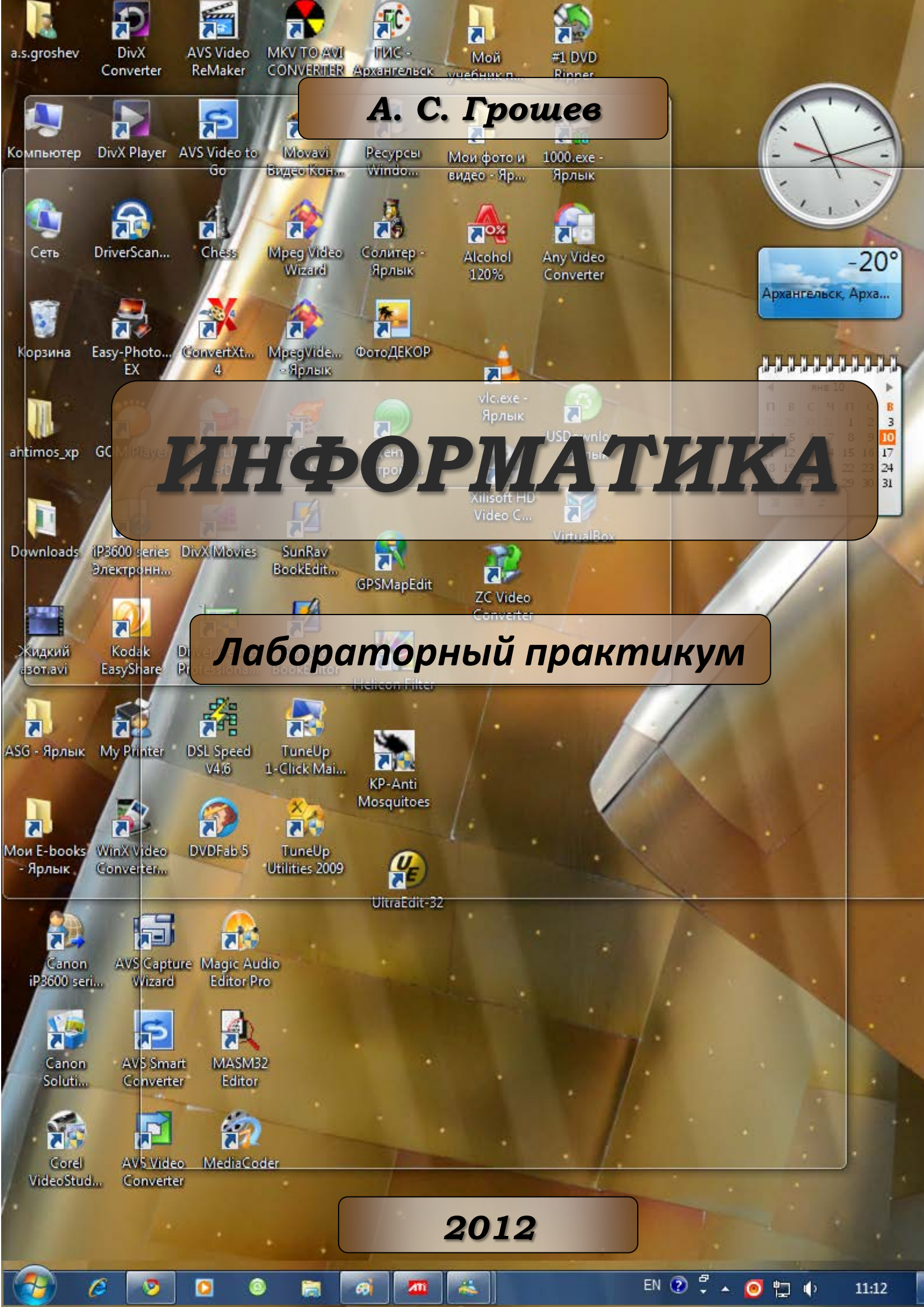
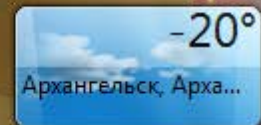


А. С. Грошев

ИНФОРМАТИКА

Лабораторный практикум

2012



*Министерство образования и науки российской федерации
Северный (Арктический) федеральный университет
им. М. В. Ломоносова*

А. С. Грошев

Информатика

Лабораторный практикум

***Архангельск
2012***

Рассмотрено и рекомендовано к изданию методической комиссией

_____ 2012 г.

Рецензент

УДК 681.3 (075)
ББК 32.973.233
Г 89

Грошев А. С.

