верхней части ленты расположены семь основных вкладок, каждая из которых представляет определенный набор функций.
2. **Группы.** На каждой вкладке имеется несколько групп, которые отображают связанные общим назначением команды.
3. **Команды.** Команда – это кнопка, поле для ввода информации или меню. В правом нижнем углу некоторых групп имеется небольшая диагональная стрелка. Эта стрелка называется кнопкой вызова диалогового окна. При нажатии на нее будут отображены дополнительные возможности, связанные с этой группой. Данные возможности часто отображаются в виде диалогового окна, либо в виде привычной области задач. В Word некоторые вкладки появляются только по мере необходимости.

**Панель быстрого доступа** – это небольшая область в левом верхнем углу ленты. Она содержит команды, которые многократно используются при каждом сеансе работы: Сохранить, Отменить, Повторить и другие. На эту панель также можно поместить команды, которые будут доступны независимо от того, какая вкладка активна.

Существует два варианта размещения панели быстрого доступа.

Левый верхний угол рядом с кнопкой Office (местоположение по умолчанию) или под лентой

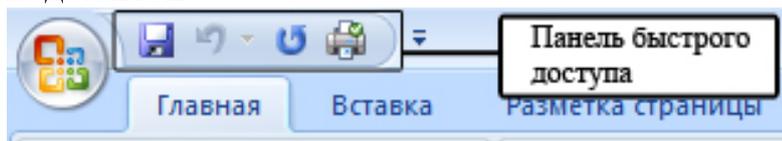


Рисунок 2. Панель быстрого доступа

Изменить расположение панели быстрого доступа возможно следующим образом:

1. Щелкните кнопку **Настройка панели быстрого доступа**
2. Выберите в списке **Разместить под лентой**.

Команды **Открыть**, **Сохранить**, **Сохранить как** и т.д. находятся в меню,

которое открывается щелчком по кнопке Office.

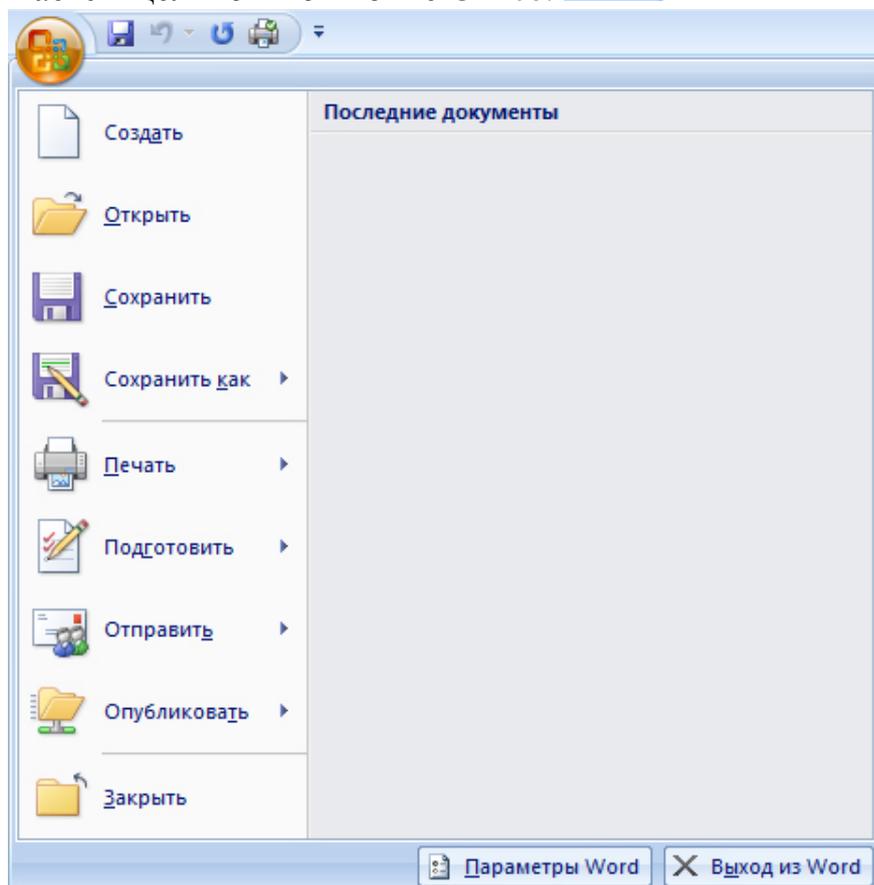


Рисунок 3. Меню Office

## Правила выделения фрагментов текста в Word

**Произвольный фрагмент** выделяется протаскиванием мыши от начала фрагмента до его конца (или наоборот).

**Слово** выделяется двойным щелчком левой кнопкой мыши внутри выделяемого слова.

**Строка** выделяется одинарным щелчком левой кнопкой мыши в *зоне выделения* (пустая зона слева от строки, указатель мыши принимает в ней вид белой стрелки, направленной вправо).

**Несколько строк** – протаскиванием мыши в зоне выделения.

**Весь текст** – одинарный щелчок левой кнопкой мыши в зоне выделения при удерживаемой клавише <Ctrl> или выбрать вкладка **Главная** – кнопка **Выделить** – команда **Выделить все**.

**Снять выделение** – одинарный щелчок левой кнопкой мыши в любом месте рабочего поля.

Для **копирования** фрагмента текста из документа в буфер обмена нужно выделить требуемый фрагмент и выбрать вкладка **Главная** – **Копировать** (или клавиши на клавиатуре Ctrl+C). Полученная копия может многократно использоваться для вставки в другое место данного документа или в другой документ с помощью команды вкладка **Главная** – **Вставить** в текущую позицию курсора (на клавиатуре клавиши Ctrl+V).

Для **перемещения** фрагмента текста из документа в буфер обмена необходимо выделить требуемый фрагмент и выполнить команду вкладка **Главная** – **Вырезать** (или на клавиатуре клавиши Ctrl+X).

**Вставить вырезанный фрагмент** в текст - команда вкладка **Главная** – **Вставить**. Текст из буфера вставляется в то место текстового документа, где находится текстовый курсор.

Для **удаления** фрагмента текста его необходимо выделить и воспользоваться клавишей (**Del**) **Delete** на клавиатуре.

При наличии выделенного фрагмента нажатие на любую клавишу ведет к удалению этого фрагмента.

Текстовый процессор Word позволяет проверять правописание вкладка **Рецензирование** – **Правописание** и расставлять переносы в словах вкладка **Разметка страницы** – **Расстановка переносов**.

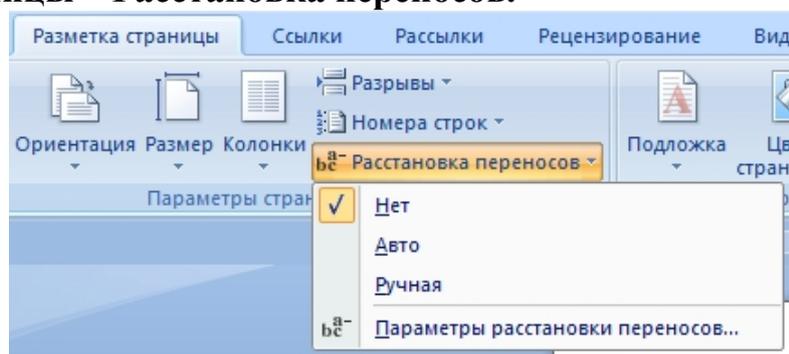


Рисунок 4. Расстановка переносов

## Технология форматирования символов

Основным объектом текста является символ. Символ характеризуется следующими параметрами: **гарнитура (вид шрифта), кегль (размер), цвет.**

Под форматированием понимают изменение параметров символа.

Форматирование шрифта выполняется с помощью вкладки **Главная** – группа **Шрифт** или в диалоговом окне **Шрифт**.

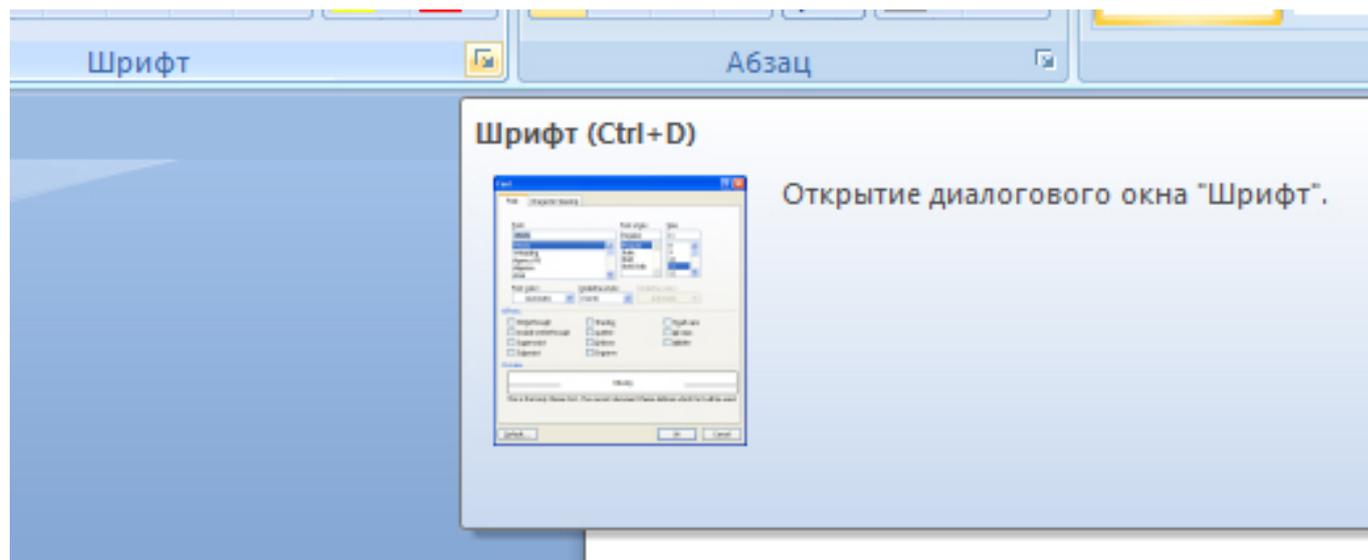


Рисунок 5. Диалоговое окно Шрифт

Если изменить формат шрифта во время ввода текста, то эта команда будет действовать до её отмены. Если же нужно переформатировать какой-либо участок текста после ввода, его необходимо выделить и затем выбрать нужные параметры. Вкладка **Интервал** окна **Шрифт** позволяет изменить интервал между символами.

## Смена регистра

Набирая текст, иногда требуется изменить регистр некоторых слов в тексте или же увеличить/уменьшить их размер.

### РЕГИСТР

Различают верхний регистр и нижний регистр. Верхний регистр – это заглавные буквы (большие), а нижний – строчные буквы (маленькие).

