

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
УКРАИНЫ  
АВТОМОБИЛЬНО–ДОРОЖНЫЙ ИНСТИТУТ  
ГОСУДАРСТВЕННОГО ВЫСШЕГО УЧЕБНОГО  
ЗАВЕДЕНИЯ  
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ»



## **МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

для выполнения лабораторных работ по дисциплине  
«Информатика и компьютерная техника»

для студентов специальности 7.050102  
“Экономическая кибернетика”

Горловка 2007

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ  
АВТОМОБИЛЬНО–ДОРОЖНЫЙ ИНСТИТУТ  
ГОСУДАРСТВЕННОГО ВЫСШЕГО ЧЕБНОГО  
ЗАВЕДЕНИЯ  
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ»

## МЕТОДИЧЕСКИЕ УКАЗНИЯ

Для выполнения лабораторных работ по дисциплине  
«Информатика и компьютерная техника»  
(для студентов специальности 7.050102  
«Экономическая кибернетика»)

Утверждено  
на заседании методической  
комиссии факультета  
«Экономика и управление»  
Протокол № \_\_\_\_ от \_\_\_\_\_ 2007 г.

Утверждено  
на заседании кафедры  
«Информационные системы в  
экономике»  
Протокол № \_\_\_\_ от \_\_\_\_\_ 2007 г.

Горловка 2007

УДК 337 (071)

Методические указания для выполнения лабораторных работ по дисциплине «Информатика и компьютерная техника» (для студентов специальности 7.050102 «Экономическая кибернетика») / Сост.: Николаенко Д.В. – Горловка: АДИ ДонНТУ, 2007. – 59 с.

Составлена по программе дисциплины «Информатика и компьютерная техника».

Составил: Д.В. Николаенко, ассистент

Ответственный за выпуск: В. Л. Николаенко, к.т.н., доцент

Рецензент: Л.П. Вовк, д.т.н., доцент

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	5
ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ОТЧЕТА .....	6
УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ .....	7
ЛАБОРАТОРНАЯ РАБОТА 1 «ПРОГРАММЫ-АНТИВИРУСЫ».....	7
ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ .....	7
ВАРИАНТЫ ЗАДАНИЙ .....	12
ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ .....	13
КОНТРОЛЬНЫЕ ВОПРОСЫ .....	14
ЛАБОРАТОРНАЯ РАБОТА 2 «АРХИВАТОРЫ» .....	15
ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ .....	15
ВАРИАНТЫ ЗАДАНИЙ .....	16
ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ .....	16
КОНТРОЛЬНЫЕ ВОПРОСЫ .....	19
ЛАБОРАТОРНАЯ РАБОТА 3 «БАЗЫ ДАННЫХ» .....	20
ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ .....	20
ВАРИАНТЫ ЗАДАНИЙ .....	25
ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ .....	26
КОНТРОЛЬНЫЕ ВОПРОСЫ .....	28
ЛАБОРАТОРНАЯ РАБОТА 4 «HTML» .....	29
ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ .....	29
ВАРИАНТЫ ЗАДАНИЙ .....	35
ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ .....	35
КОНТРОЛЬНЫЕ ВОПРОСЫ .....	36
ЛАБОРАТОРНАЯ РАБОТА 5 «ЭЛЕКТРОННЫЕ ТАБЛИЦЫ EXCEL. VBA. ».....	37
ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ .....	37
ВАРИАНТЫ ЗАДАНИЙ .....	39
ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ .....	41
КОНТРОЛЬНЫЕ ВОПРОСЫ .....	44
СПИСОК ЛИТЕРАТУРЫ .....	45
ПРИЛОЖЕНИЕ А ТАБЛИЦА БАЗОВЫХ ЦВЕТОВ .....	47
ПРИЛОЖЕНИЕ Б СПЕЦИАЛЬНЫЕ СИМВОЛЫ.....	49
ПРИЛОЖЕНИЕ В ВАРИАНТЫ ЗАДАНИЙ .....	54

## **ВВЕДЕНИЕ**

## **ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ОТЧЕТА**

Отчет по лабораторной работе должен включать в себя следующие структурные элементы:

1. Название работы.
2. Цель.
3. Вариант. Условие.
4. Решение.

В разделе «РЕШЕНИЕ» необходимо привести теоретические сведения которыми студент пользовался при решении задания, «скриншоты» пошагового выполнения задания (для всех работ кроме работы HTML, в которой необходимо вставить исходный код html файлов).

Все работы оформляются в одном документе word с названием report\_labrabN\_Sidorov\_EK00x.doc.

Для работ касающихся HTML необходимо привести файлы с расширением htm (html) содержащие исходные коды web страниц.

# УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

## Лабораторная работа 1 «Программы-антивирусы»

### *Теоретические сведения*

С увеличением количества людей, пользующихся компьютером, и возможностей обмена между ними информацией по электронной почте и через интернет возросла угроза заражения компьютера, а также порчи или хищения информации вредоносными программами.

Чтобы знать, какого рода опасности могут угрожать данным, полезно узнать, какие бывают вредоносные программы и как они работают. В целом вредоносные программы можно разделить на следующие три класса:

**Черви (Worms)** - данная категория вредоносных программ для распространения использует уязвимости в сетевой защите. Название этого класса было дано исходя из способности червей "переползать" с компьютера на компьютер, используя сети, электронную почту и другие информационные каналы. Благодаря этому черви обладают исключительно высокой скоростью распространения.

Черви проникают на компьютер, вычисляют сетевые адреса других компьютеров и рассылают по этим адресам свои копии. Помимо сетевых адресов черви часто используют данные адресной книги почтовых клиентов для рассылки зараженных сообщений. Представители этого класса вредоносных программ иногда создают рабочие файлы на дисках системы, но могут вообще не обращаться к ресурсам компьютера (за исключением оперативной памяти).

Черви также являются своего рода подготовительным этапом для проникновения на компьютер других вредоносных программ. Так, червь может создать ряд уязвимостей, используя которые, на компьютер проникают троянские программы.

**Вирусы (Viruses)** - программы, которые заражают другие программы - добавляют в них свой код, чтобы при запуске зараженного файла получить возможность выполнения несанкционированных действий. Это простое определение дает возможность выявить основное действие, выполняемое вирусом - заражение. Скорость распространения вирусов несколько ниже, чем у червей.

**Троянские программы (Trojans)** - программы, которые выполняют на поражаемых компьютерах несанкционированные пользователем действия, т.е. в зависимости от каких-либо условий уничтожают информацию на дисках, приводят систему к "зависанию", воруют конфиденциальную информацию и т.д. Данный класс вредоносных программ не является

вирусом в традиционном понимании этого термина (т.е. не заражает другие программы или данные); троянские программы не способны самостоятельно проникать на компьютеры и распространяются злоумышленниками под видом "полезного" программного обеспечения. При этом вред, наносимый ими, может во много раз превышать потери от традиционной вирусной атаки.

В последнее время наиболее распространенными типами вредоносных программ, портящими компьютерные данные, стали черви. Далее по распространенности следуют вирусы и троянские программы. Некоторые вредоносные программы совмещают в себе характеристики двух или даже трех из перечисленных выше классов.

Также широкое распространение получили следующие потенциально опасные программы:

**Программы-рекламы (AdWare)** - программный код, без ведома пользователя включенный в программное обеспечение с целью демонстрации рекламных объявлений. Как правило, программы-рекламы встроены в программное обеспечение, распространяющееся бесплатно. Реклама располагается в рабочем интерфейсе. Зачастую данные программы собирают и переправляют своему разработчику персональную информацию о пользователе, изменяют различные параметры браузера (стартовые и поисковые страницы, уровни безопасности и т.д.), а также создают неконтролируемый пользователем трафик. Все это может привести как к нарушению политики безопасности, так и к прямым финансовым потерям.

**Потенциально опасные программы (RiskWare)** - программное обеспечение, которое не имеет какой-либо вредоносной функции, но может быть использовано злоумышленниками в качестве вспомогательных компонентов вредоносной программы, поскольку содержит бреши и ошибки. В эту категорию попадают, например, программы удаленного администрирования, IRC-клиенты, FTP-сервера, всевозможные утилиты для остановки процессов или скрытия их работы.

**Программы-шпионы (SpyWare)** - программное обеспечение, целью которого является несанкционированный доступ к данным пользователя, отслеживание действий на компьютере, сбор информации о содержании жесткого диска. Они позволяют злоумышленнику не только собирать информацию, но и контролировать чужой компьютер. Программы-шпионы, как правило, распространяются вместе с бесплатным программным обеспечением и устанавливаются на компьютер незаметно для пользователя. К таковым относятся клавиатурные шпионы, программы взлома паролей, программы сбора конфиденциальной информации (например, номеров кредитных карт).



**Программы автодозвона (PornWare)** - программы, которые осуществляют модемное соединение с различными платными интернет-ресурсами, как правило, порнографического содержания.

**Хакерские утилиты (Hack Tools)** - программное обеспечение, которое используется злоумышленниками в собственных целях для проникновения на ваш компьютер. К ним относятся различные нелегальные сканеры уязвимостей, программы для взлома паролей, прочие виды программ для взлома сетевых ресурсов или проникновения в атакуемую систему.

Основными источниками распространения вредоносных программ является электронная почта и интернет, хотя заражение может также произойти через дискету или CD-диск. Это обстоятельство предопределяет смещение акцентов антивирусной защиты с простых регулярных проверок компьютера на присутствие вирусов на более сложную функцию постоянной защиты компьютера от возможного заражения.

### **Профилактика заражения компьютера**

Непрофессионалу сложно обнаружить присутствие вирусов на компьютере, поскольку они умело маскируют свое присутствие среди обычных файлов.

### **Признаки заражения**

Есть ряд признаков, предположительно свидетельствующих о заражении компьютера. Если с компьютером происходят "странные" вещи, например:

- вывод на экран непредусмотренных сообщений или изображений;
- подача непредусмотренных звуковых сигналов;
- неожиданное открытие и закрытие лотка CD-ROM-устройства;
- произвольный запуск на компьютере каких-либо программ;
- при наличии установленной на вашем компьютере антивирусной программы вывод на экран предупреждения о попытке какой-либо из программ выйти в интернет, хотя вы этого не инициировали, то, с большой степенью вероятности, можно предположить, что ваш компьютер поражен вирусом.

Кроме того, есть некоторые характерные признаки поражения вирусом через почту:

- друзья или знакомые говорят вам о получении сообщений от вас, которые вы не отправляли;
- почтовом ящике находится большое количество сообщений без обратного адреса и заголовка.

Следует отметить, что не всегда такие признаки вызываются присутствием вирусов. Иногда они могут быть следствием других причин. Например, в случае с почтой зараженные сообщения могут рассылаться с вашим обратным адресом, но не с вашего компьютера.

Есть также косвенные признаки заражения компьютера:

- частые зависания и сбои в работе компьютера;
- медленная работа компьютера при запуске программ;
- невозможность загрузки операционной системы;
- исчезновение файлов и каталогов или искажение их содержимого;
- частое обращение к жесткому диску (часто мигает лампочка на системном блоке);

Microsoft Internet Explorer "зависает" или ведет себя неожиданным образом (например, окно программы невозможно закрыть).

В большинстве случаев наличие косвенных симптомов вызвано сбоем в аппаратном или программном обеспечении. Несмотря на то, что подобные симптомы с малой вероятностью свидетельствуют о заражении, при их появлении лучше провести полную проверку компьютера.

#### **Действия при наличии признаков заражения**

- Отключите компьютер от интернета.
- Отключите компьютер от локальной сети, если он к ней был подключен.
- Если симптом заражения состоит в том, что не возможно загрузиться с жесткого диска компьютера (компьютер выдает ошибку, при включении), попробуйте загрузиться в режиме защиты от сбоев или с диска аварийной загрузки Windows, который вы создавали при установке операционной системы на компьютер.