Г89 Информатика: лабораторный практикум / А.С. Грошев. – Архангельск, Арханг. гос. техн. ун-т, 2012. – 148 с.

Для студентов ИТиПХ специальностей 240100, 240700, 241000, 020201.

Содержит лабораторные работы по основным разделами информатики.

Особенность данного учебного пособия – все работы выполняются в новых версиях операционной системы Microsoft Windows и Office.

Ил. 128. Табл. 11.

© Грошев А. С., 2012

ОГЛАВЛЕНИЕ

Лабораторная работа № 1. Основы работы в операционной системе Windows 7.....	4
Лабораторная работа № 2. Использование стандартных программ операционной системы Windows 7.....	15
Лабораторная работа № 3. Microsoft Word 2010. Настройка стилей. Работа с текстом.....	29
Лабораторная работа № 4. Microsoft Word 2010. Работа с таблицами, формулами, диаграммами, рисунками.....	41
Лабораторная работа № 5. Microsoft Excel 2010. Работа с формулами. Использование абсолютной и относительной адресации. Математические функции.....	54
Лабораторная работа № 6. Microsoft Excel 2010. Использование функций для работы с данными типа Текст и Дата/время.....	70
Лабораторная работа № 7. Microsoft Excel 2010. Построение диаграмм...	78
Лабораторная работа № 8. Microsoft Excel 2010. Технологический расчет.....	82
Лабораторная работа № 9. Microsoft Access 2010. Проектирование и создание базы данных.....	87
Лабораторная работа № 10. Microsoft Access 2010. Занесение информации в базу данных. Разработка форм.....	92
Лабораторная работа № 11. Microsoft Access 2010. Разработка отчетов и запросов.....	102
Лабораторная работа № 12. Программирование на языке VBS. Работа с окнами сообщений и ввода данных.....	114
Лабораторная работа № 13. Программирование на языке VBS. Типы данных. Константы. Переменные.....	121
Лабораторная работа № 14. Программирование на языке VBS. Массивы.....	127
Лабораторная работа № 15. Программирование на языке VBS. Условные операторы IF и CASE.....	131
Лабораторная работа № 16. Программирование на языке VBS. Операторы цикла Do и While.....	138
Лабораторная работа № 17. Программирование на языке VBS. Операторы цикла For и For Each.....	143
Приложение 1. Горячие клавиши Windows 7.....	147
Литература.....	148

Лабораторная работа № 1.

Основы работы в операционной системе Windows 7

Цель работы

Приобретение навыков работы в операционной системе Windows

Учебные вопросы

1. Регистрация пользователя в системе.
2. Изучение папок профиля пользователя и библиотек.
3. Настройка компонентов рабочего стола (меню кнопки Пуск, значки панели задач, ярлыки Рабочего стола, гаджеты).
4. Настройка и использование Проводника. Операции с папками и файлами.
5. Способы запуска программ и открытия файлов данных.

Краткое изложение учебного материала

1 Регистрация пользователя в системе

В компьютерном классе преподаватель сообщает студентам данные учётной записи, которую они должны использовать при работе на компьютерах (**имя пользователя и пароль**). Обычно используется учетная запись, зарегистрированная на контроллере домена университета.

В начале работы необходимо правильно задать эти данные на экране регистрации, показанном на рисунке 1.1.