Для того чтобы кнопка смены регистра была доступна независимо от активной вкладки можно добавить ее на панель быстрого доступа. Для добавления кнопки на панель быстрого доступа необходимо выполнить следующее:

1. Вызвать контекстное меню для ленты;
2. Выбрать команду **Настройка панели быстрого доступа**;
3. Из списка **Выбрать команды из** выбрать **Все команды**, найти команду **Регистр**;

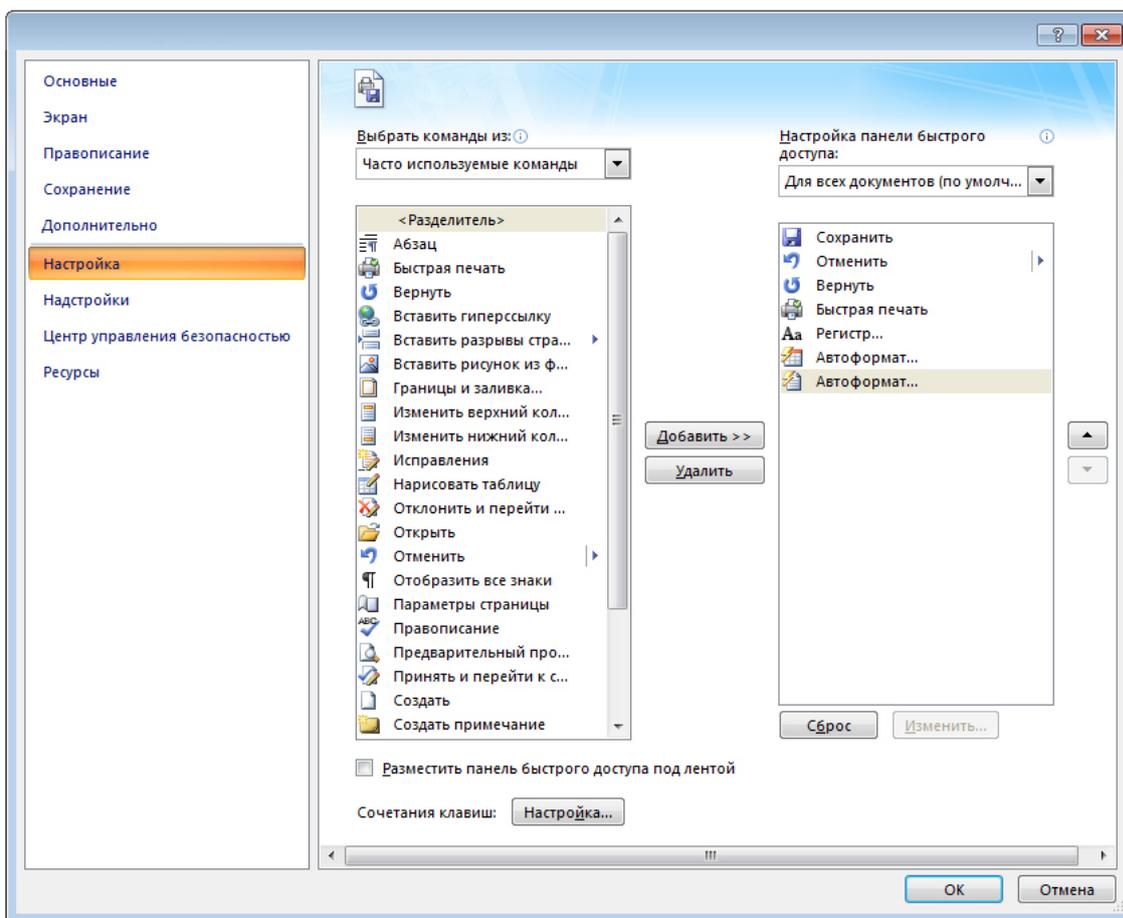


Рисунок 6. Окно настройки Панели быстрого доступа

4. Щелчком по кнопке **Добавить** добавить команду на Панель быстрого доступа;
5. Щелкнуть **ОК**.

Как набрать на одной строке несколько слов с большим расстоянием между ними? Сделать это можно, используя горизонтальную линейку Word. Включить линейку в Word можно выполнив вкладка **Вид** – группа **Показать или скрыть** – **Линейка**.



Должность

Заведующий кафедрой

ФИО

Иванов П.П.

Рисунок 7. Линейка Word

На линейке устанавливаются позиции табуляции, которые позволяют выравнивать текст по заданным параметрам. Существуют пять основных типов табуляции:

- выравнивание текста по левому краю ,
- выравнивание текста по правому краю ,
- выравнивание текста по центру ,
- выравнивание текста по разделителю (для чисел) ,
- табуляция с чертой .

Чтобы установить нужное выравнивание:

- щелчком по квадратику в левой части линейки , выберите необходимый табулятор,
- щелчком по линейке, установите значок табуляции в нужной позиции,
- нажатие клавиши **Tab** перемещает курсор по позициям табуляции.

Если понадобится вставить табулятор в ячейку таблицы, нужно нажимать **Ctrl+Tab**, а не **Tab**, так как клавиша **Tab** в таблицах служит для перехода между ячейками.

- При определении позиции табуляции, нажмите клавишу **Alt**, чтобы установить расстояния с точностью 0,25 мм (без клавиши **Alt** точность составляет 0,5 см). Тот же прием можно использовать для изменения границ таблицы, отступов и полей страницы.
- Двойной щелчок по верхней границе горизонтальной линейки вызывает окно настройки **Параметры страницы**. Если дважды щелкнуть по нижней границе, появится окно **Табуляция**.
- Если линейка не видна на экране, включить/выключить линейку можно с помощью кнопки, расположенной над вертикальной линейкой прокрутки.

## Технология форматирования абзацев

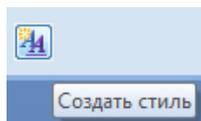
Новый абзац создается нажатием клавиши **Enter** и включает в себя весь текст от символа конца предыдущего абзаца до символа конца текущего абзаца . Под форматированием абзаца понимают изменение его параметров. Параметрами абзаца являются: **способ выравнивания, отступы, позиции табуляции, междустрочные интервалы и интервал между абзацами**. С помощью отступов (слева и справа) можно задать ширину абзаца, а отступ (выступ) первой строки определяет начало первой строки абзаца и не влияет на другие строки. Соответствующие установки выполняются посредством команд вкладки **Главная** – группа **Абзац** или аналогичных команд контекстного меню, кроме этого можно использовать маркеры на горизонтальной линейке.

## Стили абзацев

При создании больших документов или при объединении нескольких документов в один рекомендуется использовать единые стили: для заголовков, для обычного текста, для нумерованных списков, для выделения жирным шрифтом и так далее. Чтобы достичь такого единства при работе над документом возможно использовать встроенные стили или создавать пользовательские стили форматирования.

Для создания пользовательского стиля

- Создать пустой документ (основанный на шаблоне);
- Выполнить вкладка **Главная** – группа **Стили**;
- Открыть окно стилей;



Щелкнуть по кнопке **Создать стиль**;

В открывшемся диалоговом окне сделать необходимые установки:

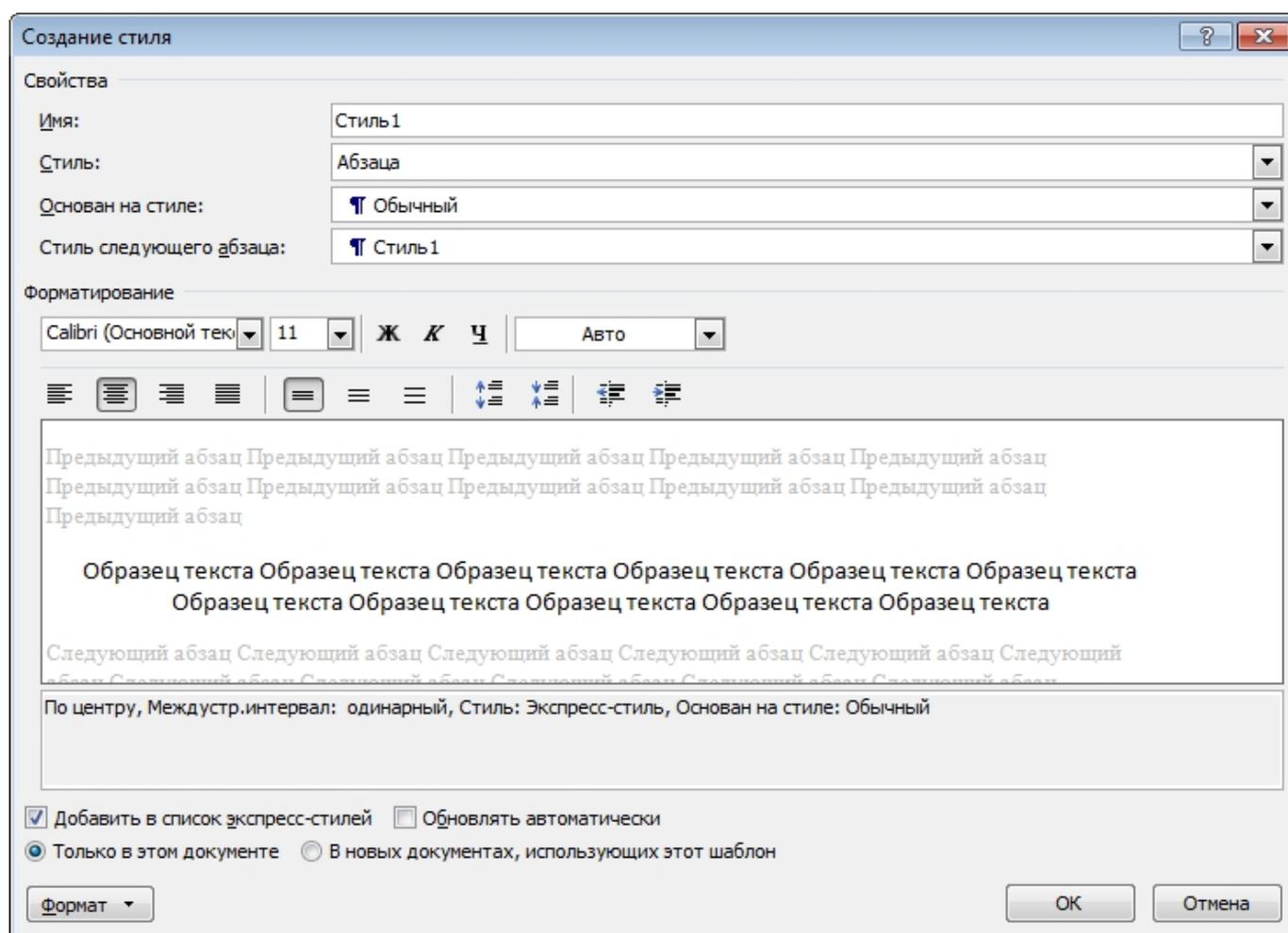


Рисунок 8. Создание стиля

1. В поле **Имя** ввести имя создаваемого стиля.
2. Поле **Стиль** оставьте без изменения. Там должна быть строка “Абзаца”.
3. В поле **Основан на стиле** выберите строку “(нет)”.
4. В поле **Стиль следующего абзаца** выберите строку “**Обычный**”. Это стиль при нажатии клавиши **Enter** на клавиатуре.
5. Для того чтобы стиль имел нужное вам форматирование, рекомендуется либо нажать соответствующие кнопки в этом окне, либо использовать кнопку **Формат** для выбора команд с расширенными настройками форматирования.
6. Ок.