#### **Классификация средств защиты**

Антивирусные средства можно разделить на классы:

- детекторы
- фаги
- ревизоры
- вакцины
- сторожа
- другие программы

Детекторы - это программы, проверяющие есть ли заражение известными им вирусами.

Фаги - это программы "вырезающие" известные им вирусы, то есть программы, которые "лечат" другие программы.

Ревизоры - это программы, определяющие попытку внести изменения в указанные программы.

Сторожа - это программы, выявляющие попытки выполнить на компьютере действия чреватые проникновением вирусов.

Вакцины - это программы, делающие указанные программы невосприимчивыми к указанным вирусам.

## **Правила безопасности**

Никакие самые надежные и разумные меры не смогут обеспечить стопроцентную защиту от компьютерных вирусов и троянских программ, но, выработав ряд правил, можно существенно снизить вероятность вирусной атаки и степень возможного ущерба.

Одним из основных методов борьбы с вирусами является, как и в медицине, своевременная профилактика. Компьютерная профилактика состоит из небольшого количества правил, соблюдение которых значительно снижает вероятность заражения вирусом и потери каких-либо данных.

Ниже перечислены основные правила безопасности, выполнение которых позволит избежать вирусных атак.

**Правило № 1:** защитите компьютер с помощью антивирусной программы и программ безопасной работы в интернете. Для этого:

- Безотлагательно установите Антивирус.
- Регулярно обновляйте антивирусные базы. Обновление можно проводить несколько раз в день при возникновении вирусных эпидемий - в таких ситуациях антивирусные базы на серверах обновлений обновляются немедленно.

- Задайте рекомендуемые настройки для постоянной защиты. Постоянная защита вступает в силу сразу после включения компьютера и затрудняет вирусам проникновение на компьютер.

- Задайте рекомендуемые для полной проверки компьютера и запланируйте ее выполнение не реже одного раза в неделю.

- Рекомендуется также установить программу для защиты компьютера при работе в интернете.

**Правило № 2:** будьте осторожны при записи новых данных на компьютер:

- Проверяйте на присутствие вирусов все съемные диски (дискеты, CD-диски, флэш-карты и пр.) перед их использованием.

- Осторожно обращайтесь с почтовыми сообщениями. Не запускайте никаких файлов, пришедших по почте, если не уверены, что они действительно должны были прийти к вам, даже если они отправлены вашими знакомыми. В особенности не доверяйте письмам якобы от антивирусных производителей.

- Внимательно относитесь к информации, получаемой из интернета. Если с какого-либо веб-сайта вам предлагается установить новую программу, обратите внимание на наличие у нее сертификата безопасности.

- Если вы копируете из интернета или локальной сети исполняемый файл, обязательно проверьте его Антивирусом.

- Внимательно относитесь к выбору посещаемых вами интернет-сайтов. Некоторые из сайтов заражены опасными скрипт-вирусами или интернет-червями.

**Правило № 3:** с недоверием относитесь к вирусным мистификациям - "страшилкам", письмам об угрозах заражения.

**Правило № 4:** уменьшите риск неприятных последствий возможного заражения:

- Своевременно делайте резервное копирование данных. В случае потери данных система достаточно быстро может быть восстановлена при наличии резервных копий. Дистрибутивные диски, дискеты, флэш-карты и другие носители с программным обеспечением и ценной информацией должны храниться в надежном месте.

- Обязательно создайте системную аварийную дискету, с которой при необходимости можно будет загрузиться, используя "чистую" операционную систему.

### ***Варианты заданий***

1. Проверить при помощи программы антивирус диск C на наличие вирусов.
2. Проверить при помощи программы антивирус локальный каталог E:\STUDENT на наличие вирусов.
3. Проверить при помощи программы антивирус сетевой каталог E:\STUDENT (любого соседнего компьютера) на наличие вирусов.
4. Обновить антивирусные базы через Интернет
5. Обновить антивирусные базы с локального диска
6. Проверить файлы диска E:\, удовлетворяющие маске \*.exe на наличие вирусов
7. Проверить файлы диска E:\, удовлетворяющие маске \*.cmd на наличие вирусов
8. Проверить файлы диска E:\, удовлетворяющие маске \*.doc на наличие вирусов
9. Проверить файлы диска E:\, удовлетворяющие маске \*.htm на наличие вирусов
10. Проверить оперативную память ПК на наличие вирусов.
11. Проверить файлы сетевого диска (любого соседнего ПК), удовлетворяющие маске \*.exe на наличие вирусов
12. Проверить файлы сетевого диска (любого соседнего ПК), удовлетворяющие маске \*.cmd на наличие вирусов
13. Проверить файлы сетевого диска (любого соседнего ПК), удовлетворяющие маске \*.doc на наличие вирусов

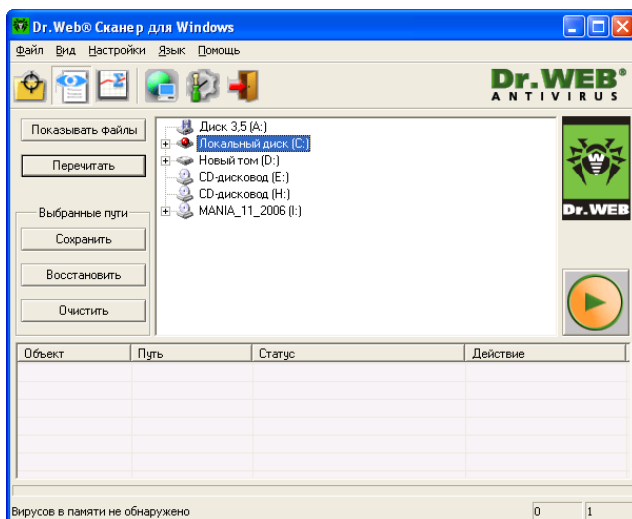
14. Проверить файлы сетевого диска (любого соседнего ПК), удовлетворяющие маске \*.htm на наличие вирусов
15. Настроить планировщик антивируса на проверку вирусов каждые 5 минут
16. Настроить планировщик антивируса на обновление антивирусных баз ежедневно.
17. Провести настройку постоянной защиты. Выбрать уровень защиты.
18. Провести настройку программы антивирус. Настроить реакцию при обнаружении вируса.
19. Выполнить полную проверку компьютера.
20. Сформировать список исключений.

### ***Пример выполнения работы***

Задание: проверить при помощи программы антивирус диск С на наличие вирусов с использованием антивируса Dr.Web.

Решение: в иерархическом списке в центральной части главного окна сканера выберите объект, подлежащий проверке.

Для выбора щелкните по соответствующему объекту левой кнопкой мыши. При этом выбранный объект будет выделен красной точкой, как показано на рисунке



Для того чтобы приступить к сканированию выбранных объектов, нажмите на кнопку старт, которая находится в правой части главного окна.

По умолчанию наряду с выбранными объектами проверяются подкаталоги всех выбранных каталогов и логических дисков и загрузочные сектора всех логических дисков, на которых выбран хотя бы один каталог или файл, а также главные загрузочные сектора соответствующих физических дисков.

### ***Контрольные вопросы***

1. Назовите классы вирусов.
2. Что является профилактикой заражения компьютера.
3. Перечислите признаки заражения компьютера.
4. Перечислите действия при наличии признаков заражения
5. Назовите правила безопасности
6. Проведите классификацию программ антивирусов

## **Лабораторная работа 2 «Архиваторы»**

### ***Теоретические сведения***

Благодаря уплотнению данных можно сократить время, а значит, и уменьшить стоимость передачи данных по каналам связи, на 10 - 80 %.

Хранение программных изделий в упакованном виде требует меньших объемов памяти внешних носителей, следовательно, позволяет уменьшить затраты на ведение архивов информации.

Облегчается процесс копирования, переноса и инсталляции программных систем, если они предварительно упакованы.

Архиваторы-деархиваторы - это программные системы, назначением которых является уплотнение данных перед их сохранением и разуплотнение при их возвращении из архивов.

Архив - это набор из одного или более файлов, размещенных в одном архивном файле.

Для каждого архивируемого файла в архиве запоминается обычно следующая информация:

- объем файла до сжатия;
- объем файла после сжатия;
- время и дата архивации;
- 16-ти битовая строка циклической добавочной проверки для обеспечения целостности данных при их восстановлении (CRC сумма);
- метод сжатия.

Самораспаковывающийся (SFX, от англ. Self-eXtracting) архив — это архив, к которому присоединен исполняемый модуль. Этот модуль позволяет извлекать файлы простым запуском архива как обычной программы. Таким образом, для извлечения содержимого SFX-архива не требуется дополнительных внешних программ.

Многотомный архив - архивный файл, физически располагающийся на нескольких магнитных носителях.

Тома — это фрагменты архива, состоящего из нескольких частей.

Уровень сжатия - процентное значение, указывающее долю упакованного файла по сравнению со своим оригиналом.

Непрерывный архив — это архив RAR, упакованный специальным способом, при котором все сжимаемые файлы рассматриваются как один последовательный поток данных.

### **Классификация программ сжатия данных**

Уплотнение данных сейчас используется в таких случаях:

- резервное копирование
- динамическое сжатие при чтении-записи на диски

- сжатие исполнимых модулей
- сжатие отдельных файлов и каталогов
- динамическое сжатие при чтении-записи в RAM

Утилиты резервного копирования - это программные системы, предназначенные для образования резервных копий логических дисков.

### ***Варианты заданий***

1. Создать обычный архив.
2. Создать многотомный архив.
3. Создать самораспаковывающийся архив.
4. Создать многотомный самораспаковывающийся архив.
5. Создать непрерывный архив.
6. Обновить непрерывный архив (добавить несколько файлов).
7. Извлечь один файл из непрерывного архива.
8. Создать обычный архив с минимальной степенью сжатия.
9. Создать архив с максимальной степенью сжатия.
10. Создать самораспаковывающийся архив с максимальной степенью сжатия.
11. Создать архив с томом для восстановления.
12. Создать обычный архив с добавлением шифрования.
13. Создать многотомный архив с добавлением шифрования.
14. Создать самораспаковывающийся архив с добавлением шифрования.
15. Добавить комментарий в обычный архив.
16. Добавить комментарий в самораспаковывающийся архив.
17. Создать обычный многотомный архив с добавлением информации для восстановления.
18. Создать самораспаковывающийся архив с добавлением информации для восстановления.
19. Создать установочный пакет на основе архива с возможностью выбора пути установки, запуска программы по окончании установки, создания ярлыка на рабочем столе.
20. Извлечь все файлы из многотомного архива.

### ***Пример выполнения работы***

Задание: Создать обычный архив.

Решение: (Приведенное ниже решение реализовано с использованием графической оболочки WinRAR)

Прежде всего, необходимо запустить оболочку WinRAR. Дважды щёлкните мышью или нажмите <Enter> на значке WinRAR. Можно запустить

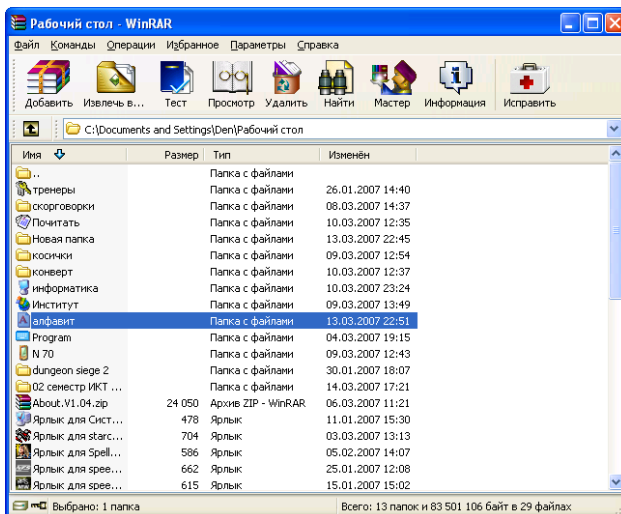


его из меню "Пуск": зайти в группу "Программы", потом в группу "WinRAR" и выбрать пункт "WinRAR".