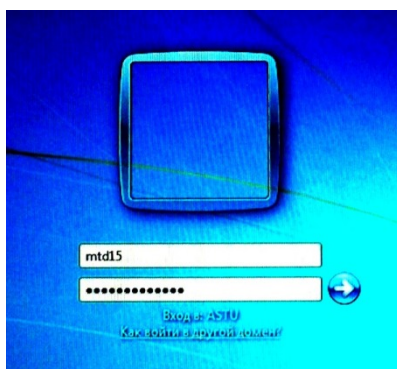


Рисунок 1.1 – Экран регистрации пользователя

При этом следует учитывать, что в имени пользователя нельзя ставить лишние пробелы в конце слова, а в пароле – необходимо правильно указывать прописные и строчные буквы (пароль чувствителен к регистру букв!).

2 Изучение папок профиля пользователя и библиотек

При первом входе пользователя в систему создается локальный профиль учётной записи (Ваши папки на том компьютере, где Вы реги-

стрируетесь). На правой панели кнопки **Пуск** верхняя позиция – *Открытие персональной папки* (рисунок 1.2). Именно в этой папке и находятся все папки профиля пользователя.

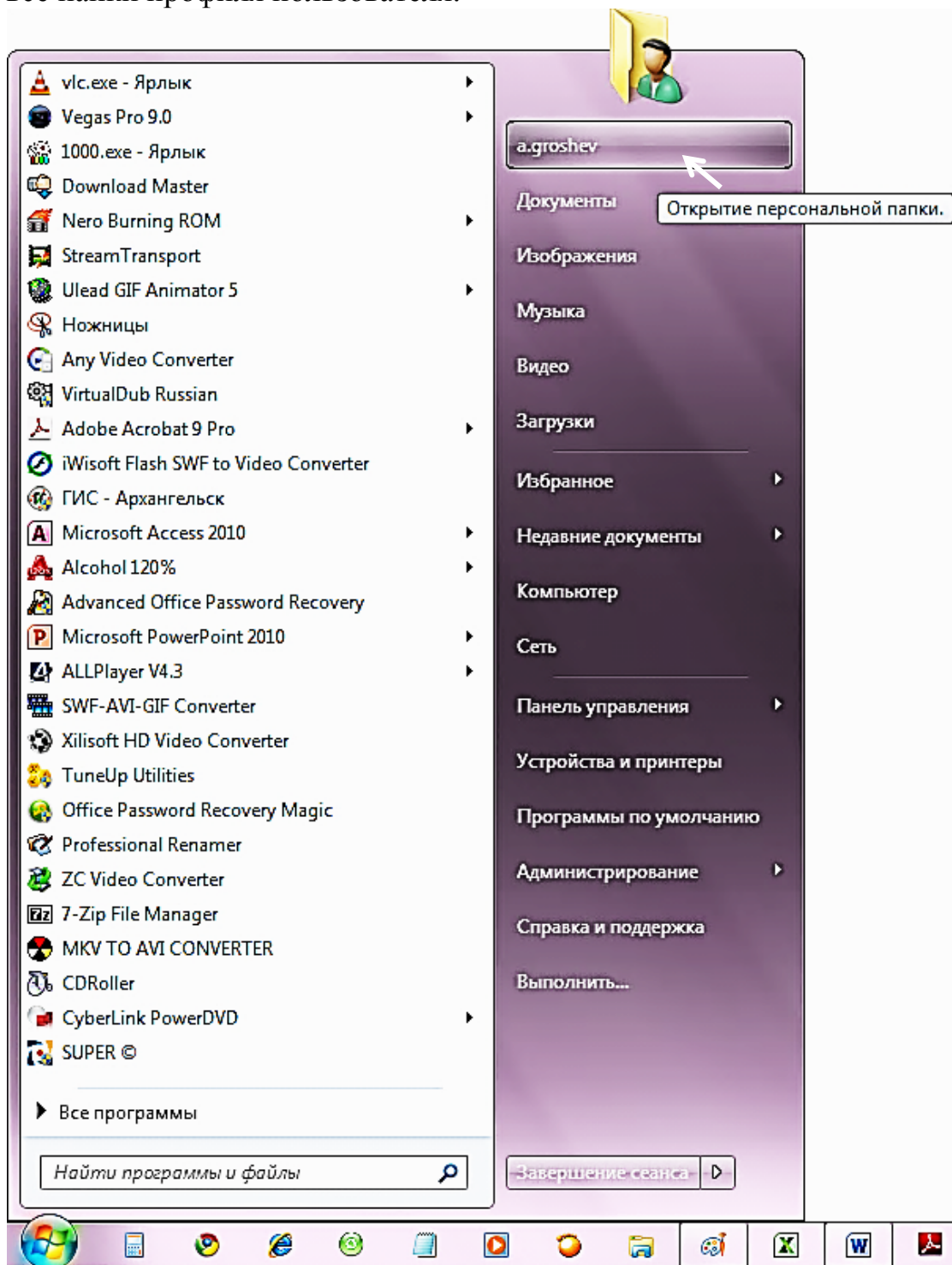


Рисунок 1.2 – Меню кнопки Пуск

Папки профилей находятся на диске, где установлена операционная система, в папке *Пользователи* (рисунок 1.3).

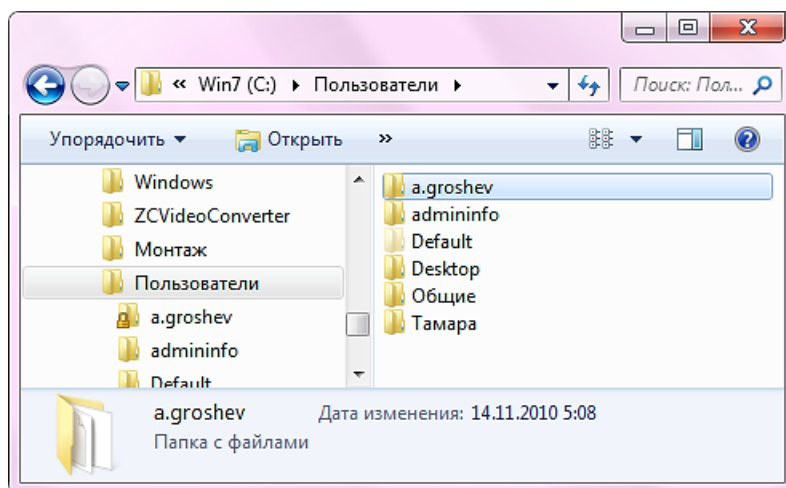


Рисунок 1.3 – Папка профилей пользователей в Проводнике

В папке каждого пользователя находятся вложенные папки *Загрузки*, *Изображения*, *Контакты*, *Мои видеозаписи*, *Мои документы*, *Моя музыка*, *Поиски*, *Рабочий стол*, *Ссылки* (рисунок 1.4) и несколько скрытых системных папок.