Созданный стиль появится в окне **Стили**.

Если подвести указатель мыши к названия стиля в окне **Стили** и задержать на несколько мгновений, то появится подсказка с параметрами данного стиля.

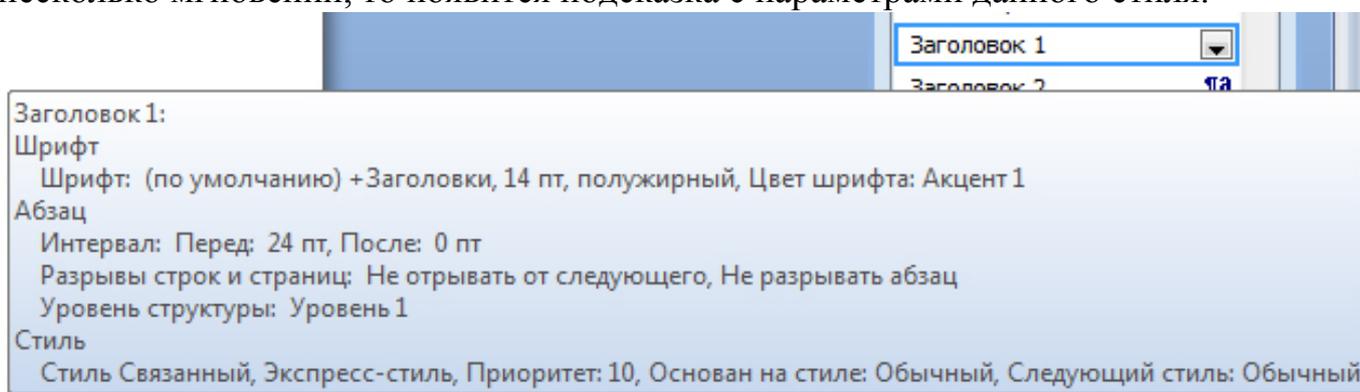


Рисунок 9. Подсказка в окне стилей

Существует несколько способов применения стилей. Стиль для текста:

1. Выбрать необходимый стиль из списка и ввести текст,
2. Ввести текст, выделить его и применить стиль форматирования. Если необходимо применить стиль для отдельно взятого слова или нескольких слов, то предварительно выделить их.

Стиль для абзаца:

Для применения стиля к одному абзацу, достаточно сделать его текущим (установить курсор внутри абзаца), для нескольких абзацев – выделить их.

### **Управление стилями. Импорт/Экспорт стилей.**

При работе над документами нескольких авторов возникает необходимость одинакового форматирования. Word позволяет копировать стили из одного документа в другой. Для импорта стилей необходимо:

1. Открыть окно **Управление стилями** в документе, из которого стили импортируются, щелкнув по кнопке – Управление стилями;
2. В окне Управление стилями Щелкнуть по кнопке Импорт/Экспорт. Откроется диалоговое окно **Организатор**;

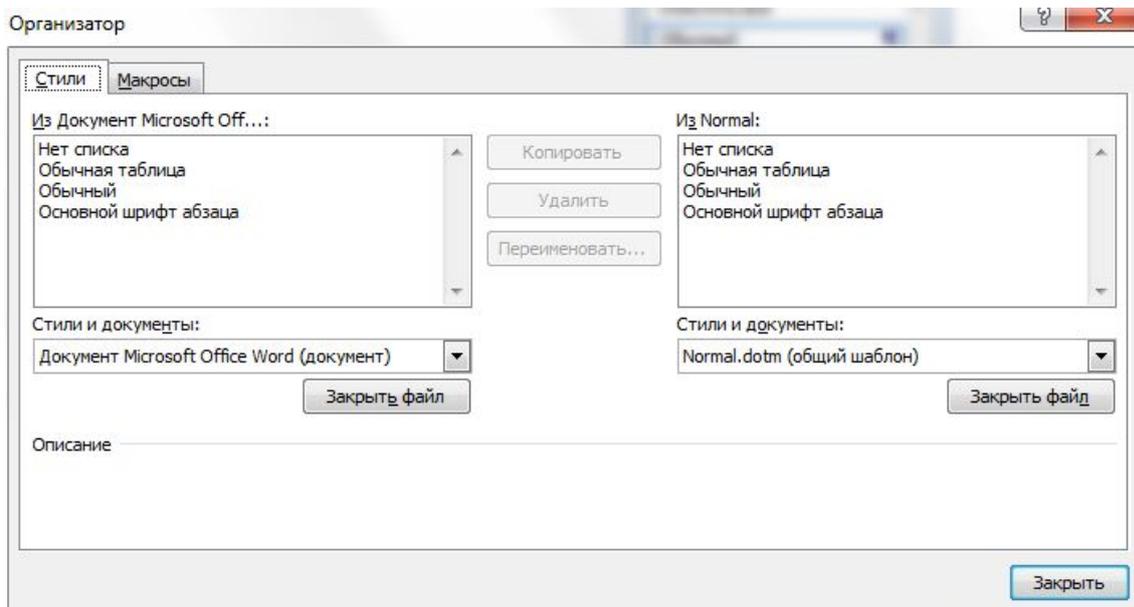


Рисунок 10. Окно Организатор

3. В списке **Из...** выбрать необходимые стили и щелкнув по кнопке копировать, добавить их **В Normal...**
4. Открыть в документе, в который стили экспортируются, окно **Организатор**;
5. Выделить в списке **В Normal...** стили для экспорта и скопировать их в текущий файл.

## Формат по образцу

Кнопка на вкладке **Главная** – группа **Буфер обмена**  **Формат по образцу** дает возможность скопировать формат с выделенного фрагмента на другой фрагмент текста. Для применения формата по образцу:

- Выделить фрагмент текста, форматирование которого используется как образец;
- Щелкнуть по кнопке , рядом с указателем мыши появится изображение кнопки;
- “Закрасить” кисточкой фрагмент, к которому применяется форматирование.

## Стили списков

Программа Word является достаточно гибким инструментом для работы со списками. Редактор позволяет создавать маркированные и нумерованные, одноуровневые и многоуровневые списки, сочетающие в себе номера и маркеры. Если начать абзац с некоторого числа (например, “1”) или маркера (например “\*”), то программа Word, соответственно пронумерует или промаркирует следующие абзацы после нажатия клавиши Enter. Создание списков относительно готового текста может быть произведено с помощью кнопок  на вкладке **Главная** – группа **Абзац**.

Как известно, списки считаются самыми капризными и непредсказуемыми элементами. А если в документе используются сложные многоуровневые списки,

то нередко случаи, когда редактор “самовольно” нарушает нумерацию списков, и все попытки пользователя исправить эту ситуацию, вызывают затруднения. Для достойного выхода из такой ситуации рекомендуется создать для списка свой стиль.

На вкладке **Главная** – группа **Абзац** выберите команду **Многоуровневый список**. Откроется следующее диалоговое окно.

В **Библиотеке списков** уже есть стандартные стили. Для создания пользовательского стиля:

1. Выбрать команду **Определить новый стиль списка**. Откроется диалоговое окно **Определение нового стиля списка**:
2. Введите название для стиля в поле **Имя**.
3. Щелкните кнопку **Формат** и выберите из раскрывшегося подменю команду **Нумерации**. Откроется другое диалоговое окно **Изменение Многоуровневого списка**.