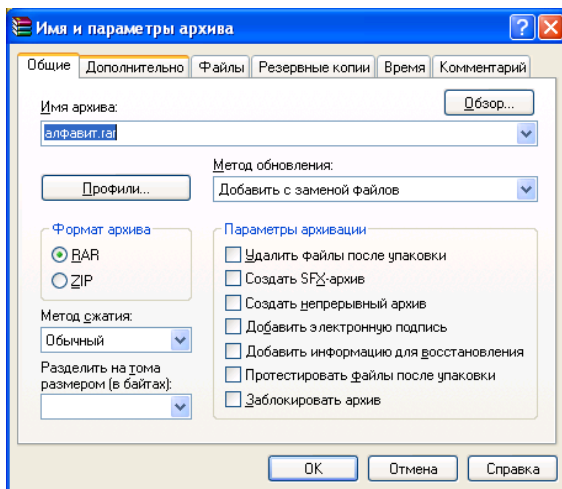
При запуске WinRAR показывает в своем окне список файлов и папок в текущей папке. Перейдите в папку, в которой находятся файлы, предназначенные для архивации. Для изменения текущего диска можно нажать комбинацию клавиш <Ctrl+D>, воспользоваться списком дисков (под панелью инструментов), или щелкнуть на маленьком значке диска в нижнем левом углу окна. Для перехода в родительскую папку нажмите клавиши <Backspace> или <Ctrl+PgUp>, щёлкните на маленькой кнопке "Вверх" (под панелью инструментов) или дважды щёлкните на папке ".." в списке файлов. Для перехода в другую папку нажмите <Enter>, <Ctrl+PgDn> или дважды щёлкните мышью на этой папке. Для перехода в корневую папку текущего диска нажмите сочетание клавиш <Ctrl+|>.

После того как вы вошли в папку с нужными файлами, выделите файлы и папки, которые хотите заархивировать. Это можно сделать клавишами управления курсором или левой кнопкой мыши при нажатой клавише <Shift> (как в Проводнике и других программах Windows). Выделять файлы в окне WinRAR можно также клавишами <Пробел> и <Insert>. Клавиши <+> и <-> на цифровой клавиатуре позволяют выделять и снимать выделение с группы файлов с помощью шаблонов (т.е. задавая маски файлов символами '\*' и '?').

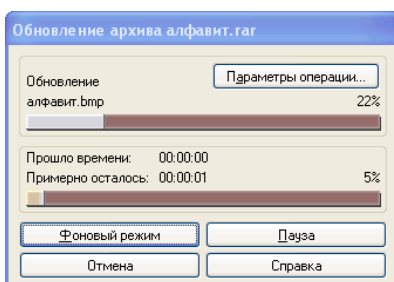
Выделив один или несколько файлов, нажмите кнопку "Добавить" (то же действие происходит при нажатии <Alt+A>).



В появившемся диалоговом окне введите имя архива или просто подтвердите имя, предложенное по умолчанию. Здесь же можно выбрать формат нового архива (RAR или ZIP), метод сжатия, размер тома и прочие параметры архивации. Нажмите кнопку "OK", чтобы создать архив.



Во время архивации отображается окно со статистикой. Если вы хотите прервать процесс упаковки, нажмите кнопку "Отмена". Окно WinRAR можно минимизировать в область уведомлений панели задач: для этого нажмите кнопку "Фоновый режим".



По окончании архивации окно статистики исчезнет, а созданный архив станет текущим выделенным файлом.

### ***Контрольные вопросы***

1. Что такое архив?
2. Что такое SFX-архив?
3. Для чего нужна архивация данных?
4. Какой формат данных лучше сжимается, а какой хуже. Почему?
5. Какая информация обычно запоминается в архиве для каждого файла?
6. Что такое многотомный архив?
7. Что такое том?
8. Поясните понятие «непрерывный архив»
9. Проведите классификацию программ сжатия данных.
10. Что такое утилиты резервного копирования. Область применения.

### **Лабораторная работа 3 «Базы данных»**

#### ***Теоретические сведения***

База данных БД – электронные хранилища информации, доступ к которым осуществляется с помощью одного или нескольких компьютеров. Обычно БД создаются для хранения и доступа к данным, содержащим сведения о некоторой предметной области, то есть некоторой области деятельности или области реального мира.

Системы управления базами данных (СУБД) – это программные средства, предназначенные для создания, наполнения, обновления и удаления баз данных. Различают три основных вида СУБД: промышленные универсального назначения, промышленные специального назначения и разрабатываемые для конкретного заказчика. Специализированные СУБД создаются для управления базами данных конкретного назначения – бухгалтерские, складские, банковские и т.д. Универсальные СУБД не имеют четко выраженных рамок применения, они рассчитаны на все случаи жизни и, как следствие, достаточно сложны и требуют от пользователя специальных знаний. Как специализированные, так и универсальные промышленные СУБД относительно дешевы, достаточно надежны (отлажены) и готовы к немедленной работе, в то время как заказные СУБД в максимальной степени учитывают специфику работы заказчика, их интерфейс обычно интуитивно понятен пользователям и не требует от них специальных знаний.

Пользователь БД - это программа или человек, обращающиеся к БД на языке манипулирования данными.

Запрос - это процесс обращения пользователя к БД с целью ввода, получения или изменения информации в БД.

Транзакция - это последовательность операций модификации данных в БД, переводящая БД из одного непротиворечивого состояния в другое непротиворечивое состояние.

Логическая структура БД - это определение БД на физически независимом уровне, ближе всего соответствующем концептуальной модели БД.

Топология БД (структура РБД) - это схема распределения физических БД по сети. Локальная автономность означает принадлежность локальному владельцу информации локальной БД и связанных с ней определенных данных.

Удаленный запрос - это запрос, который выполняется с использованием модемной связи.

Возможность реализации удаленной транзакции - это обработка одной транзакции, состоящей из множества SQL-запросов, на одном удаленном узле.

Поддержка распределенной транзакции допускает обработку транзакции, состоящей из нескольких SQL-запросов, которые выполняются на нескольких узлах сети (удаленных или локальных), но каждый запрос в этом случае обрабатывается только на одном узле, т.е. запросы не являются распределенными. При обработке одной распределенной транзакции разные локальные запросы могут обрабатываться в разных узлах сети.

В зависимости от расположения отдельных частей СУБД различают локальные и сетевые СУБД. Все части локальной СУБД расположены на компьютере пользователя. Если к одной БД обращаются несколько пользователей одновременно, то каждый пользовательский ПК должен иметь свою локальную копию БД. Недостатки очевидны (это слово специально для студентки ЭК-06) – проблема синхронизации копий данных.

К сетевым относят файл-серверные, клиент-серверные и распределенные СУБД. Непременным атрибутом таких систем является сеть, обеспечивающая аппаратную связь компьютеров и делающая возможной корпоративную работу множества пользователей с одними и теми же данными.

В файл-серверных СУБД все данные расположены в одном или нескольких каталогах достаточно мощной машины, специально выделенной для этих целей и постоянно подключенной к сети. Такой компьютер называется файл-сервером. Безусловным достоинством является относительная простота ее создания и обслуживания – фактически все сводится лишь к созданию сети, установки на рабочих станциях ПО для работы в сети и с БД. Локальные и файл-серверные СУБД имеют мало различий, т.к. в обоих случаях вся информация хранится на одном ПК. Недостатком таких систем является большая нагрузка на локальную сеть. Файл-серверные СУБД могут применяться на небольших предприятиях для обработки незначительного объема информации.

Модель "клиент-сервер" связана с принципом открытых систем. Термин "клиент-сервер" исходно применялся в архитектуре ПО, которое ориентировало распределение процесса выполнения по принципу взаимодействия 2-х программ, процессов, один из которых в этой модели назывался клиентом, а другой - сервером. При этом предполагалось, что один серверный процесс может обслуживать множество клиентских процессов.

Основной принцип технологии "клиент-сервер" в БД заключается в разделении функций стандартного интерактивного приложения на 5 групп:

Функция ввода и отображения данных (PL);

- Прикладные функции, определяющие основные алгоритмы решения задач приложения (BL);
- Функции обработки данных внутри приложения (DL);
- Функции управления информационными ресурсами (DML);
- Служебные функции, играющие роль связок между функциями 1-х и 4-х групп.

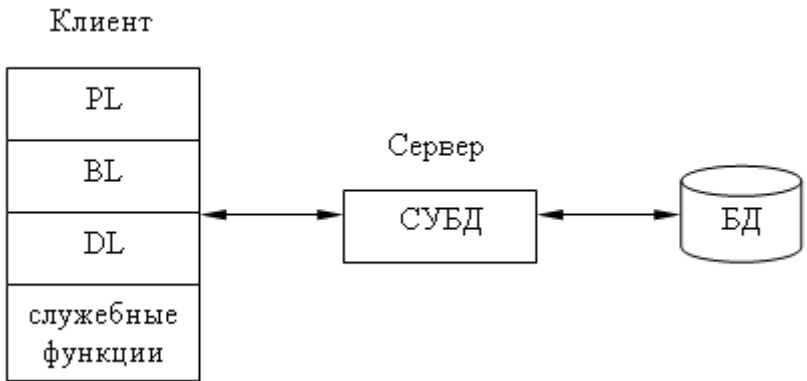


Рисунок 1. - Структура типичного приложения, работающего с БД.

PL - это часть приложения, которая определяется тем, что пользователь видит на экране, когда работает приложение (интерактивные экранные формы, а также все то, что выводится пользователю на экран, результаты решения некоторых промежуточных задач, справочная информация).

BL - это часть кода приложения, которая определяет алгоритмы решения конкретных задач приложения. Обычно этот код пишется с использованием различных языков программирования.

DL - это часть кода приложения, которая связана с обработкой данных внутри приложения (данными управляет собственно СУБД), где используется язык запросов и средства манипулирования данными стандартного языка SQL.

Клиент-серверные БД значительно снижают нагрузку на сеть, т.к. клиент общается с данными через специального посредника – сервер базы данных, который размещается на машине с данными. Сервер принимает запрос, выполняет его и отправляет результаты пользователю (клиенту). Таким образом передается лишь запрос и ответ на него, а не все данные из БД. Запрос формируется на основе специального языка SQL

Распределенные СУБД могут содержать несколько десятков и сотен серверов БД. Количество клиентских мест в них может достигать десятков и

сотен тысяч. Обычно такие СУБД работают на предприятиях государственного масштаба, отдельные подразделения которых разнесены на значительной территории. К таким СУБД относят ГАИ МВД СБУ. Некоторые сервера могут дублировать друг друга для достижения предельно малой вероятности отказа от сбоев.

### **Уровни архитектуры БД**

Физический – включает все связанное с физическим хранением данных. Конечных пользователей не интересует.

Концептуальный – представляет абстрактное отображение данных.

### **Таблицы БД и связи между ними**

Единицей хранящейся в БД есть таблица. Каждая таблица представляет собой совокупность строк и столбцов, где строки соответствуют экземпляру объекта, конкретному событию или явлению, а столбцы – атрибутам (признакам, характеристикам, параметрам) этого объекта, события или явления.

Столбцы – поля

Строки – записи

Между отдельными таблицами БД могут существовать связи. Например, информация о покупателе в предыдущей таблице может дополняться другой.

Базы данных, между отдельными таблицами которых существуют связи, называются реляционными (relation – связь, отношение).