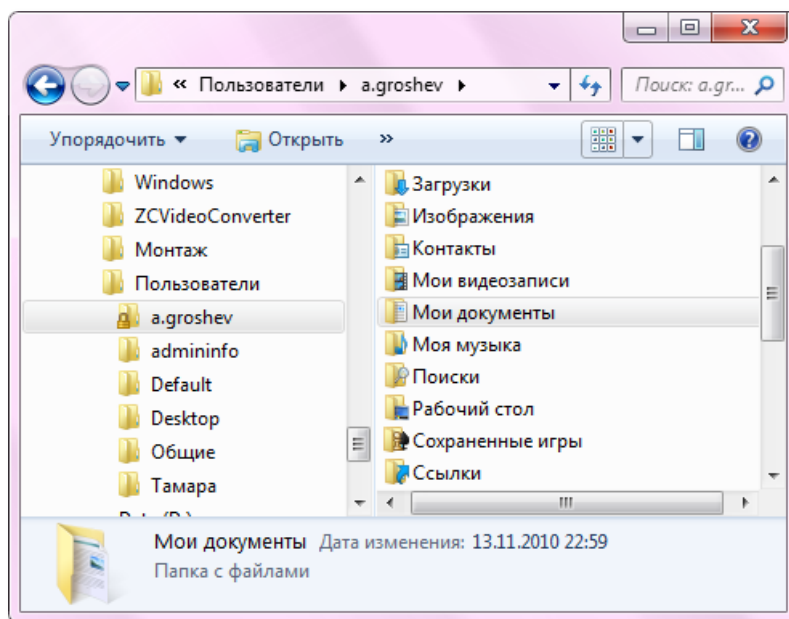


Рисунок 1.4 – Папки в профиле пользователя

Следует знать, что в меню кнопки **Пуск** присутствуют пункты *Документы*, *Изображения*, *Музыка*, *Видео*, *Загрузки* (см. ранее рисунок 1.2), которые открывают библиотеки пользователя с соответствующими названиями [1, стр. 162 – 163]. В состав библиотек могут входить различные папки, список которых пользователь может изменять. Например, в состав библиотеки **Документы** входят папки **Мои документы** и **Общие документы**.

Для сохранения файлов пользователя, которые будут созданы во время выполнения лабораторных работ, следует внутри папки **Мои доку-**

менты создать папку с именем <фамилия и инициалы пользователя>.

3 Настройка компонентов рабочего стола

Для настройки Рабочего стола [1, стр. 153 – 156] следует использовать контекстное меню, которое появляется при щелчке правой кнопкой мыши на экране компьютера (рисунок 1.5).

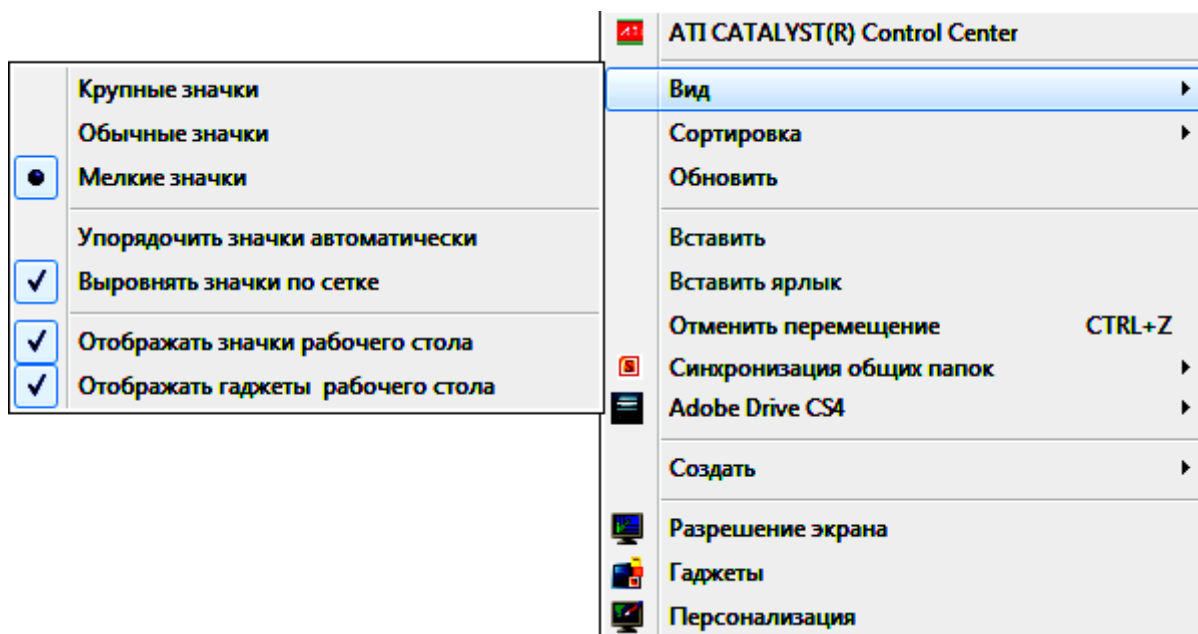


Рисунок 1.5 – Контекстное меню настройки Рабочего стола

Как видно из рисунка 1.5, меню позволяет настроить вид значков (ярлыков) на **Рабочем столе**, разрешение экрана (для жидкокристаллических мониторов следует задавать их штатное разрешение), используемые гаджеты (миниприложения) и обратиться к диалоговому окну **Персонализация**, содержащему настройки тем **Рабочего стола** и других параметров (рисунок 1.6).