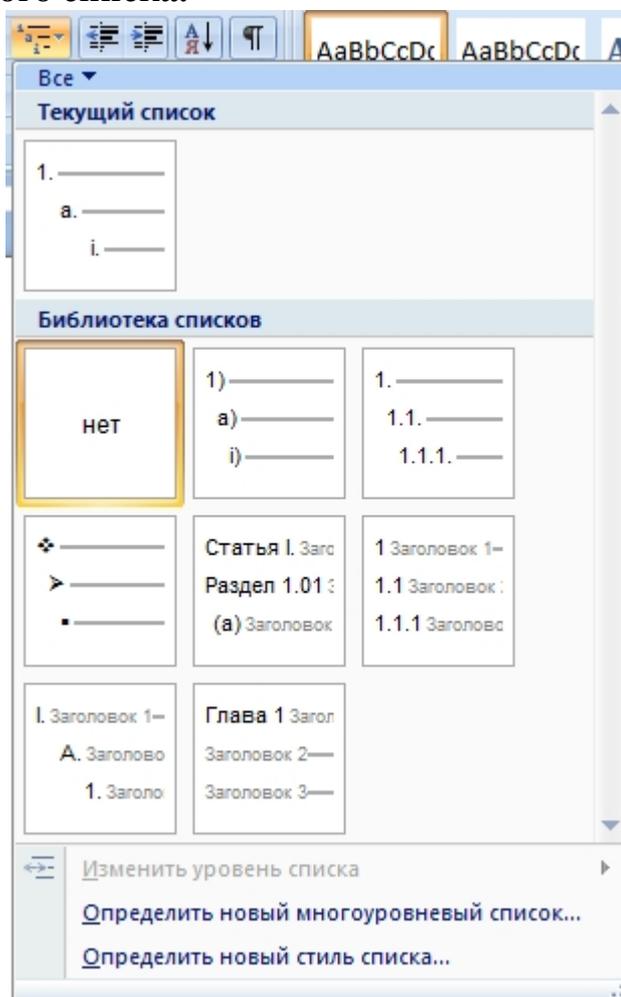


Рисунок 11. Окно Список

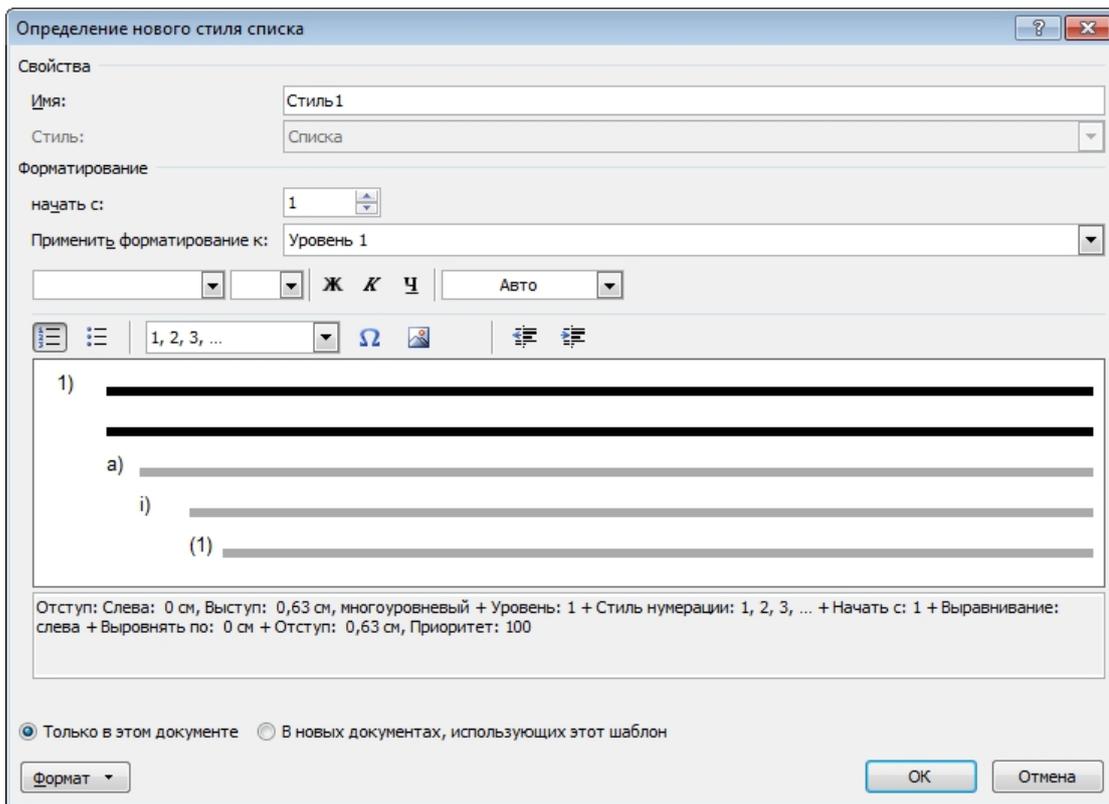


Рисунок 12. Окно определение нового стиля списка

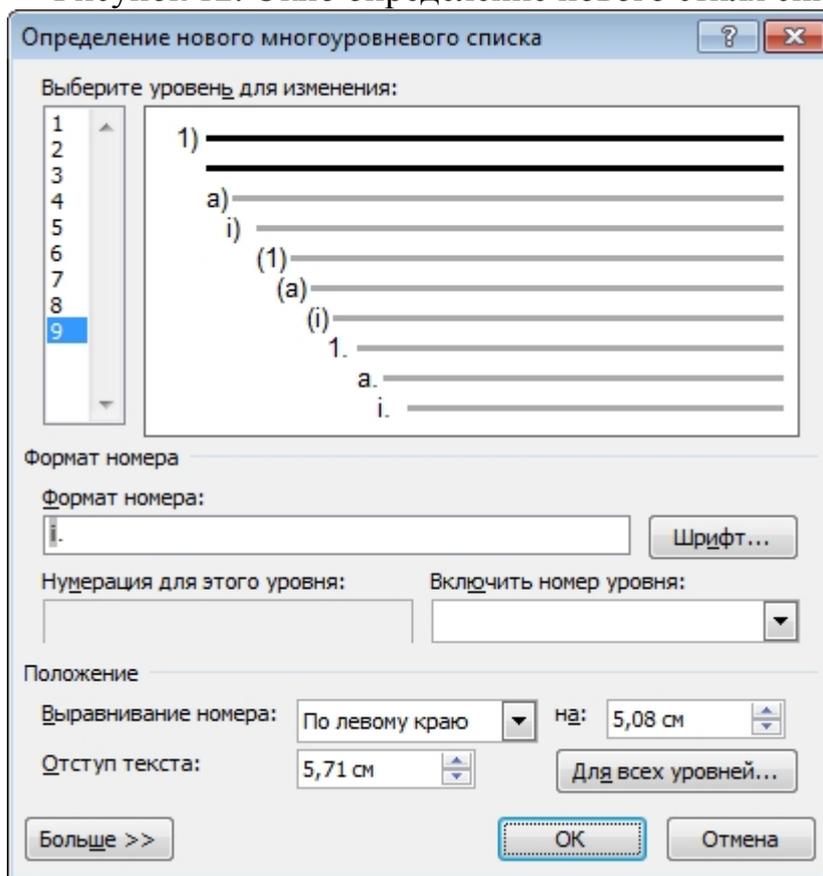


Рисунок 13. Изменение многоуровневого списка

4. В списке **Выберите уровень для изменения** выделите первый уровень списка (цифру 1).
5. Щелкните кнопку **Больше**.
6. В поле **Формат номера** установить свой формат или оставьте имеющийся.
7. В группе параметров **Положение** установите **Выравнивание номера** по левому краю (первая строка) на нужное значение, задайте величину табуляции (расстояние от левого поля страницы до текста) и, если необходимо, отступ от левого поля страницы для продолжающегося текста на второй строке. В окне **Образец** можно наблюдать все сделанные изменения.
8. Щелчок по кнопке **Шрифт** откроет одноименное диалоговое окно, в котором можно установить параметры для форматирования шрифта.
9. В списке **Выберите уровень для изменения** диалогового окна **Изменение многоуровневого списка** выберите второй уровень. Повторите для второго уровня вышеописанные действия.
- 10.ОК.

После того, как создан стиль для многоуровневого списка, можно приступить к применению этого стиля в документах.

Чтобы применить к тексту созданный стиль многоуровневого списка:

1. Выделите текст и выберите в списке ваш стиль. Текст мгновенно преобразуется в нумерованный список (первый уровень);
2. Выделите текст второго уровня, а затем нажмите кнопку **Увеличить отступ**;
3. Выделите текст третьего уровня, нажмите кнопку **Увеличить отступ** (вкладка **Главная** – группа **Абзац**) и т.д.

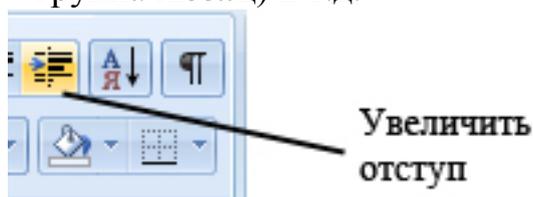


Рисунок 14. Увеличить отступ

Возможна ситуация, когда нумерация списка сбивается и для списка отображается начальный уровень списка (с цифры 1, например). В этом случае следует щелкнуть правой кнопкой мыши на номере списка и выбрать из контекстного меню команду **Продолжить нумерацию**.

При создании стиля списка в шаблоне документа, имеет смысл защитить документ и ограничить выбор разрешенных стилей. В этом случае есть хорошие шансы, что нумерация не собьется, и в документ не будут внедрены лишние стили, портящие форматирование документа.

Чтобы защитить документ выберите вкладку **Рецензирование** группа **Защитить** команду **Защитить документ**. Откроется область задач **Ограничить форматирование редактирование**:

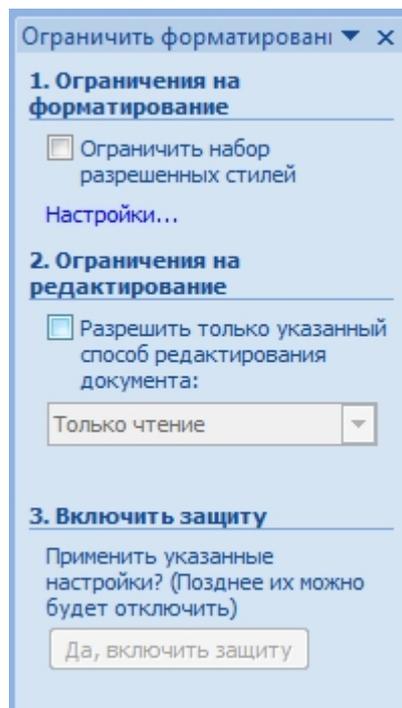


Рисунок 15. Защита документа

Установите флажок на параметре **Ограничить набор разрешенных стилей**, а затем нажмите кнопку **Да, включить защиту**. Откажитесь от установки пароля (если вы работаете один и не передаете шаблон другим людям).

## Технология форматирования таблиц

Достаточно распространенной формой представления текстовой информации является таблица. Основными структурными элементами таблицы являются: **ячейка**- клеточка, находящаяся на пересечении столбца и строки, **строка**, **столбец** и **блок ячеек**.

**Движение по таблице** осуществляется клавишей **Tab** или с помощью мыши.

### Вставка таблицы:

- Установите курсор в место вставки;
- Выберите вкладку **Вставка** – группа **Таблицы** команда **Таблица** – **Вставить таблицу...**;
- задайте число строк и столбцов.

Представим, что документ содержит достаточно большую таблицу, которая по длине не умещается на одной странице. В результате, на следующую страницу переносятся некоторые строки таблицы. Текст в таких строках разбивается и выглядит примерно так, как на рисунке ниже:



Рисунок 16. Разрыв строки таблицы

Чтобы предотвратить разрыв строк и перенос текста на следующую страницу:

- Поместите курсор ввода в ячейку или выделите таблицу целиком;
- Нажмите правую кнопку мыши, чтобы вызвать контекстное меню;
- В контекстном меню выберите команду **Свойства таблицы**. Откроется диалоговое окно, в котором нужно перейти на вкладку **Строка**;
- Снимите флажок с параметра **Разрешить перенос строк на следующую страницу**;

### Разбиение таблицы на две части

При работе с таблицей в документе может потребоваться вставить текст перед таблицей, например, ее название.

Поставьте курсор ввода в первую строку таблицы и нажмите сочетание клавиш **Ctrl+Shift+Enter** или выберите в дополнительной вкладке **Макет** - группа **Объединить** команду **Разбить таблицу**.

Если требуется разделить таблицу на две части:

- Поместите курсор ввода в ту строку, которая должна быть в начале второй таблицы;
- Нажмите комбинацию клавиш **Ctrl+Shift+Enter** или выберите в дополнительной вкладке **Макет** – группа **Объединить** команду **Разбить таблицу**.

Таблица разделится на две части, между которыми можно ввести обычный текст. Сочетание клавиш **Ctrl+Enter** вставляет разрыв страницы и таблица окажется в новом разделе.

### Преобразование таблицы в текст и текста в таблицу

Преобразование таблицы в текст:

1. Выделите таблицу, данные из которой преобразовываются в текст;

2. Выполните вкладка **Макет** – группа **Данные** команда – **Преобразовать в текст**. Откроется диалоговое окно **Преобразование в текст**.

В области **Разделитель** укажите знак разделителя, которым будут отмечены в тексте границы столбцов.

Строки будут разделены знаком абзаца.

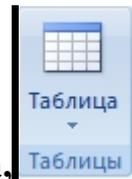
Для преобразования текста в таблицу следует проделать следующие действия:

1. Вставьте знаки разделителей, такие как запятая или знак табуляции, в местах, где текст должен быть разбит по столбцам.

Используя знак абзаца, укажите, где должны начинаться новые строки.

Например, в списке с двумя словами в строке, вставьте запятые или знаки табуляции после первого слова, чтобы создать таблицу с двумя столбцами.

2. Выделите текст, который требуется преобразовать.



3. На вкладке **Вставка** в группе **Таблицы** выберите пункт **Таблица**, а затем выберите команду **Преобразовать в таблицу**.
4. В диалоговом окне **Преобразовать в таблицу** в области **Разделитель** укажите знак разделителя, использованный в тексте.
5. В поле **Число столбцов**, проверьте количество столбцов.

Если количество столбцов отличается от ожидаемого, возможно, пропущен знак разделителя в одной или нескольких строках текста.

6. Укажите остальные необходимые параметры.