Связанные отношениями таблицы взаимодействуют по принципу главная (master) – подчиненная (detail)/

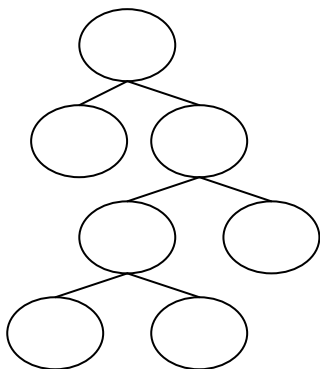


Рисунок 2. - Иерархическая модель (бинарное дерево)

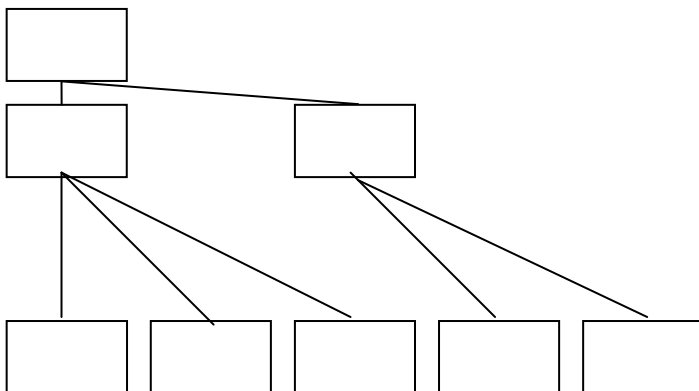


Рисунок 3. - Сетевая модель (не путать с бинарным деревом)

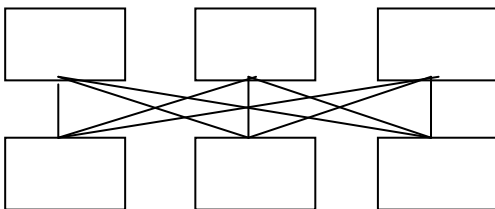


Рисунок 4. - Реляционная модель

Построить базу данных значит задать структуру таблицы данных, т.е.:

- а) дать имя полям (столбцам таблицы)
- б) задать тип поля
- в) задать длину поля
- г) задать свойства поля
- д) определить как будут вводиться данные

Форма ввода данных используется с целью обеспечения корректности ввода данных.

Сортировка – используется для настройки отображения данных для дальнейшего анализа и принятия решения.

Фильтрация – почти сортировка. При фильтрации пользователь может отображать данные, удовлетворяющие заданному условию. Аналог фильтрации можно наблюдать при поиске файлов с использованием маски.



Запрос – используется при сортировке и фильтрации, однако отображает данные, удовлетворяющие сложным условиям для формирования которых используется специальный язык SQL (Structural Query Language).

Отчет – используется для отображения данных в удобной форме. Как правило, применяется при заполнении некоторых форм.

Перечисленные структурные элементы форма – ... – отчет создаются тремя способами:

1. авто – нажал кнопку – получил результат
2. мастер – на каждом шаге создания можно выбрать несколько решений
3. конструктор – все создается вручную. Процесс трудоемкий, но результат, как правило, лучше, чем при использовании предыдущих двух способов

### **Первичные ключи и индексы**

В каждой таблице БД может существовать первичный ключ – поле или набор полей, однозначно идентифицирующий запись. Значение первичного ключа в таблице БД должно быть уникальным, то есть в таблице не должно существовать двух или более записей с одинаковыми значениями первичного ключа.

Первичный ключ облегчает установление связи между таблицами. Поскольку первичный ключ должен быть уникальным, то для него могут использоваться не все поля таблицы. Если в таблице таких уникальных полей нет, то можно ввести дополнительное поле. Как правило, это порядковый индекс.

Индекс отличается от первичных ключей тем, что не требует уникальности. Они устанавливаются по полям, которые часто используются при поиске и сортировке. Индексы помогут системе значительно быстрее найти нужные данные или отсортировать их в нужной последовательности.

### ***Варианты заданий***

1. Создать базу данных, используя СУБД MS Access с произвольным заполнением удовлетворяющих следующим условиям:
  - Количество полей не менее 5.
  - Поля должны быть как минимум 2 типов (строки, числа).
  - Количество записей не менее 20.
  - Наличие первичного ключа.
  - Наличие связанной таблицы.
2. Сформировать форму ввода используя мастер.
3. Создать запрос используя конструктор.
4. Провести фильтрацию данных в таблице.
5. Выполнить сортировку.

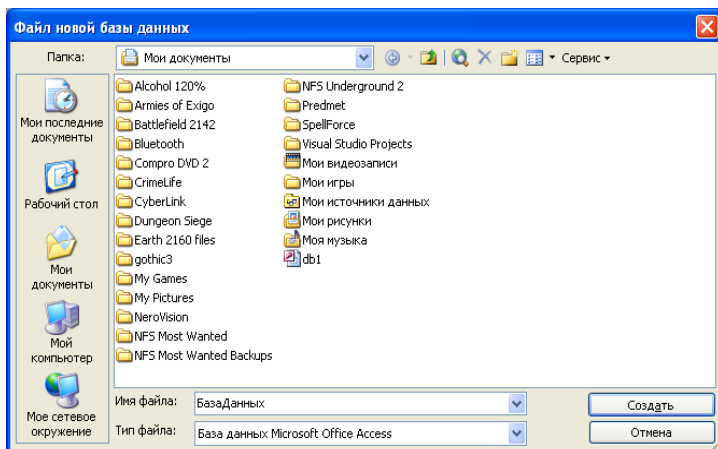
6. Сформировать отчет на основе запроса.

Критерии запроса и фильтрации, направления сортировки выдает преподаватель в зависимости от выбранных студентом данных.

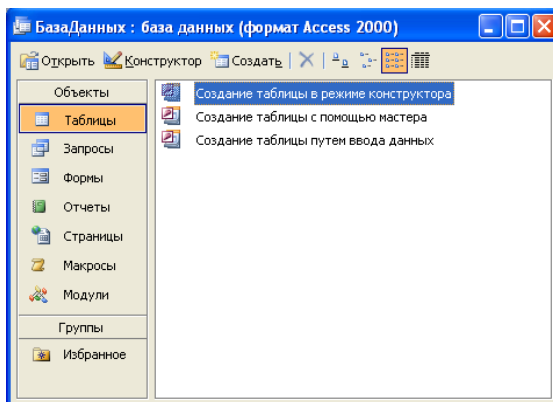
### ***Пример выполнения работы***

Для того чтобы создать базу данных запустите Microsoft Office Access и в меню Файл выберите Создать....

Далее выберите пункт Новая база данных.... В появившемся окне введите имя новой базы данных и нажмите Создать.



В появившемся окне выберите способ создания таблицы.



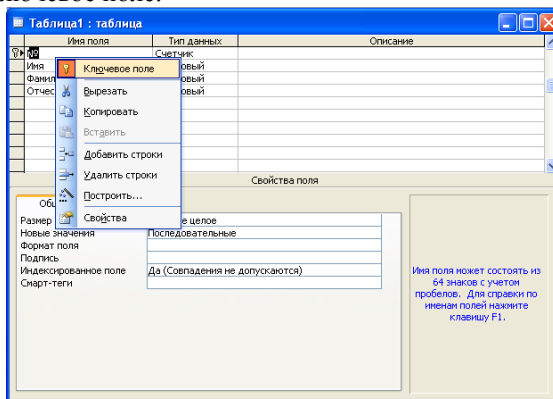
## Создание таблицы в режиме конструктора.

Окно конструктора таблиц разделяется на две области. В верхней области отображается сетка, каждая строка которой описывает один столбец базы данных. Для каждого столбца отображаются его основные характеристики: имя столбца, тип данных, длина и параметр допустимости пустых значений.

В нижней области окна конструктора таблиц отображаются дополнительные характеристики каждого столбца, выделенного в верхней области.

Из окна конструктора таблиц можно также получить доступ к окнам свойств, с помощью которых создаются и изменяются межтабличные связи, ограничения, индексы и ключи таблицы.

После заполнения необходимо указать ключевое поле. Для этого правой клавишей мыши нажмите на необходимом поле и в появившемся меню выберите Ключевое поле.



Следующие сведения полезны при создании первичного ключа.

Все столбцы, участвующие в первичном ключе таблицы должны содержать уникальные значения, отличные от значения Null. Чтобы добавить в столбец первичный ключ, снимите флажок Разрешить Null в представлении Свойства столбца.

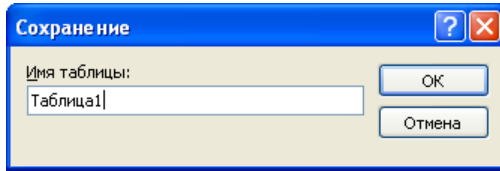
Первичный ключ должен содержать по меньшей мере один столбец и уникальные значения данных.

Нельзя создать первичный ключ, используя столбцы со следующими типами данных: text, ntext, image и bit.

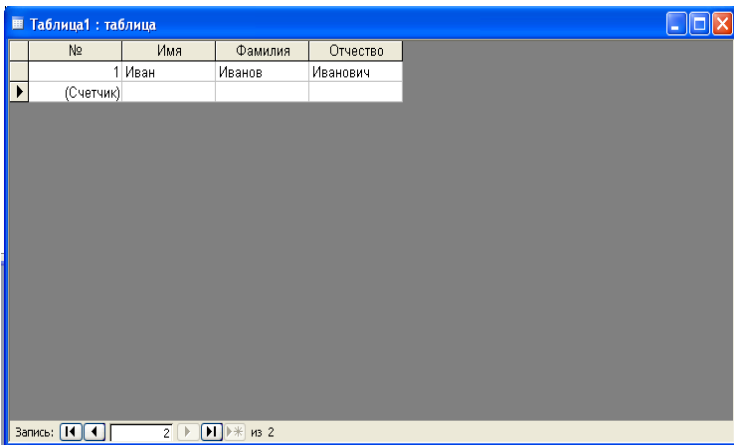
Для создания первичного ключа можно использовать не более 16 столбцов.

Сумма размеров всех столбцов в первичном ключе не может превышать 900 байт.

После заполнения необходимо сохранить таблицу. (Меню Файл, пункт сохранить) В появившемся окне укажите имя таблицы.



Перейдите в режим таблицы (Меню Вид, пункт режим таблицы), и вводите данные.



### ***Контрольные вопросы***

1. Дать определение понятию «База данных»
2. Что такое «системы управления базами данных» (СУБД)?
3. Архитектура СУБД.
4. Что такое локальные и сетевые СУБД?
5. Таблицы БД и связи между ними.
6. Что такое реляционные БД?
7. Что такое первичный ключ?
8. Что такое индексы?
9. Язык построения запросов SQL.
10. Какие типы связей возможны между таблицами БД?

## Лабораторная работа 4 «HTML»

### *Теоретические сведения*

World Wide Web (Web) - это сеть информационных ресурсов. Для того, чтобы сделать эти ресурсы доступными наиболее широкой аудитории, в Web используются три механизма:

Единая схема наименования для поиска ресурсов в Web (например, URI).

Протоколы для доступа к именованным ресурсам через Web (например, HTTP).

Гипертекст для простого перемещения по ресурсам (например, HTML).

Каждый ресурс в Web - документ HTML, изображение, видеоклип, программа и т.д. - имеет адрес, который может быть закодирован с помощью универсального идентификатора ресурсов (Universal Resource Identifier), или URI.

URI обычно состоят из трех частей:

- Схема наименования механизма, используемого для доступа к ресурсу.

- Имя машины, на которой располагается ресурс.

- Имя собственно ресурса, заданное в виде пути.

Например URL:

<http://www.w3.org/TR/PR-html4/cover.html>

Этот URL может читаться следующим образом: этот документ можно получить по протоколу HTTP, он располагается на машине [www.w3.org](http://www.w3.org), путь к этому документу - `"/TR/PR-html4/cover.html"`. Кроме того, в документах в формате HTML можно увидеть схемы `"mailto"` для электронной почты и `"ftp"` для протокола FTP.

Еще один пример URL. Он относится к почтовому ящику пользователя:

Комментарии отправляйте `<A href="mailto:joe@mail.ru">Джо</A>`.

Чтобы представить информацию для глобального использования, нужен универсальный язык, который понимали бы все компьютеры. Языком публикации, используемым в World Wide Web, является HTML (HyperText Markup Language - язык разметки гипертекстов).

HTML дает авторам средства для:

1. публикации электронных документов с заголовками, текстом, таблицами, списками, фотографиями и т.д.
2. загрузки электронной информации с помощью щелчка мыши на гипертекстовой ссылке.

3. разработки форм для выполнения транзакций с удаленными службами, для использования в поиске информации, резервировании, заказе продуктов и т.д.
4. включения электронных таблиц, видеоклипов, звуковых фрагментов и других приложений непосредственно в документы.

HTML-документ начинается тэгом <HEAD> и заканчивается закрывающим тэгом </HEAD>. Состоит документ из двух основных частей – "голова" и "тело" (<HEAD> и <BODY> соответственно)

```
<html>
<head>
  <title>Мой первый шаг </title>
</head>
<body>
  Это моя первая страница.<br>
  Добро пожаловать! :)
</body>
</html>
```

Для форматирования текста предусмотрена целая группа тэгов, которые можно разделить на логические и физические.

Тэги логического форматирования своими именами обозначают структурные типы своих текстовых фрагментов (такие как программный код <CODE>, цитата <CITE>, аббревиатура <ABBR> и т.д.)

Такие структуры отображают данные в закрытом формате по умолчанию.

И каждого пользователя они могут быть отличны друг от друга и могут быть легко переопределены.

Тэги физического уровня определяют формат отображения указанного в них фрагмента текста согласно предпочтения самого автора документа, т.е. как указал создатель WEB-страницы так будут видеть ее все пользователи.

К тэгам физического форматирования относят:

- <B> - жирный
- <I> - курсив
- <TT> - машинописный шрифт
- <U> - подчеркнутый
- <BIG> - больший шрифт, чем основной
- <SMALL> - меньший
- <SUB> - текст ниже уровня строки (K<sub>i</sub>)
- <SUP> - выше (X<sup>2</sup>)
- <BLINK> - мигающий
- <FONT> - указывает шрифт
- <FONT FACE="Verdana", "Arial", "Helvetica">

<BASE FONT> - размер, тип и цвет по умолчанию

<BASE FONT SIZE = 4>

**Гипертекстовый документ** – это документ, содержащий ссылки на другие документы и при помощи нажатия мыши между ними можно легко и быстро перемещаться.

Организация ссылок. Ссылка состоит из двух частей. Первая – видимая на экране и она называется указатель ссылки (anchor). Вторая часть – дающая инструкцию браузеру и называется адресной частью ссылки (URL - адрес)

Пример:

<A HREF="index.HTM">это ссылка</A>

но в качестве ссылки можно использовать и картинку, например,

<A HREF="index.htm"> <IMG SRC="pic.gif"> </A>

Правила записи ссылок. Для организации ссылок используют тэг-контейнер <A>, в котором в качестве значения указывают URL-адрес.

Для создания ссылок внутри документа тоже используют тэг-контейнер <A>, однако вначале необходимо создать указатель, определяющий местоположение. Это осуществляется следующим образом

<A NAME=top\_1> </A>

Тут нет ни содержимого, ни HREF, а затем в любом месте можно сделать ссылку на закладку top\_1

<A HREF="# top\_1">глава 1</A>

Ссылка на почтовый адрес осуществляется таким образом:

<A HREF="mailto:ivan\_petrov@mail.ru">пишите письма</a>

**Списки.** В HTML предусмотрена возможность создания списков, которые могут быть использованы в качестве:

- объединения информации в единую структуру для удобочитаемости;
- описание пошаговых процессов;
- расположение информации в стиле оглавления

Маркированные списки. Иначе этот тип списка называют нумерованным. (UL-Unordered List, неупорядоченный список). Для выделения элементов используют спец. символы (маркеры).

Тэги <UL> и <LI>

Для создания маркированного списка используется тэг-контейнер <UL> </UL>, внутри которого располагается все элементы списка. Каждый элемент списка начинается тэгом <LI> (LI – List Item – элемент списка) он не нуждается в закрывающем тэге.

Пример:

<UL>

<B> Знаки зодиака </B>

<LI> Овен

```
<LI> Телец  
<LI> Близнецы  
<LI> Рак
```

.....

```
<LI>  
</UL>
```

Тэг <UL> имеет 2 параметра: COMPACT и TYPE

COMPACT- записывается без значений, выводит в компактном виде

TYPE- имеет значение disc, circle, square, что соответствует закрашенному кругу, просто кругу и квадрату.

```
<UL TYPE=CIRCLE>
```

Графические маркеры списка. В качестве маркера можно использовать графические изображения, что позволяет сделать документ более красивым и привлекательным.

Пример:

```
<UL>  
<B> Знаки зодиака </B>  
<IMG SRC="oven.jpg">Овен<BR>  
<IMG SRC="telec.jpg">Телец<BR>
```

...

```
<IMG SRC="fish.jpg">Рыбы<BR>  
</UL>
```

Нумерованный список <OL> - Order List, упорядоченный список

Для создания используется тэг-контейнер <OL>...</OL> внутри которого располагают все элементы списка.

Аналогично тэгу UL имеет параметры COMPACT и TYPE, где

TYPE = A – прописные латинские

TYPE = a – строчные латинские

TYPE = I – большие римские

TYPE = i – малые римские

TYPE = 1 – арабские цифры

```
<OL TYPE = I>
```

Или

```
<LI TYPE = A>
```

Значение позволяет менять номер данного элемента.

```
<LI VALUE = 1990>
```

Списки типа <DIR> и <MENU>

В настоящее время не используются, хотя задумывались для:

DIR – отображение в виде списка каталога

MENU – создание меню.



Вложенные списки. Иногда необходимо в элемент списка одного элемента включить целый список такого же типа или другого. При этом получаются многоуровневые или вложенные списки.

В HTML допустимо произвольное вложение различных типов списков, однако следует соблюдать аккуратность при их организации.

Пример:

```
<UL>
  <B> Спутники планет </B>
  <LI> Земля
  <OL>
    <LI> Луна
  </OL>
  <LI> Марс
  <OL>
    <LI>Фобос
    <LI>Деймос
  </OL>
</UL>
```

**Таблицы.** Одним из наиболее мощных и широко применяемых в HTML средств являются таблицы. Таблицы используются для предоставления числовых данных, оформление страниц и т.д. Каждая таблица должна начинаться тэгом <TABLE> и заканчиваться тэгом </TABLE>. Внутри располагается описание содержимого таблицы для чего используются тэги:

```
<TR>...</TR> (TableRow) – строка
<TR>...</TR> (TableData) -ячейка
<TR>...</TR> (Table Header) -ячейка-заголовок (аналог TD)
```

Завершающие тэги </TR>, </TD>, </TH> могут отсутствовать, но не все браузеры корректно воспринимают отсутствие закрывающих тэгов. Закрывающий </TABLE> должен быть всегда!

Пример простейшей таблицы

```
<TABLE>
  <TR>
    <TD> 1.1 </TD>
    <TD> 2.1 </TD>
  </TR>
  <TR>
    <TD> 1.2 </TD>
    <TD> 2.2 </TD>
  </TR>
</TABLE>
```

**Фреймы.** Выбор фреймовой структуры оправдан в случаях:

1. необходимо организовать загрузку документов в одну из под областей браузера.
2. для расположения в определенном месте информации, которая постоянно должна быть на экране (меню).
3. для предоставления информации в разных областях, каждая из которых должна просматриваться отдельно (шапка таблицы).

Типичное использование фреймов:

```
<FRAMESET ROWS="80%,20%">
  <FRAME SRC="LEFT.htm">
  <FRAME SRC=MAIN.htm" name = "pages" =auto>
</FRAMESET>
<FRAME SRC= "top.htm" scrolling = norize>
</FRAMESET>
```

Правила описания. Рассмотрим пример таблицы с фреймами (см. п. 10.1) этот пример задает три фрейма. Горизонтальные в отношении 15% к 85%, горизонтальные 80% к 20%, причем наибольший фрейм может быть с полосой прокрутки, в то время как в нижнем ее быть не может.

```
<FRAMESET>
```

Фреймы определяются структурой FRAMESET, которая используется для страниц с фреймами вместо <BODY>. Внутри контейнера <FRAMESET> могут быть только <FRAME> и вложенные <FRAMESET>

<FRAMESET> имеет два параметра: ROWS(строки) и COLS (столбцы) записываются в виде:

```
<FRAMESET ROWS= "10%, 40%, 50%">
<FRAMESET COLS="*,2*,3*">
```

\* - пропорциональное деление пространства. \* представляет собой одну часть целого. Можно использовать три варианта значений:

```
<FRAMESET COLS = "100,25%, *,2*">
```

<FRAME> - определяет одиночный фрейм. Должен располагаться внутри контейнера <FRAMESET>. Тэг <FRAME> не является контейнером!!!

И имеет 6 параметров:

SRC – адрес док-та загруженного во фрейм.

NAME- имя фрейма, используется для ссылки на него

MARGINWIDTH – ширина полей фрейма в пикселях

MARGINHEIGHT– ширина полей фрейма в пикселях

SCROLLING- (YES,NO,AUTO) наличие полосы прокрутки

NORESIZE- предотвращение изменения размера фрейма.

Пример:

```
<A HREF = "history.htm" TARGET = "Frame_Name"> щелкни тут </A>
<NOFRAMES>
```

Все что размещено между тэгами <NOFRAMES> и </NOFRAMES> будет отображаться браузерами не поддерживающими фреймы.

### ***Варианты заданий***

Создать сайт, используя фреймовую структуру (структура выдается преподавателем), содержащий минимум 3 страницы (автобиография, списки, таблицы). Автобиография должна быть оформлена с использованием тэгов логического и физического форматирования текста. На странице, содержащей списки, необходимо использовать маркированные, нумерованные, вложенные списки. Структура таблицы приведена в приложении В. В одном из фреймов необходимо организовать меню с использованием гипертекстовых переходов как минимум трех видов: абсолютные, относительные, внутри документа.

### ***Пример выполнения работы***

```
<TABLE BORDER=1>
  <CAPTION ALIGN=bottom> Таблица №1 </CAPTION>
  <TR>
    <TD ROWSPAN=2></TD>
    <TH COLSPAN=2>Среднее значение</TH>
  </TR>
  <TR>
    <TH>Рост</TH>
    <TH>Вес</TH>
  </TR>
  <TR>
    <TD>Мужчины</TD>
    <TD ALIGN=center>174</TD>
    <TD ALIGN=center>78</TD>
  </TR>
  <TR>
    <TD>Женщины</TD>
    <TD ALIGN=center>165</TD>
    <TD ALIGN=center>56</TD>
  </TR>
</TABLE>

<FRAMESET ROWS="*,*">
  <NOFRAMES>
    <H1>Ваша версия WEB-броузера не поддерживает фреймы!</H1>
  </NOFRAMES>
```

```
<FRAMESET COLS="65%,35%">
  <FRAME SRC="link1.php">
  <FRAME SRC="link2.php">
</FRAMESET>
<FRAMESET COLS="*,40%,*">
  <FRAME SRC="link3.php">
  <FRAME SRC="link4.php">
  <FRAME SRC="link5.php">
</FRAMESET>
</FRAMESET>
```

<A HREF=index.htm> относительная ссылка </A>

### ***Контрольные вопросы***

1. Что обозначает аббревиатура HTML?
2. Дать определение понятию ТЭГ
3. Какие тэги называют тэгами-контейнерами?
4. Из каких основных частей состоит HTML-документ?
5. Каких рекомендаций следует придерживаться при названии HTML-файлов для избегания ошибок в работе сайта на различных WEB-серверах?
6. Дать определение понятию относительная ссылка
7. Дать определение понятию абсолютная ссылка
8. Что такое URL адрес документа, отношения между документами.
9. Назначение META-тэгов. Параметры тэга META.
10. Основное назначение необязательных параметров тэга BODY.
11. Назовите тэги, описывающие таблицу.
12. Какими тэгами описывается строка таблицы?
13. Каким тэгом описывается ячейка таблицы?
14. Как организовывается объединение строк и столбцов таблицы?
15. Как задать выравнивание в таблице?
16. Как задать толщину границы таблицы?
17. Каким тэгом задается фиксированная ширина колонок таблицы?
18. Дать определение тэгу <FRAMESET>
19. Что определяет тэг <FRAME>?
20. Какой параметр позволяет задавать имя фрейма?
21. Для каких целей фреймы именуются.
22. Что такое планирование фреймов?