В документе появится таблица с вашим текстом

В процессе работы встречается ситуация, когда вставляемая таблица либо выходит за границу листа, либо занимает очень мало места на странице. Если необходимо таблицу расположить по всей ширине окна документа (от левого до правого поля) то:

- Выделить таблицу (для этого можно навести указатель мыши на таблицу, в левом верхнем углу таблицы появится небольшой квадратик с “плюсом”, щелчок мышью на этом квадратике и таблица будет выделена),
- Вкладка **Макет** – группа **Размер ячейки** команда **Автоподбор** – **Автоподбор по ширине окна**.

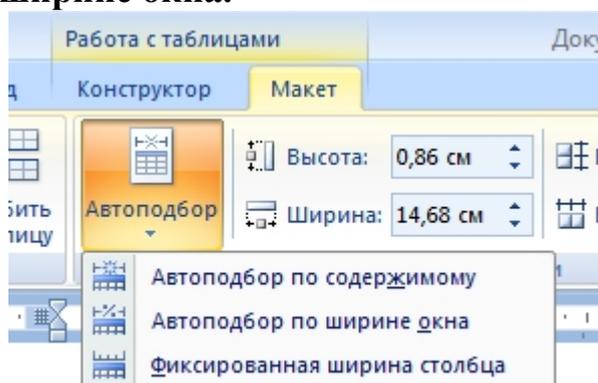


Рисунок 17. Автоподбор ширины таблицы

## Разделитель окна

Иногда необходимо видеть сразу две части одного документа, например, чтобы сравнить их или сделать перевод.

В этом случае необязательно открывать два отдельных окна. Выберите вкладку **Вид** – группу **Окно** – команда **Разделить**, определите место, где разместите разделитель и щелкните мышью. Или найдите справа от линейки, над полосой прокрутки, узкий разделитель (когда вы подводите к нему курсор мыши, появляется стрелка). При нажатой левой клавиши мыши, установите разделитель в нужном месте окна.

Чтобы вернуться к одному окну, выберите во вкладке **Вид** – группу **Окно** – команда **Снять разделение** или перетащите разделитель на его обычное место. Этот способ работает не только в Word, но и в Excel и во многих других программах.

## Создание формул. Панель MS Equation Editor

С помощью MS Equation, входящего в состав Microsoft Office, можно вставлять в текст математические формулы. Формулы могут быть достаточно сложными в написании, и могут быть добавлены в любое место документа.

Для вставки в документ формулы выполните следующие действия:

1. Поместите текстовый курсор в то место, где должна появиться формула.

Вкладка **Вставка** – группа **Текст** команда **Объект**.

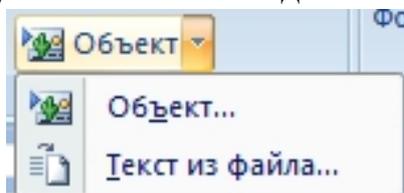


Рисунок 18. Команда Объекта

2. На вкладке **Создание** выберите элемент **Microsoft Equation 3.0**

При этом появится панель инструментов **Формула**. Вместе с панелью инструментов появится пустое поле для ввода формулы с пунктирной рамкой, называемое **слотом**.

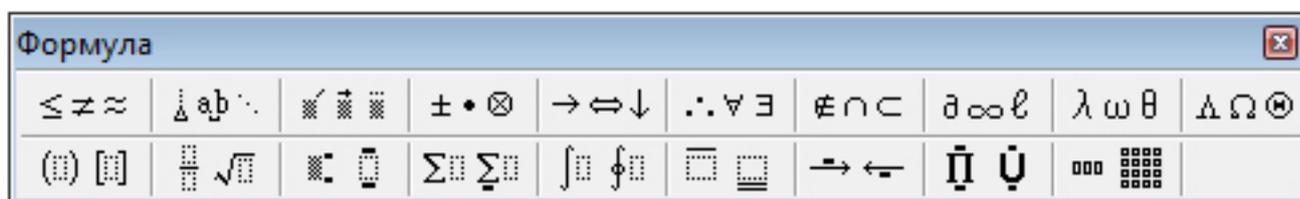


Рисунок 19. Панель инструментов Формула

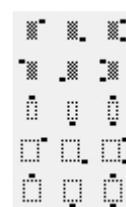
**Слот** – это место, в котором находится часть формулы. Например, дробь имеет два слота: для числителя и знаменателя.

### Состояние формулы

При записи и редактировании формулы ввод очередного символа выполняется в основную линию ввода – место вводимого символа автоматически помечается **слотом**. С помощью мыши или клавиши **Tab** можно перемещать курсор из слота в слот и таким образом, набирать необходимые элементы формулы.

Рассмотрим пример создания формулы:

- Создайте новый документ и вызовите редактор формул
- Установите курсор в слот и наберите текст «у = ». Обратите внимание на начертание символов. Редактор формул автоматически применяет к каждому символу математический стиль (чтобы изменить их внешний вид, в меню **Стиль** выберите другой стиль оформления вводимого текста, например, **Текст**).
- На панели инструментов **Формула** нажмите на кнопку **Шаблоны дробей и радикалов** и выберите шаблон, соответствующий кубическому корню.
- Установите курсор в слот степени корня и введите число 3.
- Установите курсор в слот ввода содержимого корня. Здесь вам необходимо ввести дробь. Нажмите на кнопку **Шаблоны дробей и радикалов**  и выберите шаблон дроби.
- В формуле появляется два новых слота для ввода числителя и знаменателя дроби.
- В формуле появятся два новых слота для ввода числителя и знаменателя дроби.
- Установите курсор в слот для ввода числителя дроби и введите **x**.
- Установите курсор после символа **x** и нажмите на панели инструментов **Формула** кнопку **Шаблоны верхних и нижних индексов**. Выберите из раскрывающегося списка шаблон верхнего индекса и введите туда число 8.
- Чтобы ввести текст «+4», установите курсор после верхнего индекса в слот ввода числителя дроби и введите необходимые символы.
- Установите курсор в слот ввода знаменателя дроби и введите число 2.
- Для продолжения знака корня поместите курсор после дроби в слот для ввода подкоренного выражения и введите оставшуюся часть формулы.
- Если установить курсор неправильно, например, в слот ввода основного текста формулы, то знак корня продолжаться не будет, и вы получите другую формулу.



$$y = \sqrt[3]{\frac{x^8 + 4}{2}}$$

Чтобы закончить ввод формулы, щелкните вне области ввода или нажмите клавишу **Esc**, и вы вернетесь обратно в окно MS WORD со стандартными панелями инструментов. Кроме того, объект Microsoft Equation – это стандартный

объект, для которого можно настраивать все стандартные параметры: обтекание текстом, заливку, цвет границ и т.п.

Если по каким-то причинам не устраивают параметры по умолчанию, то их можно переопределить. Для изменения стандартных типов размеров используется пункт меню **Размер – Определить**.

Применение этой команды позволяет быстро изменить размер всех символов имеющих определённый тип.

В окне диалога рядом с нужным типом размера требуется указать размер символа в пунктах.

Кнопка **Применить** позволяет увидеть влияние изменений на формулу перед их сохранением, а кнопка **По умолчанию** служит для восстановления размеров установленных по умолчанию.

При внесении изменений в окне диалога **Размеры** редактируемая формула изменяется в соответствии с ними. Остальные формулы, сохраненные в документе, не изменятся, если после изменения размера шрифтов они не будут редактироваться.

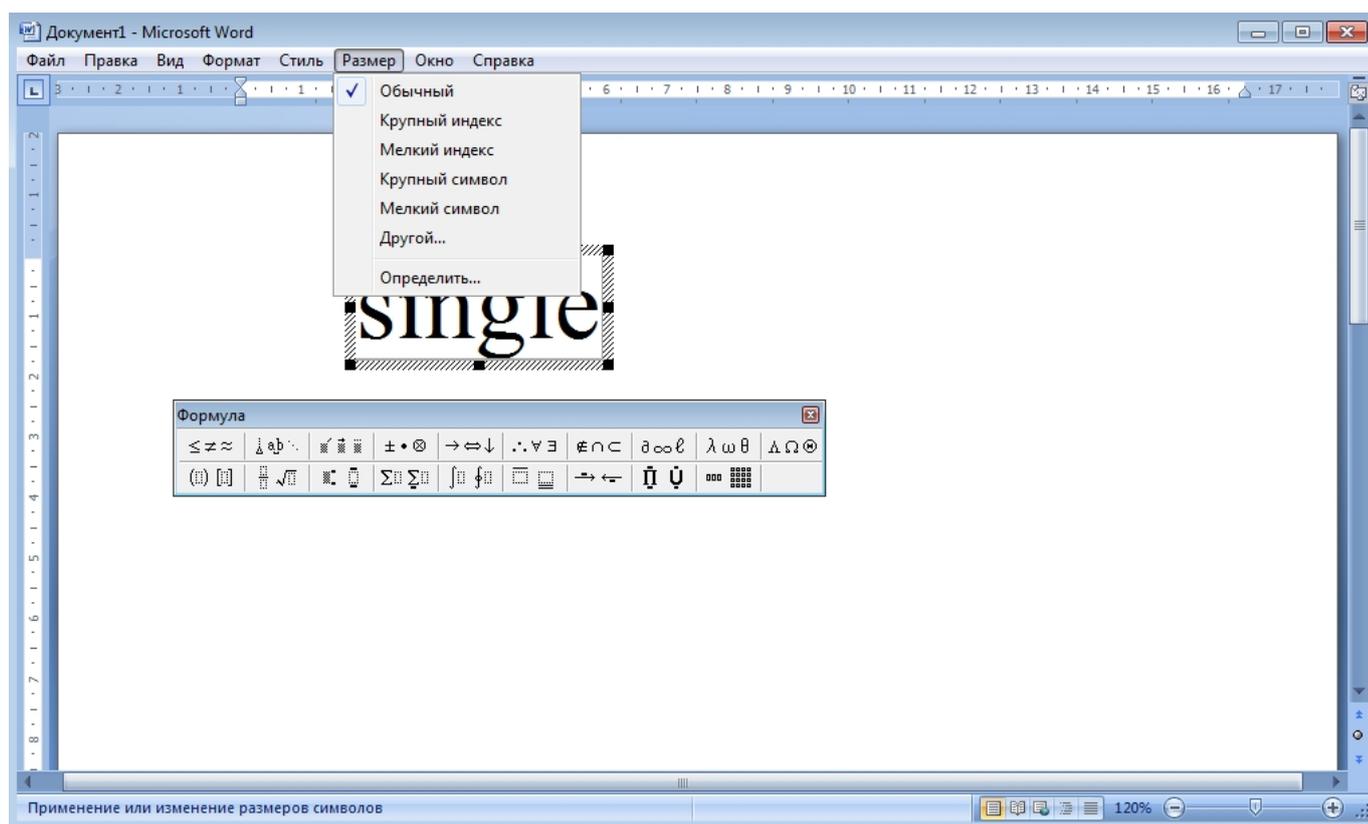


Рисунок 20. Окно редактора формул

Допускается задание размера элемента в процентах к обычному размеру. Тогда при изменении обычного размера все стандартные типы размеров изменяются пропорционально.