## **Лабораторная работа 5 «Электронные таблицы Excel. VBA.»**

### ***Теоретические сведения***

Excel относится к программным продуктам, которые известны под названием электронные таблицы. Электронная таблица – это интерактивная программа, состоящая из набора строк и столбцов, изображенных на экране в специальном окне. Область, находящаяся на пересечении строки и столбца, называется ячейкой. В ячейке могут находиться число, текст или формула, с помощью которой осуществляется вычисления, относящиеся к одной или нескольким ячейкам. Ячейки можно копировать, перемещать, изменять их содержимое. При изменении содержимого ячейки производится автоматический перерасчет содержимого всех ячеек, использующих в формулах измененную ячейку. На основе групп ячеек создаются диаграммы, сводные таблицы и карты.

Создаваемые в Excel файлы называются рабочими книгами (Work Book). Рабочая книга состоит из нескольких листов. Листы могут быть следующих типов:

1. Рабочий лист (WorkSheet) – это самый распространенный лист, который обычно имеют ввиду, говоря об электронной таблице.
2. Лист модуля (Module) – на этом листе пишутся программы Visual Basic for Applications
3. Лист диалогового окна (DialogSheet) – на этом листе создаются диалоговые окна
4. Лист диаграмм (ChartSheet) – на этом листе с помощью мастера диаграмм можно строить диаграммы.

На активном рабочем листе одна ячейка является активной (active) или, другими словами выделенной. Эта ячейка обрамлена черной рамкой. Перемещение рамки по рабочему листу осуществляется мышью или клавишами со стрелками. Каждая ячейка на активном рабочем листе определяется своим адресом, состоящим из имени столбца и номера строки, например A1. Ячейка на неактивном листе идентифицируется именем листа и ее адресом на листе, например Лист2!A1, разделенным восклицательным знаком.

Существует другой способ адресации ячейки – по имени. Имя или адрес активной ячейки выводится в поле имен, которое расположено у левого края строки формул. В именах ячеек не допускаются пробелы, поэтому в качестве разделителя между словами обычно используют символ подчеркивания «\_».

Ввод текстовых данных в ячейку заканчивается нажатием клавиши Enter. Нажатие Esc отменяет ввод или результат редактирования. Для

редактирования содержимого ячейки необходимо ее выделить и нажать клавишу F2 или дважды щелкнуть ячейку.

Формула в ячейке начинается со знака равенства «=» и может содержать операторы:

1. сложение +
2. вычитание –
3. умножение \*
4. деление /
5. возведение в степень ^
6. конкатенация &
7. логическое сравнение =
8. логическое сравнение на больше >
9. логическое сравнение на меньше <

Числовые значения, которые вводятся или выводятся, как правило, не отформатированы и представляют собой последовательности цифр. Для более наглядного представления результатов вычисления их желательно отформатировать. Рассмотрим типы числовых форматов:

Общий (General) – этот формат принят по умолчанию.

Числовой (Number) – этот формат позволяет определить число вводимых знаков после десятичной точки.

Денежный (Currency) – этот формат позволяет определить число выводимых знаков после десятичной точки.

Финансовый (Accounting) – этот формат отличается от денежного тем, что числа выравниваются по десятичной точке.

Дата (Date) – этот формат позволяет вывести дату в одном из 12 возможных вариантов.

Время (Time) – эта категория позволяет вывести время в одном из восьми форматов.

Процентный (Percentage) – этот формат позволяет вывести число, умноженное на 100, со знаком процента и определить число выводимых знаков после десятичной точки.

Дробный (Fraction) – эта категория позволяет вывести число в одном из 9 форматов. При вводе в ячейку чисел в этом формате следует отделять целую часть от дробной пробелом.

Текстовый (Text) – применение этого формата к числовому значению позволяет Excel рассматривать его как текст.

Дополнительный (Special) – в эту категорию входят четыре дополнительных формата: почтовый индекс, индекс+4, номер телефона, табельный номер.

Все форматы (Custom) – эта категория позволяет создавать свои форматы.

Форматировать данные можно как в одной ячейке, так и в диапазоне ячеек или группе несмежных диапазонов ячеек. Диапазон (range) – это прямоугольная группа ячеек. Для его адресации необходимо указать адреса верхней левой и нижней правой ячеек диапазона, разделенные двоеточием, например A1:C3.

### ***Варианты заданий***

Вариант	Условие
1	<p>Построить таблицу «Журнал оценок преподавателя» предусматривающий:</p> <ul style="list-style-type: none"> <li>- лист1 первая группа</li> <li>- лист2 вторая группа</li> <li>- лист3 сводные сведения</li> </ul>
2	<p>Построить таблицу «Журнал оценок преподавателя» предусматривающий:</p> <ul style="list-style-type: none"> <li>- лист1 первая группа</li> <li>- лист2 вторая группа</li> <li>- лист3 сводные сведения</li> </ul> <p>Для первых двух листов:</p> <ul style="list-style-type: none"> <li>- подсчет Качества по каждой работе</li> </ul>
3	<p>Построить таблицу «Журнал оценок преподавателя» предусматривающий:</p> <ul style="list-style-type: none"> <li>- лист1 первая группа</li> <li>- лист2 вторая группа</li> <li>- лист3 сводные сведения</li> </ul> <p>Для первых двух листов:</p> <ul style="list-style-type: none"> <li>- оценки по нескольким видам работ</li> </ul>
4	<p>Построить таблицу «Журнал оценок преподавателя» предусматривающий:</p> <ul style="list-style-type: none"> <li>- лист1 первая группа</li> <li>- лист2 вторая группа</li> <li>- лист3 сводные сведения</li> </ul> <p>Для первых двух листов:</p> <ul style="list-style-type: none"> <li>- подсчет Среднего Балла по каждой работе</li> </ul>
5	<p>Построить таблицу «Журнал оценок преподавателя» предусматривающий:</p> <ul style="list-style-type: none"> <li>- лист1 первая группа</li> <li>- лист2 вторая группа</li> <li>- лист3 сводные сведения</li> </ul> <p>Для первых двух листов:</p> <ul style="list-style-type: none"> <li>- подсчет Успеваемости по каждой работе</li> </ul>

6	Построить таблицу «Журнал оценок преподавателя» предусматривающий: - лист1 первая группа - лист2 вторая группа - лист3 сводные сведения Для первых двух листов: - подсчет Качества Среднего Балла Успеваемости по всей группе
7	Построить таблицу «Журнал оценок преподавателя» предусматривающий: - лист1 первая группа - лист2 вторая группа - лист3 сводные сведения Для сводного листа: - подсчет Качества Среднего Балла Успеваемости для двух групп
8	Построить таблицу «Журнал оценок преподавателя» предусматривающий: - лист1 первая группа - лист2 вторая группа - лист3 сводные сведения Для сводного листа: - диаграмму Качества Среднего Балла Успеваемости для каждого студента каждой группы
9	Построить таблицу «Журнал оценок преподавателя» предусматривающий: - лист1 первая группа - лист2 сводные сведения Для первого листа: - подсчет количества пропусков для каждого студента Для сводного листа: - диаграмма посещения по всем студентам.
10	Построить таблицу «Журнал оценок преподавателя» предусматривающий: - лист1 первая группа - лист2 сводные сведения Для первого листа: - подсчет количества пропусков всех студентов группы
11	Построить таблицу «Журнал оценок преподавателя» предусматривающий: - лист1 первая группа



	<ul style="list-style-type: none"> <li>- лист2 сводные сведения</li> <li>Для первого листа:</li> <li>- подсчет среднего балла и количество пропусков для каждого студента.</li> <li>Для сводного листа:</li> <li>- диаграмма среднего балла и пропусков по всем студентам</li> </ul>
12	<ul style="list-style-type: none"> <li>Построить таблицу «Архив работ» предусматривающий:</li> <li>- лист1 список студентов группы с оценками по работе</li> <li>- лист2 сводные сведения</li> <li>Для сводного листа:</li> <li>- количество сданных работ</li> </ul>
13	<ul style="list-style-type: none"> <li>Построить таблицу «Архив работ» предусматривающий:</li> <li>- лист1 список студентов группы с оценками по работе</li> <li>- лист2 сводные сведения</li> <li>Для первого листа:</li> <li>- количество сданных работ на оценку удовлетворительно.</li> <li>Для сводного листа:</li> <li>- диаграмма оценок по каждому студенту</li> </ul>
14	<ul style="list-style-type: none"> <li>Построить таблицу «Архив работ» предусматривающий:</li> <li>- лист1 список студентов группы с оценками по работе</li> <li>- лист2 сводные сведения</li> <li>Для первого листа:</li> <li>- количество сданных работ, средний балл</li> </ul>
15	<ul style="list-style-type: none"> <li>Построить таблицу «Архив работ» предусматривающий:</li> <li>- лист1 список студентов группы с оценками по работе</li> <li>- лист2 сводные сведения</li> <li>Для первого листа:</li> <li>- средний балл</li> <li>Для сводного листа:</li> <li>- количество сданных работ на оценку удовлетворительно</li> </ul>

### ***Пример выполнения работы***

Создадим таблицу студентов группы, содержащую данные о студенте, итоговую оценку и средний балл по всей группе.

	А	В
1	ФИО	оценка
2	Иванов И.П.	12
3	Петров П.И.	14
4	Сидоров А.А.	17
5		
6		
7		
8	средний балл	14,33

Рисунок – исходная таблица

Для вычисления среднего балла воспользуемся стандартной формулой MS Excel СРЗНАЧ.

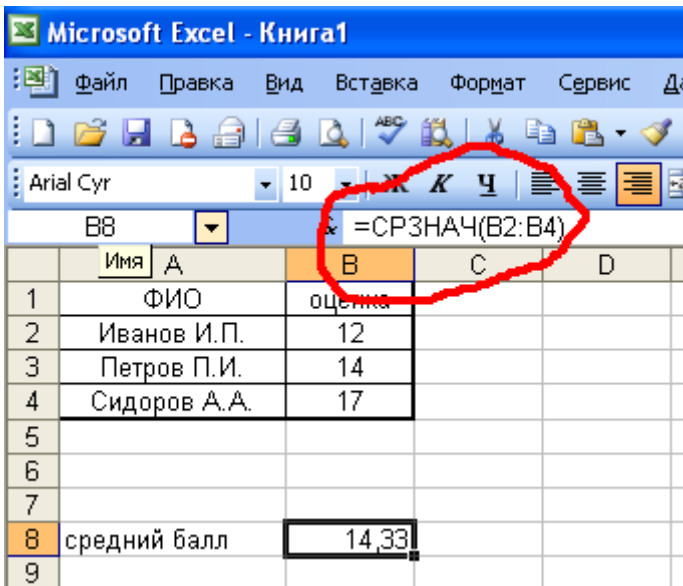


Рисунок – вычисление среднего значения.

Формула СРЗНАЧ возвращает среднее (арифметическое) своих аргументов.

Синтаксис

СРЗНАЧ(число1; число2; ...)

Число1, число2, ... — это от 1 до 30 аргументов, для которых вычисляется среднее.

## Заметки

Аргументы должны быть либо числами, либо именами, массивами или ссылками, содержащими числа.

Если аргумент, который является массивом или ссылкой, содержит тексты, логические значения или пустые ячейки, то такие значения игнорируются; однако ячейки, которые содержат нулевые значения, учитываются.