По окончании редактирования формулы и закрытии окна редактора созданная формула вставляется в документ в позиции курсора ввода.

Напомним, что повторный вход в уже созданную формулу для редактирования осуществляется двойным щелчком мыши. Размер формулы можно изменять так

же, как и размер графических объектов. Для этого нужно выделить формулу с помощью одинарного щелчка мыши, а затем перетянуть в нужное положение один из черных маркеров по периметру формулы.

## Режимы просмотра документа

Работая с документами в **Word**, можно использовать один из основных режимов. Вкладка **Вид** – группа **Режимы просмотра документа**. Режим **Черновик** является стандартным режимом работы с документов. Он наиболее удобен для выполнения большинства операций по редактированию и форматированию текста.

Режим **Веб-документ**, в котором разметка документа изображена упрощенно, служит для облегчения просмотра документа на экране. Текст изображается символами большего размера, длина строк устанавливается равно размеру окна. Вид текста в этом режиме не соответствует тому, как он будет выглядеть в напечатанном виде.

Режим **Веб-документа** и **Разметка страницы** позволят задавать фон определенного цвета для страницы. Для задания фона служит вкладка **Разметка страницы** – группа **Фон страницы** команда **Цвет страницы**. Особые виды фона можно задать, выбрав подпункт **Способы заливки**. Здесь уже можно установить фоновый рисунок.

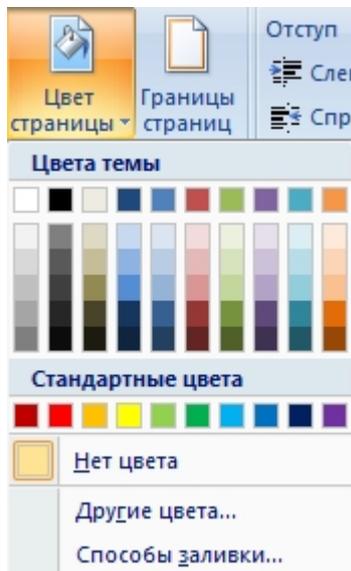


Рисунок 21. Установка фона документа

Фон, установленный в режиме **Веб-документ**, не влияет на печать документа на принтере, а только позволяет разнообразить вывод на экран.

Режим **Разметка страницы** обеспечивает отображение документа постранично в точном соответствии с тем, как он будет выведен на печать, то есть страницы выводятся по отдельности с соблюдением полей.

Режим **Структура** позволяет сворачивать и разворачивать заголовки и относящийся к ним текст, работать с заголовками определенной глубины вложенности и т.п. Данный режим можно использовать при работе с документами, состоящими из нескольких отдельных документов.

Режим **Предварительный Просмотр** позволяет видеть несколько страниц документа в уменьшенном размере в таком виде, в котором они будут напечатаны (см. рис. 1.4).

## Советы по оформлению документов

Оформление страниц формата А4. Рекомендуемые поля: левое – 10 мм; правое – 20мм, верхнее – 20 мм; нижнее – 20 мм. Изменить размер полей можно выполнив вкладка **Разметка страницы** – группа **Параметры страницы**, команда **Поля**.

## Правила набора документа

Знаки препинания. **Точка** не ставится:

- в конце заголовков, названиях колонтитулов, в подписях таблиц и рисунков;
- во многих сокращениях (*мм, см, кг и пр.*)

В качестве **разделителя целой и дробной части** используется **запятая**, например: 36,6

После любого знака препинания ставится **пробел**, исключения: сокращение инициалов (Пушкин А.С.), некоторые стандартные сокращения (и т.д. и т.п.).

**Многоточие** вставляется сочетанием **Alt+Ctrl+точка**.

**Кавычки** следует использовать только угловые. Чтобы включить/отключить этот режим, выполните следующие действия:

1. Вывести на панель быстрого доступа **Автоформат**.
2. Вызвать окно **Автоформат**.
3. Щелкнуть кнопку **Параметры** – вкладка **Автоформат**
4. В группе **Заменять** установить/снять флажок **“Прямые” кавычки «парными»**.

**Пробелы**. Различаются пробелы: обычный, неразрывный, принудительный.

**Неразрывные пробел** ставиться, если необходимо запретить перенос на следующую строку в данном месте **Ctrl+Shift+пробел**.

**Принудительный разрыв** строки ставиться, если необходимо перейти на новую строку без начала нового абзаца **Shift+Enter**.

**Переносы**. В большинстве случаев автоматическая расстановка переносов работает нормально, однако следует следить за тем, чтобы разрыв слова переносом не создавал неблагозвучных или комичных отрывков. На двусторонних документах следует избегать переноса в последней строке правой (нечетной) страницы.

Включить/отключить расстановку переносов возможно на вкладке **Разметка страницы** группа **Параметры страницы** команда **Расстановка переносов**.

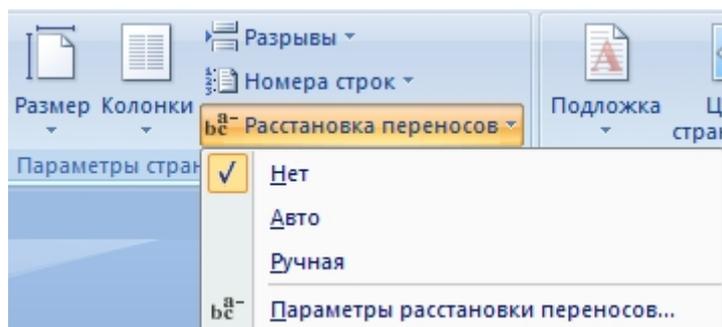


Рисунок 22. Расстановка переносов

### Минус, дефис, короткое тире, длинное тире

В текстах, грамотно подготовленных, можно встретить минусы, дефисы и два рода тире: короткое тире– и длинное тире—. Короткое тире обычно используется при указании диапазона, например, “страницы 13-14”, а также в контекстах, вроде “Рис. А-12”. Длинное тире используется в качестве знака пунктуации в предложениях —.

Короткое тире, подобно дефисам, обычно не имеет пробелов ни с одной из сторон. Это же касается длинного тире в англоязычных текстах, а при наборе текстов на русском языке следует, напротив, помнить о пробелах с двух сторон длинного тире.

В Word’е чтобы поставить дефис, нажмите клавишу <->, для короткого тире используйте комбинацию <Ctrl>+<Gray ->, для длинного — <Ctrl>+<Alt>+<Gray - > (клавиша <Gray -> — это клавиша со знаком “-” на дополнительной цифровой панели клавиатуры). Минус в тексте лучше изображать в шрифте Symbol.

### Советы короткой строкой

1. Иногда, при наборе текста, забывают отключить клавишу **CAPS LOCK**, и тогда весь текст идет большими буквами или наоборот. Чтобы исправить ситуацию:
  - выделите фрагмент текста, в котором необходимо изменить регистр,
  - удерживая клавишу **Shift**, нажмите функциональную клавишу **F3**.
2. В Word’е можно перемещать абзацы в тексте. Для этого сделать абзац текущим (установить курсор внутри абзаца) или выделить несколько абзацев, затем, удерживая клавиши **Shift+Alt** перемещать его с помощью клавиш управления курсором “Стрелка вверх” и “Стрелка вниз”. “Стрелка

вправо” или “Стрелка влево”, позволяет менять стиль выделенного фрагмента.

3. Для выделения больших участков текста используйте клавишу **Shift**. Если она нажата, вы легко сможете выделить любой участок.
4. Кроме того, выделить вертикальный блок можно с помощью мышки, удерживая клавишу **Alt**.

### Удаление лишних пробелов

При копировании текста из другого документа или из Интернета возможна ситуация, когда между словами вставляются дополнительные пробелы. Исправить такую ситуацию возможно с помощью вкладки **Главная** – группа **Редактирование** команда **Заменить**. Откроется диалоговое окно **Заменить**. Для доступа к параметрам поиска нажмите кнопку **Больше**, если диалоговое окно представлено в краткой форме,

1. Удалите всё, что находится в поле **Найти** и в поле **Заменить**. Снимите все галочки с режимов в группе **Параметры поиска**, но оставьте или поставьте галочку **Подстановочные знаки**. Выберите в открывающемся списке **Направление** **Везде**.
2. Выделите весь текст документа, нажав на сочетание клавиш “CTRL+A”.
3. Щелкните мышью в поле **Найти**, нажмите пробел (в поле появится мигающий курсор), а потом введите следующее выражение: {2;}

Здесь в фигурных скобках введена цифра 2, а сразу за ней точка с запятой. Это выражение означает, что будет произведен поиск двух и более расположенных подряд вхождений того знака, который стоит перед открывающейся фигурной скобкой. В нашем случае будет произведен поиск двух и более пробелов.

1. Щелкните мышью в поле **Заменить на** и нажмите клавишу пробел.
2. Нажмите кнопку **Заменить все**. Таким образом, если будет найдено несколько подряд стоящих пробелов, то все они заменятся на один пробел.
3. После окончания поиска и замены закройте диалоговое окно **Найти и заменить**. Снимите выделение текста, нажав на клавиатуре клавишу стрелки.

При вводе текста распространенная ошибка – пробел перед знаком препинания. Для того чтобы быстро исправить ее, воспользуемся знакомой командой вкладки **Главная** – группа **Редактирование** – команда **Заменить**.

1. Введите в поле **Найти** пробел, а потом следующее выражение: {1;}([.,;\!/?]). Это выражение буквально означает, что будет произведен поиск одного и более расположенных подряд пробелов, стоящих перед любым перечисленным в квадратных скобках знаком препинания.

2. Перейдите в поле **Заменить на** и введите \1
3. Нажмите кнопку **Заменить все**.
4. После окончания поиска и замены закройте диалоговое окно **Найти и заменить**. Снимите выделение текста, нажав на клавиатуре на клавишу стрелки.



7. В поле **Заменить** ввести пробел.

8. Нажмите кнопку **Заменить**, если контролировать процесс замены, или кнопку **Заменить все** (без запросов на подтверждение).

После этой операции откроется окно с уведомлением о количестве произведенных замен и предложением проверить остальную часть документа. Если дальнейшая замена не требуется, откажитесь от нее и закройте диалоговое окно.

В случае, если вместо знака разрыва строки имеется абзацный разрыв, то для его удаления в диалоговом окне **Найти и заменить** необходимо снять все флажки с параметров поиска, в том числе и с параметра **Подстановочные знаки**.

В поле **Найти** нужно вставить символ абзаца, выбрав его из списка по щелчку на кнопке **Специальный**. Дальнейшие действия аналогичны описанным выше.

## Сноска

**Сноска** – это дополнительный текст, связанный с определенным местом в основном тексте, которое отмечается специальным символом – символом сноски. Сам текст сноски размещается в другом месте, но ему предшествует тот же самый символ сноски. Можно выделить два вида сносок:

- Обычная сноска размещается внизу той страницы, на которой находится символ сноски, и отделяется от основного текста горизонтальной чертой;
- Концевая сноска размещается в конце раздела документа или всего документа, в котором присутствует символ сноски, и также отделяется от основного текста горизонтальной чертой.