Устанавливаем формат отображения данных в ячейке, вызвав контекстное меню ячейки для которой необходимо изменить формат.

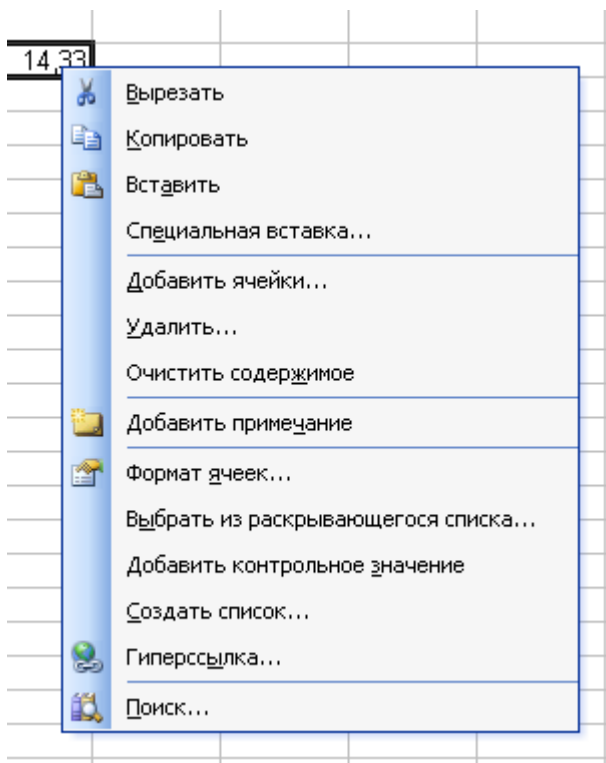


Рисунок – контекстное меню активной ячейки.

Выбираем пункт **Формат ячеек ...** и в появившемся диалоговом окне задаем формат.

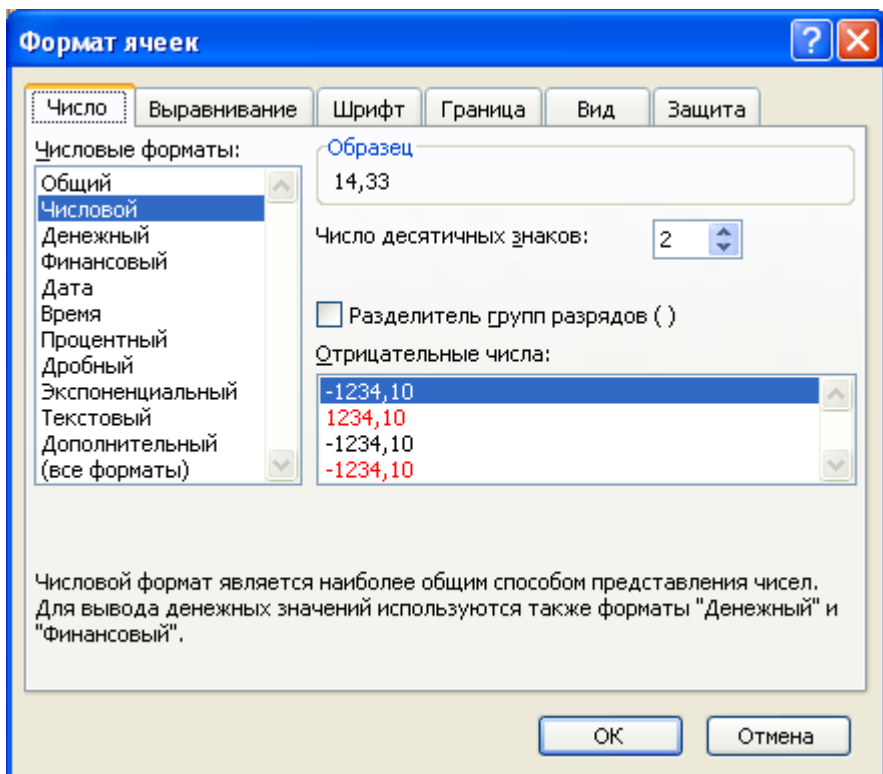


Рисунок – диалоговое окно установки формата ячеек.

### ***Контрольные вопросы***

1. Что такое диапазон ячеек?
2. Какие программы называют электронными таблицами?
3. Что такое электронные таблицы?
4. Что такое ячейка?
5. Как определяется адрес ячейки?
6. Какие данные может содержать ячейка?
7. Что такое рабочая книга и из каких элементов она состоит?
8. Перечислите типы листов.
9. Что называется активной ячейкой?
10. Какие есть способы адресации ячеек?
11. Для чего необходимо форматировать данные в ячейках?
12. Какие существуют типы форматов ячеек?

## СПИСОК ЛИТЕРАТУРЫ

БЕЗРУКОВ Н.Н. КОМПЬЮТЕРНАЯ ВИРУСОЛОГИЯ. Часть 1: Общие принципы функционирования, классификация и каталог наиболее распространенных вирусов в операционной системе MS DOS / Редакция 5.5 от 10.11.90. - 1990. - 450 с. Ил. 11, список лит.: 340 назв., прил. 9.

Хижняк П.Л. "Пишем вирус... и антивирус!", М.: Инфо, 1991

Абакумов А.А., Абрамов С.М. и др. Правдивая история о жизни и смерти одного вируса // Наука в СССР, 1989, N 4.- С.83-87

Абрамов С.М., Пименов С.П. и др. Компьютерный вирус // Микропроцессорные средства и системы, 1989, N 2.- С.22-24

Агасандян Г. Не вреди ближнему своему // Компьютер, 1990, N 1.- С.47-49

Агеев А.С. "Компьютерные вирусы" и безопасность информации // Зарубежная радиоэлектроника, 1989, N 12.- С.71-75

Безруков Н.Н. Классификация компьютерных вирусов и средства защиты от них // Эксплуатация программного обеспечения вычислительных систем реального времени, построенных на базе микро и мини ЭВМ. - Киев: КИИГА, 1989.- С.3-21

Безруков Н.Н. Классификация компьютерных вирусов в MS DOS // Программирование, 1990, N 3.- С.3-22

Безруков Н.Н. Классификация вирусов: попытка стандартизации // ИК, 1990, N 2.- С.37-39; N 3.- С.38-47

Безруков Н.Н. Классификация компьютерных вирусов MS DOS и методы защиты от них.- М.: СП "Информэйшн Компьютер Энтерпрайз", 1990. - 48 с.

Держинский Ф.Я. Комментарий о вирусах и компьютерном пиратстве // Программирование, 1990, N 3.- С.23-24

Диев С.И. Защита информации в персональных компьютерах // Зарубежная радиоэлектроника, 1989, N 12.- С.57-59

Карасик И.Ш. К вопросу о компьютерных вирусах // Мир ПК, 1989, N 3.- С.127-131

Карасик И.Ш. Типология вирусов // ИК, 1989, N 2.- С.14-15

Карасик И.Ш. Анатомия и физиология вирусов // ИК, 1990, N 1.- С.39-47

Лозинский Д. Одна из советских антивирусных программ: AIDSTEST // КП, 1990, N 6.- С.17-20

Николаев А. Осторожно - вирус ! // КП, 1990, N 6.- С.3-16

Селль М. Антивирусные программы // Компьютер, 1990, N 2.- С.48-50

Чижов А.А. Некоторые соображения по поводу компьютерных вирусов // В мире персональных компьютеров, 1988, N 1.- С.121-124

Шерстюк Ф.Н. Вирусы и антивирусы на IBM-совместимых ПК // ИК,  
1990, N 2.- С.46-47

## ПРИЛОЖЕНИЕ А

### ТАБЛИЦА БАЗОВЫХ ЦВЕТОВ

Aliceblue	Antiquewhite	Aqua	Aquamarine
F0F8FF	FAEBD7	00FFFF	7FFFD4
Azure	Beige	Bisque	Black
F0FFFF	F5F5DC	FFE4C4	000000
Blanchedalmond	Blue	Blueviolet	Brown
d	0000FF	8A2BE2	A52A2A
FFEBCD			
Burlywood	Cadetblue	Chartreuse	Chocolate
DEB887	5F9EA0	7FFF00	D2691E
Coral	Cornflowerblue	Cornsilk	Crimson
FF7F50	6495ED	FFF8DC	DC143C
Cyan	Darkblue	Darkcyan	Darkgoldenrod
00FFFF	00008B	008B8B	B8860B
Darkgray	Darkgreen	Darkkhaki	Darkmagenta
A9A9A9	006400	BDB76B	8B008B
Darkolivegreen	Darkorange	Darkorchid	Darkred
556B2F	FF8C00	9932CC	8B0000
Darksalmon	Darkseagreen	Darkslateblue	Darkslategray
E9967A	8FBC8F	483D8B	2F4F4F
Darkturquoise	Darkviolet	deeppink	Deepskyblue
00CED1	9400D3	FF1493	00BFFF
Dimgray	Dodgerblue	Firebrick	Floralwhite
696969	1E90FF	B22222	FFFAF0
Forestgreen	Fuchsia	Gainsboro	Ghostwhite
228B22	FF00FF	DCDCDC	F8F8FF
Gold	Goldenrod	Gray	Green
FFD700	DAA520	808080	008000
Greenyellow	Honeydew	Hotpink	Indianred
ADFF2F	F0FFF0	FF69B4	CD5C5C
Indigo	Ivory	Khaki	Lavendar
4B0082	FFFFFF0	F0E68C	E6E6FA
Lavenderblush	Lawngreen	Lemonchiffon	Lightblue
FFF0F5	7CFC00	FFFACD	ADD8E6
Lightcoral	Lightcyan	Lightgoldenrodyello	Lightgreen
F08080	E0FFFF	w	90EE90
		FAFAD2	
Lightgrey	Lightpink	Lightsalmon	Lightseagreen

D3D3D3	FFB6C1	FFA07A	20B2AA
Lightskyblue	Lightslategray	Lightsteelblue	Lightyellow
87CEFA	778899	B0C4DE	FFFFE0
Lime	Limegreen	Linen	Magenta
00FF00	32CD32	FAF0E6	FF00FF
Maroon	Mediumaquamarine	Mediumblue	Mediumorchid
800000	66CDAA	0000CD	BA55D3
Mediumpurple	Mediumseagreen	Mediumslateblue	Mediumspringgreen
9370D8	3CB371	7B68EE	00FA9A
Mediumturquoise	Mediumvioletred	Midnightblue	Mintcream
48D1CC	C71585	191970	F5FFFA
Mistyrose	Moccasin	Navajowhite	Navy
FFE4E1	FFE4B5	FFDEAD	000080
Oldlace	Olive	Olivedrab	Orange
FDF5E6	808000	688E23	FFA500
Orangered	Orchid	Palegoldenrod	Palegreen
FF4500	DA70D6	EEE8AA	98FB98
Paleturquoise	Palevioletred	Papayawhip	Peachpuff
AFEEEE	D87093	FFEFD5	FFDAB9
Peru	Pink	Plum	Powderblue
CD853F	FFC0CB	DDA0DD	B0E0E6
Purple	Red	Rosybrown	Royalblue
800080	FF0000	BC8F8F	4169E1
Saddlebrown	Salmon	Sandybrown	Seagreen
8B4513	FA8072	F4A460	2E8B57
Seashell	Sienna	Silver	Skyblue
FFF5EE	A0522D	C0C0C0	87CEEB
Slateblue	Slategray	Snow	Springgreen
6A5ACD	708090	FFFAFA	00FF7F
Steelblue	Tan	Teal	Thistle
4682B4	D2B48C	008080	D8BFD8
Tomato	Turquoise	Violet	Wheat
FF6347	40E0D0	EE82EE	F5DEB3
White	Whitesmoke	Yellow	YellowGreen
FFFFFF	F5F5F5	FFFF00	9ACD32



## ПРИЛОЖЕНИЕ Б

### СПЕЦИАЛЬНЫЕ СИМВОЛЫ

Некоторые символы не входят в базовую часть таблицы кодов ASCII. К ним относятся буквы алфавитов части европейских языков, математические и некоторые другие символы. Некоторые символы, введенные в HTML документ будут интерпретироваться не так, как задумал автор. Это, например, символы "<" и ">", используемые для указания тегов.