Чтобы создать обычную или концевую сноску нужно:

- Установить курсор после текста, который должен быть отмечен символом сноски.
- Выбрать вкладку **Ссылки** – группа **Сноски** открыть диалоговое окно **Сноски**.
- В появившемся диалоговом окне установить переключатель сноски или концевые сноски.
- Изменить систему нумерации строк в группе **Формат** или принять предлагаемые параметры. Кнопка **Символ** позволяет в качестве ссылки на сноску выбрать символ.
- Щелкнуть **Вставить**.
- В нижней части экрана появится маркер сноски и мигающий курсор.
- Ввести текст сноски.

Чтобы удалить обычную или концевую сноску, нужно выделить ее символ и нажать клавишу **“Delete”** или **“Backspace”**. При удалении символа сноски текст сноски удаляется автоматически.

## Размещение в тексте двух одинаковых сносок

1. Вставить первую сноску в документе как обычно. Поместить курсор ввода в то место документа, где необходимо вставить вторую сноску.
2. Выбрать вкладку **Ссылки** – группа **Названия** команду **Перекрестная ссылка**. Откроется одноименное диалоговое окно.
3. В поле **Тип ссылки** выбрать строку **Сноска**, в поле **Вставить ссылку на** выбрать строку **Номер сноски (форм)**.
4. Установить флажок на параметре **Вставить как гиперссылку**.
5. Выбрать из списка сносок ту сноску, на которую необходимо повторно сослаться.
6. Нажать кнопку **Вставить**.

В текст будет вставлен символ сноски под тем номером (маркером), который соответствует номеру (маркеру) первой сноски.

## Автоматическая вставка подписи к рисунку, таблице, объекту

Наличие подписей придает профессиональный вид вашим документам и предоставляет вам еще много преимуществ и возможностей, таких, например, как создание списка этих объектов.

Нужно избегать ручного подписывания. Редактор Word имеет функцию автоматического добавления названий к различным объектам: таблицам, диаграммам, мультимедиа-файлам и др.

Чтобы настроить редактор на автоматическую вставку названия одновременно с вставкой рисунку или таблицы, нужно сделать следующее:

1. Выполнить вкладка **Ссылки** – группа **Названия** – команда **Вставить название**. Откроется диалоговое окно:

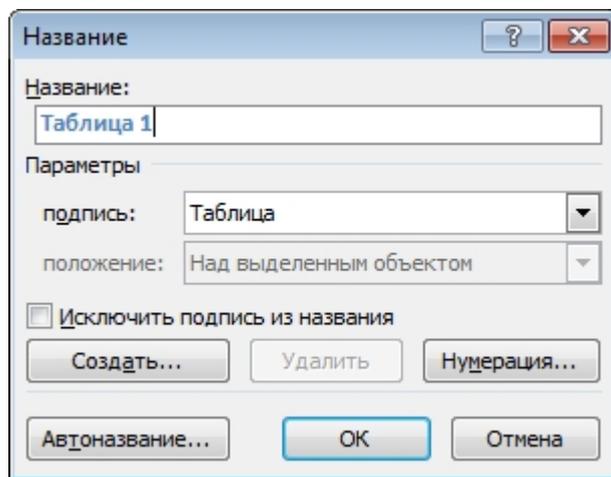


Рисунок 24. Вставка подписи к таблице

2. Нажать кнопку **Автоназвание**, откроется другое диалоговое окно, в котором следует выбрать тот объект, к которому нужно добавлять подпись. Для таблиц – **Таблица Microsoft Word** и т.д.
3. Можно выбрать из списков поля **Подпись** или поля **Положение** нужные значения или создать свои значения, нажав в этом окне на кнопку **Создать**.
4. Щелчок по кнопке **Нумерация** позволяет выбрать стиль нумерации. По умолчанию используется стиль 1, 2, 3... Кроме того, можно включить в название объекта номер главы, в которой располагается данный объект.
5. Нажать ОК для сохранения установок и закрытия окон.

Теперь при вставке таблицы автоматически будет вставляться название Таблица 1, Таблица 2 и т.д.

## Разделы

**Разделы** - это части документа, которые могут быть отформатированы независимо друг от друга. Разделы позволяют начать каждую главу с нечетной страницы и задать колонтитулы, нумерацию страниц и строк для этой главы. При верстке разделы позволяют менять количество колонок текста в пределах одной страницы. По умолчанию весь документ представляет собой один раздел. Для того чтобы увеличить количество разделов, необходимо вставить разрыв раздела:

- Поместить курсор в то место, где должен начать новый раздел.
- Выбрать вкладку **Разметка страницы** – группа **Параметры страницы** – команда **Разрывы**.
- В группе **Разрывы разделов** выберите тип разрыва раздела, соответствующий необходимым изменениям формата.

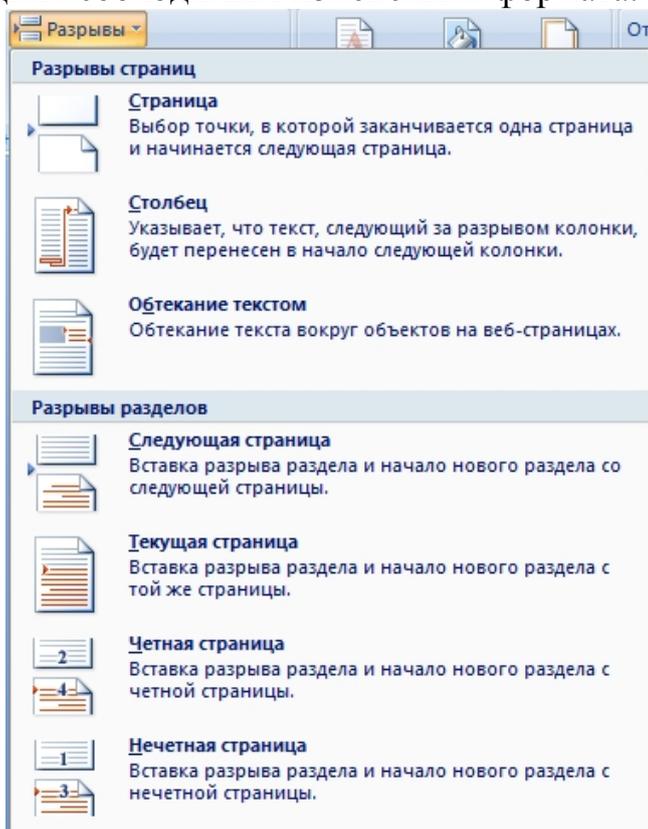


Рисунок 25. Окно вставки разрыва

Удалить разрыв раздела можно с помощью клавиш “Delete” или “Backspace”, предварительно выделив их. Чтобы удалить все разрывы раздела в документе нужно воспользоваться вкладкой **Главная** – группа **Редактирование** и заменить символ **Разрыв раздела** на пустое поле (Рисунок 23).

## Шаблоны

Создаваемый в Word документ может базироваться на определенном шаблоне. Основу шаблона составляют текст и графика. Шаблон можно рассматривать как стандартный бланк, подготовленный для заполнения. В шаблоне устанавливаются поля страницы, колонтитулы и другие параметры документа, хранятся стили форматирования. Кроме стилей, которые являются обязательным элементом шаблона, каждый шаблон может содержать макрокоманды, элементы автотекста, установки, выполненные при адаптации команд, панелей инструментов и клавиатуры. Любые элементы шаблона доступны для изменения.

После запуска Word открывается пустой документ, базирующийся на шаблоне **normal.dot**, который представляет собой чистый лист бумаги формата А4 и содержит все стандартные стили форматирования. Шаблон **normal.dot** активен по умолчанию.

Существует три способа создания шаблонов документов:

- На основе существующего шаблона
- Путем модификации существующего шаблона
- Создание совершенно нового шаблона.

Если ни один из стандартных шаблонов Word не удовлетворяет требованиям пользователя, следует разработать новый шаблон. Для создания нового шаблона необходимо выполнить следующие действия.

- Выберите в меню **Office – Создать**;

## Глава 7. Практическая работа с текстовым документом в процессоре word

Для начала работы с текстом пользователю необходимо знать правила работы с документом и правила оформления текста. Процесс начинается с ввода текста в компьютер (либо это набор вручную с помощью клавиатуры, либо сканирование документа, либо голосовой ввод). Далее текст редактируют, форматируют и оформляют под нужные требования. Последнее является более творческим процессом, что для пользователя достаточно приятно. Основные действия связанные по обработке текстовой информации производятся в одном из видов прикладного программного обеспечения – текстовых редакторах и процессорах.

Для всех текстовых программ основой являются четыре парадигмы: символ, слово, абзац, документ. Совокупность функций обслуживающих эти парадигмы определяют возможности и функциональность того или иного редактора и процессора. Наиболее полно и ярко это представлено в текстовом процессоре пакета Microsoft Office – Microsoft Word.

1. Запустите редактор Word.
2. Введите заданные стихи Шекспира и оформите таким образом как в примере (определите тип, начертание и размер шрифта, а также оформление абзаца). Далее, под набранными строками сделайте их копию и расположите по номерам в порядке возрастания, чтобы получился текст оригинала (номера не убирать).

\* \* \*

5 И во всем, без дальних слов,  
2 Верен, в счастье и беде;  
6 Он помочь тебе готов.  
8 Верный друг и льстец негодный.  
3 Грусть твоя его тревожит,  
1 Настоящий друг везде  
4 Ты не спишь – он спать не может,  
7 Да, по действиям несходны

*В. Шекспир*

\*\*\*  
Настоящий друг везде  
Верен, в счастье и беде;  
Грусть твоя его тревожит,  
Ты не спишь – он спать не  
может,

3. Полученный оригинал стихов оформите шрифтом без засечек 10 кеглем.

4. Перейдите (используя вставку разрыва или комбинации перехода) на новый лист документа.

5. Введите отрывки из книги (Энциклопедия мысли. Книга вторая. – СПб.: «Кристалл», 1997. – 592 с.) и оформите введенные абзацы по указанному примеру (без переносов, выравнивание по ширине, полуторный интервал). Вставьте сноски внизу страницы, описывающих авторов высказываний.

*Наихудшее, чему может научиться молодежь, – легкомыслие. Ибо последнее порождает те удовольствия, из которых развивается порок.*

Демокрит<sup>1</sup>

Тот, кто делает добро другому, делает добро самому себе, не в смысле последствий, но самим актом делания добра, так как сознание сделанного добра само по себе дает уже большую радость.

Сенека<sup>2</sup>

*Кто рассчитывает обеспечить себе здоровье, пребывая в лени, тот поступает так же глупо, как и человек, думающий молчанием усовершенствовать свой голос.*

Плутарх<sup>3</sup>

6. К первому слову каждого введенного изречения древних мыслителей сделайте примечание - "Энциклопедия мысли. Книга вторая. – СПб.: «Кристалл», 1997. – 592 с."

7. Создайте горизонтальную черту под текстом, набрав несколько дефисов подряд и нажав Enter

8. В точности введите текст, ниже сделайте его копию и, используя проверку орфографии, исправьте в копии ошибки.

---

<sup>1</sup> Демокрит из Абдер (ок.460 – после 360 гг. до н.э.) – древнегреческий философ-энциклопедист

<sup>2</sup> Сенека Луций Анней (ок.4 г. до н.э. – 65 г.) – римский философ-стоик, поэт, государственный деятель

<sup>3</sup> Плутарх (ок.46 г. – ок.127 г.) – древнегреческий историк, философ, писатель

*Прэ работи на кампьютере ниобходимо саблюдаць правіла тэхнекі  
бесопаснасці!!!*

9. Стварыце яшчэ адну гарызонтальную чэрту пад тэкстам

---

10. Введзіце тэкст і аформіце яго ў дзве калонкі з дзеліцельнай чэртой і  
неаднакавай шырыняй калонак як у прыкладзе.

*Когда между собакой и кошкой вдруг  
возникает дружба, то это не иначе,  
как союз против повара.*

С. Цвейг

Противник, вскрывающий  
ваши ошибки, гораздо  
полезнее, чем друг,  
скрывающий их.

*Леонардо да Винчи*

11. Создайте по примеру оглавление (соблюдая шрифты и размеры), используя для отточия формат табуляции

## **СОДЕРЖАНИЕ**

<b>Введение</b> .....	<b>3</b>
<b>Глава 1.</b> .....	<b>5</b>
<b>1.1. Раздел обзорный</b> .....	<b>5</b>
<b>1.2. Раздел актуальности</b> .....	<b>10</b>
<b>1.3. Раздел постановки задачи</b> .	<b>15</b>
<b>Глава 2.</b> .....	<b>20</b>
<b>2.1. Исследовательская часть</b> ..	<b>20</b>
<b>2.2. Проекционная часть</b> .....	<b>25</b>
<b>2.3. Расчетная часть</b> .....	<b>30</b>
<b>Глава 3</b> .....	<b>36</b>
<b>3.1. Внедрение</b> .....	<b>36</b>
<b>3.2. Его результаты</b> .....	<b>38</b>
<b>Заключение</b> .....	<b>39</b>

12. Скопируйте созданное оглавление и под ним же вставьте его как рисунок!!!

13. Увеличьте размер получившегося рисунка.

14. Перейдите на новую страницу.

15. Сохраните документ в папке "Мои документы" под именем "Практика ФИО", указав вместо ФИО свою фамилию и инициалы.

16. Задайте в параметрах страницы все поля по 2 см.

17. Введите несколько математических выражений:

$$a^2 + b^2 = c^2 \quad \alpha + \beta = \gamma \quad x = x_0 + vt + \frac{at^2}{2}$$

18. Задайте нумерацию страниц сверху посередине, а в нижний колонтитул слева поместите текущую дату. Сохраните файл.
19. В новом файле оформите приглашение на празднование нового года (в виде открытки) и вставьте в него стандартный рисунок из картинок Word.
20. Практическая работа с объектами Microsoft Word. Создайте по примеру таблицы в отдельном файле.

**ТАБЛИЦА № 1**

Таблица для изучения возможностей ее оформления						
№ №	Фамилия	Тест				Примечание
		1	2	3	4	
1	Иванов	+			+	Тест сдан на 4
2	Петров		+	+	+	Тест сдан на 5
3	Сидоров			+		Тест сдан на 3

Таблица №2

### Планирование работы

№	Раздел	Виды работ				
		Теория	СР	Создание таблиц	ДЗ	Итог
1	<b>Таблицы</b>	+		+	+	+
2	<b>Объекты MS Word</b>	+	+		+	+
	<b>ИТОГО:</b>	+	+	+	+	+

21. Редактор формул. Создайте и оформите текст, как показано ниже (используйте 14 размер шрифта). Сохраните выполненную работу в отдельном файле.

### Формула Симпсона

Из формулы

$$\int_a^b f(x)dx = (b-a) \sum_{i=0}^n H_i y_i$$

при  $n = 2$  (это равносильно замене подынтегральной функции  $f(x)$  на отрезке  $[a, b]$  квадратичной функцией) получим

$$H_0 = \frac{1}{2} \cdot \frac{1}{2} \int_0^2 (t-1)(t-2)dt = \frac{1}{6};$$

$$H_1 = -\frac{1}{2} \cdot \frac{1}{1} \int_0^2 t(t-2)dt = \frac{2}{3};$$

$$H_2 = \frac{1}{2} \cdot \frac{1}{2} \int_1^2 t(t-1)dt = \frac{1}{6}.$$

Поэтому, с учетом  $(b-a) = 2h$ , получим

$$\int_a^b f(x)dx \approx \frac{h}{3} (y_0 + 4y_1 + y_2).$$

Эта формула называется формулой парабол или формулой Симпсона.

22. Создание деловых бумаг.

При выполнении данной работы необходимые шаблоны загрузите из интернета, для этого воспользуйтесь главной кнопкой «Office»  и выберите команду «Создать».

Для создания календаря в разделе **Шаблоны** из части **Microsoft Office online** выберите «Календари».

С помощью мастера календарей создайте одностраничный календарь на текущий год.

- 1) Разместите в качестве заголовка название университета.
- 2) Поместите на календарь логотип университета.
- 3) В созданном календаре отметьте дату Вашего рождения, используя заливку ячейки и форматирование текста.
- 4) Выполните предварительный просмотр документа. Ваш календарь должен размещаться не более, чем на одной странице.
- 5) Сохраните полученный календарь в отдельном файле (**doc**).



## 1 Январь

Пн	Вт	Ср	Чт	Пт	Сб	Вс
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
<sup>23</sup> <sub>30</sub>	<sup>24</sup> <sub>31</sub>	25	26	27	28	29

## 2 Февраль

Пн	Вт	Ср	Чт	Пт	Сб	Вс
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29				

## 3 Март

Пн	Вт	Ср	Чт	Пт	Сб	Вс
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

## 4 Апрель

Пн	Вт	Ср	Чт	Пт	Сб	Вс
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
<sup>23</sup> <sub>30</sub>	24	25	26	27	28	29

## 5 Май

Пн	Вт	Ср	Чт	Пт	Сб	Вс
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

## 6 Июнь

Пн	Вт	Ср	Чт	Пт	Сб	Вс
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

## 7 Июль

Пн	Вт	Ср	Чт	Пт	Сб	Вс
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
<sup>23</sup> <sub>30</sub>	<sup>24</sup> <sub>31</sub>	25	26	27	28	29

## 8 Август

Пн	Вт	Ср	Чт	Пт	Сб	Вс
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

## 9 Сентябрь

Пн	Вт	Ср	Чт	Пт	Сб	Вс
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

## 10 Октябрь

Пн	Вт	Ср	Чт	Пт	Сб	Вс
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

## 11 Ноябрь

Пн	Вт	Ср	Чт	Пт	Сб	Вс
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

## 12 Декабрь

Пн	Вт	Ср	Чт	Пт	Сб	Вс
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
<sup>24</sup> <sub>31</sub>	25	26	27	28	29	30

23. В новом документе сделаем титульный лист следующего содержания:

**Министерство образования и науки Российской Федерации**  
**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**  
**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**  
**МЕХАНИКИ И ОПТИКИ**

**ЛАБОРАТОРНАЯ РАБОТА**  
**ПО ИНФОРМАТИКЕ**

«Изучаем WORD:

Работа с документами»

Работу выполнил

Группа № \_\_\_\_

ФИО

Санкт-Петербург

2012

## 24. Стилиевое форматирование документов.

### Стилиевое форматирование документа

Осн\_текст

11пт  
перед - 6пт

Используя для форматирования стили, можно быстро создать профессионально оформленные документы и, при необходимости, быстро изменить их внешний вид. Если при форматировании текста пользоваться стилями, то создание оглавления, организация ссылок на части (разделы) документа и реорганизация документа становятся действительно быстро решаемыми задачами.

Заг\_вправо

12пт\_Ж

#### Ответы на вопросы

Что такое стиль?

При ручном форматировании мы последовательно шаг за шагом применяем к абзацу различные форматы: шрифт и его размер, выравнивание, обрамление и т.д. **Стиль** позволяет применить к абзацу все форматы за один шаг. **Стиль** - это команда форматирования, которую мы сами создали и дали этой команде уникальное имя.

Что такое стиль абзаца?

**Стиль абзаца** управляет внешним видом абзаца, т.е. форматами, которые влияют на весь абзац целиком.

Абз\_вправо

11пт  
перед - 6пт  
после - 6пт

Что такое стиль символа?

**Стиль символа** управляет внешним видом отдельных символов, слов, фраз и применяет любые форматы команды ФОРМАТ/ШРИФТ. Стиль символов используется для выделения слов или фраз в тексте.

Заг\_влево

12пт\_Ж

Что такое шаблон NORMAL.DOT?

При наборе текста новый документ форматируется стилем Нормальный - **стандартным** стилем для обычного документа. Стиль Нормальный является базовым стилем.

Что такое шаблон?

**Шаблон** это эскиз (план) документа, который может содержать заранее созданные стили, текст, графику, поля и др.

Абз\_влево

11пт  
перед - 6пт  
после - 6пт

Зачем нужны шаблоны?

Для **быстрого** форматирования документов одного типа, требующих одинакового внешнего представления, для документов, содержащих одинаковый текст и графику (справки, фирменные листы, письма) и т.п. используются шаблоны.



В 2009 году Университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена программа его развития на 2009–2018 годы. В 2011 году Университет получил наименование «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»

---

## **КАФЕДРА ИНФОРМАТИКИ И ПРИКЛАДНОЙ МАТЕМАТИКИ**

Кафедра образована в 1976 году из сотрудников кафедры Вычислительной техники для подготовки специалистов в области программирования и методов вычислений. Кафедру возглавляет д.т.н., профессор О. Ф. Немолочнов, работающий в области создания систем автоматизации проектирования ЭВМ.

Долгое время основным научным направлением кафедры было создание систем автоматизированного контроля цифровой аппаратуры. Кафедра активно сотрудничала с такими организациями, как НИЦЭВТ в Москве, НПО Электроавтоматика в Ленинграде, КБЭ в Харькове, завод САМ в Минске.

В последнее десятилетие на кафедре продолжают работы по исследованию методов построения контролирующих и диагностических тестов, по автоматизации проектно-конструкторских работ в оптике, по информационному обеспечению САПР.

Сотрудники кафедры разрабатывают новые учебные программы и циклы лабораторных работ, ориентированные на привитие студентам всех специальностей навыков практической работы на ЭВМ. На кафедре ведутся работы по внедрению методов дистанционного обучения, создаются учебники и учебно-методические комплексы по различным дисциплинам, связанным с разработкой программного обеспечения.

Денисова Эльвира Викторовна

## **Информатика. Базовый курс**

**Учебное пособие**

В авторской редакции

Редакционно-издательский отдел НИУ ИТМО

Зав. РИО

Н.Ф. Гусарова

Лицензия ИД № 00408 от 05.11.99

Подписано к печати

Заказ №

Тираж

Отпечатано на ризографе

## **Редакционно-издательский отдел**

Санкт-Петербургского национального  
исследовательского университета  
информационных технологий, механики  
и оптики

197101, Санкт-Петербург, Кронверкский пр., 49