В этих случаях можно вводить нужные символы с помощью специальных кодов. Коды начинаются с символа "амперсанд"(&). За ним следует название символа либо его числовой код в десятичной или шестнадцатеричной системе. Завершает код символ "точка с запятой"(;).

<b>Имя</b>	<b>Код</b>	<b>Вид</b>	<b>Описание</b>
&quot;	&#34;	"	двойная кавычка
&amp;	&#38;	&	амперсанд
&lt;	&#60;	<	знак 'меньше'
&gt;	&#62;	>	знак 'больше'
&nbsp;	&#160;		неразрывный пробел
&iexcl;	&#161;	¡	перевернутый восклицательный знак
&cent;	&#162;	¢	цент
&pound;	&#163;	£	фунт стерлингов
&curren;	&#164;	¤	денежная единица
&yen;	&#165;	¥	иена или юань
&brvbar;	&#166;	¦	разорванная вертикальная черта
&sect;	&#167;	§	параграф
&uml;	&#168;	¨	умляут
&copy;	&#169;	©	знак соруригth
&ordf;	&#170;	ª	женский порядковый числитель
&laquo;	&#171;	«	левая двойная угловая скобка
&not;	&#172;	¬	знак отрицания

&shy;	&#173;		место возможного переноса
&reg;	&#174;	®	знак зарегистрированной торговой марки
&macr;	&#175;	-	верхняя горизонтальная черта
&deg;	&#176;	°	градус
&plusmn;	&#177;	±	плюс-минус
&sup2;	&#178;	²	"в квадрате"
&sup3;	&#179;	³	"в кубе"
&acute;	&#180;	´	знак ударения
&micro;	&#181;	μ	микро
&para;	&#182;	¶	символ параграфа
&middot;	&#183;	·	точка
&cedil;	&#184;	¸	седиль (орфографический знак)
&sup1;	&#185;	¹	верхний индекс 'один'
&ordm;	&#186;	º	мужской порядковый числитель
&raquo;	&#187;	»	правая двойная угловая скобка
&frac14;	&#188;	¼	одна четвертая
&frac12;	&#189;	½	одна вторая
&frac34;	&#190;	¾	три четвертых
&iquest;	&#191;	¿	перевернутый вопросительный знак
&fnof;	&#402;	<i>f</i>	знак функции
<b>Символы для букв</b>			
&circ;	&#710;	ˆ	диакритический знак над гласной
&tilde;	&#732;	˜	тильда
<b>Греческие буквы</b>			
&Alpha;	&#913;	Α	заглавная альфа
&Beta;	&#914;	Β	заглавная бета

&Gamma;	&#915;	Г	заглавная гамма
&Delta;	&#916;	Δ	заглавная дельта
&Epsilon;	&#917;	Ε	заглавная эpsilon
&Zeta;	&#918;	Z	заглавная дзета
&Eta;	&#919;	Η	заглавная эта
&Theta;	&#920;	Θ	заглавная тета
&Iota;	&#921;	Ι	заглавная иота
&Kappa;	&#922;	Κ	заглавная каппа
&Lambda;	&#923;	Λ	заглавная лямбда
&Mu;	&#924;	Μ	заглавная мю
&Nu;	&#925;	Ν	заглавная ню
&Xi;	&#926;	Ξ	заглавная кси
&Omicron;	&#927;	Ο	заглавная омикрон
&Pi;	&#928;	Π	заглавная пи
&Rho;	&#929;	Ρ	заглавная ро
&Sigma;	&#931;	Σ	заглавная сигма
&Tau;	&#932;	Τ	заглавная тау
&Upsilon;	&#933;	Υ	заглавная ипсилон
&Phi;	&#934;	Φ	заглавная фи
&Chi;	&#935;	Χ	заглавная хи
&Psi;	&#936;	Ψ	заглавная пси
&Omega;	&#937;	Ω	заглавная омега
&alpha;	&#945;	α	строчная альфа
&beta;	&#946;	β	строчная бета
&gamma;	&#947;	γ	строчная гамма
&delta;	&#948;	δ	строчная дельта
&epsilon;	&#949;	ε	строчная эpsilon
&zeta;	&#950;	ζ	строчная дзета
&eta;	&#951;	η	строчная эта
&theta;	&#952;	θ	строчная тета
&iota;	&#953;	ι	строчная иота
&kappa;	&#954;	κ	строчная каппа
&lambda;	&#955;	λ	строчная лямбда
&mu;	&#956;	μ	строчная мю

&nu;	&#957;	v	строчная ню
&xi;	&#958;	ξ	строчная кси
&omicron;	&#959;	ο	строчная омикрон
&pi;	&#960;	π	строчная пи
&rho;	&#961;	ρ	строчная ро
&sigmaf;	&#962;	ς	строчная сигма (final)
&sigma;	&#963;	σ	строчная сигма
&tau;	&#964;	τ	строчная тау
&upsilon;	&#965;	υ	строчная ипсилон
&phi;	&#966;	φ	строчная фи
&chi;	&#967;	χ	строчная хи
&psi;	&#968;	ψ	строчная пси
&omega;	&#969;	ω	строчная омега

### Общая пунктуация

&ndash;	&#8211;	—	тире
&mdash;	&#8212;	—	длинное тире
&lsquo;	&#8216;	‘	левая одиночная кавычка
&rsquo;	&#8217;	’	правая одиночная кавычка
&sbquo;	&#8218;	‚	нижняя одиночная кавычка
&ldquo;	&#8220;	“	левая двойная кавычка
&rdquo;	&#8221;	”	правая двойная кавычка
&bdquo;	&#8222;	„	нижняя двойная кавычка

### Прочие символы

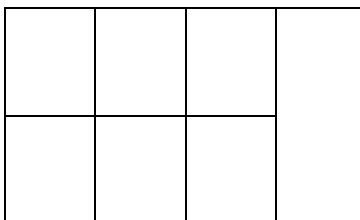
	&#8224	†	латинский крест
	&#8225	‡	двойной крест
&bull;	&#8226;	•	маленький черный кружок
&hellip;	&#8230;	...	многоточие ...
	&#8240;	%	знак промилле

			(процент с дополнительным ноликом)
&prime;	&#8242;	'	одиночный штрих - минуты
&Prime;	&#8243;	"	двойной штрих - секунды
&oline;	&#8254;	-	надчеркивание
&frasl;	&#8260;	/	косая дробная черта
	&#8364;	€	евро
	&#8470;	№	знак номера
&trade;	&#8482;	™	знак торговой марки
<b>Стрелки</b>			
&larr;	&#8592;	←	стрелка влево
&uarr;	&#8593;	↑	стрелка вверх
&rarr;	&#8594;	→	стрелка вправо
&darr;	&#8595;	↓	стрелка вниз
&harr;	&#8596;	↔	стрелка влево-вправо
<b>Масти</b>			
&spades;	&#9824;	♠	знак масти 'пики'
&clubs;	&#9827;	♣	знак масти 'трефы'
&hearts;	&#9829;	♥	знак масти 'червы'
&diams;	&#9830;	♦	знак масти 'бубны'

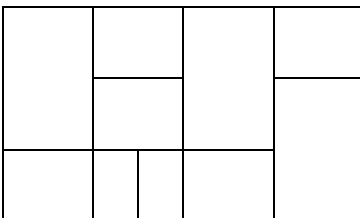
## ПРИЛОЖЕНИЕ В

### ВАРИАНТЫ ЗАДАНИЙ

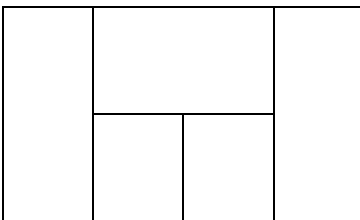
1



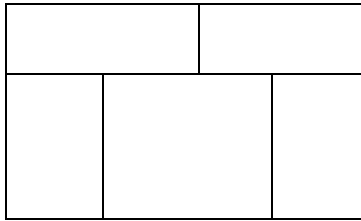
2



3



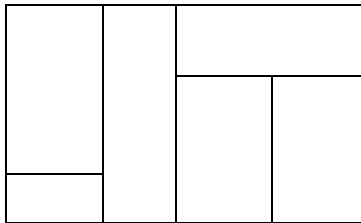
4



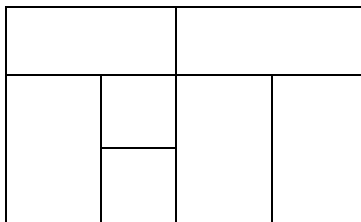
5



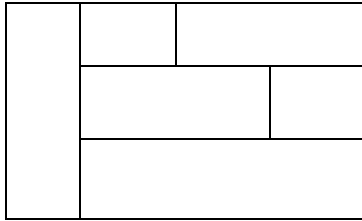
6



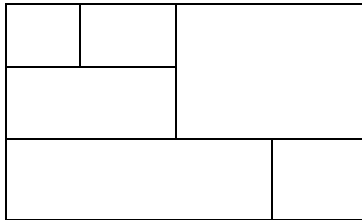
7



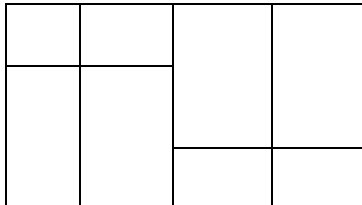
8



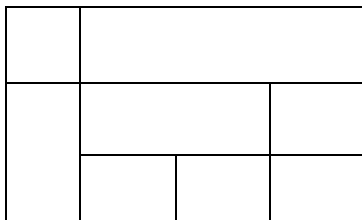
9



10

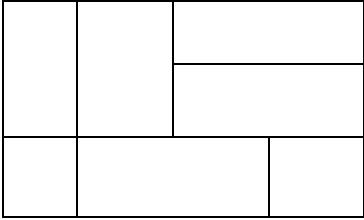


11

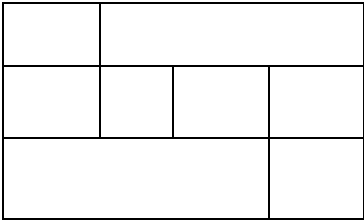




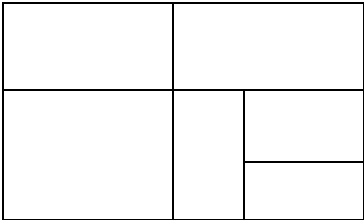
12



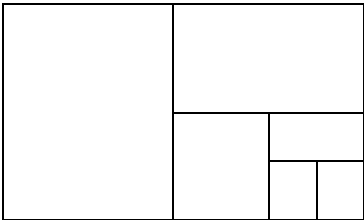
13



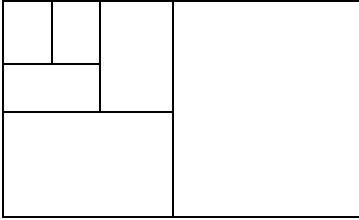
14



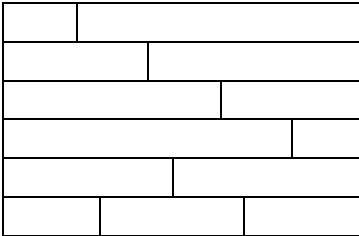
15



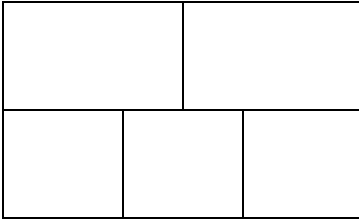
16



17



18



19