Для настройки **Панели задач**, меню кнопки **Пуск** и **Панелей инструментов** (которые могут располагаться на **Панели задач**) следует использовать пункт **Свойства** контекстного меню этих элементов.

В настройках меню Пуск можно задать наличие или отсутствие пунктов меню и их вид (подменю или ссылка). По умолчанию некоторые пункты меню отключены (*Администрирование*, *Выполнить* и пр.), при необходимости можно задать их использование (рисунок 1.7). В этом окне можно настроить также количество строк в списке недавних программ (левая панель при нажатии на кнопку **Пуск**) и в списке **Недавние документы** (см. ранее на рисунке 1.2).

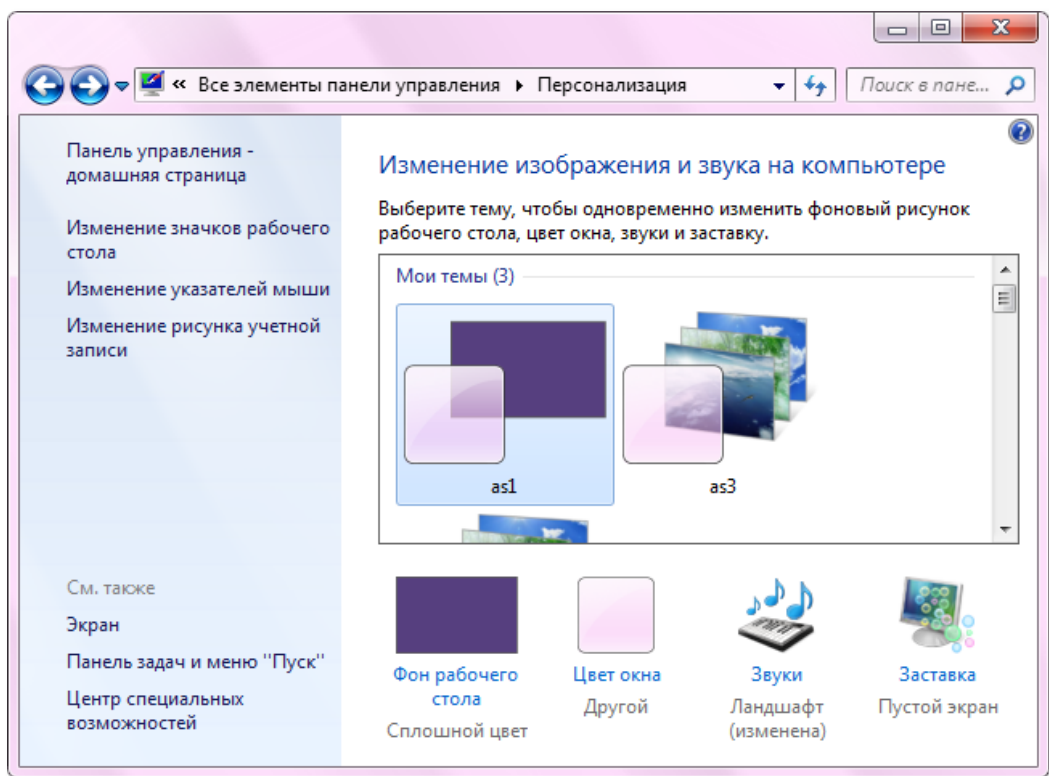


Рисунок 1.6 – Окно **Персонализация** для настройки **Рабочего стола**

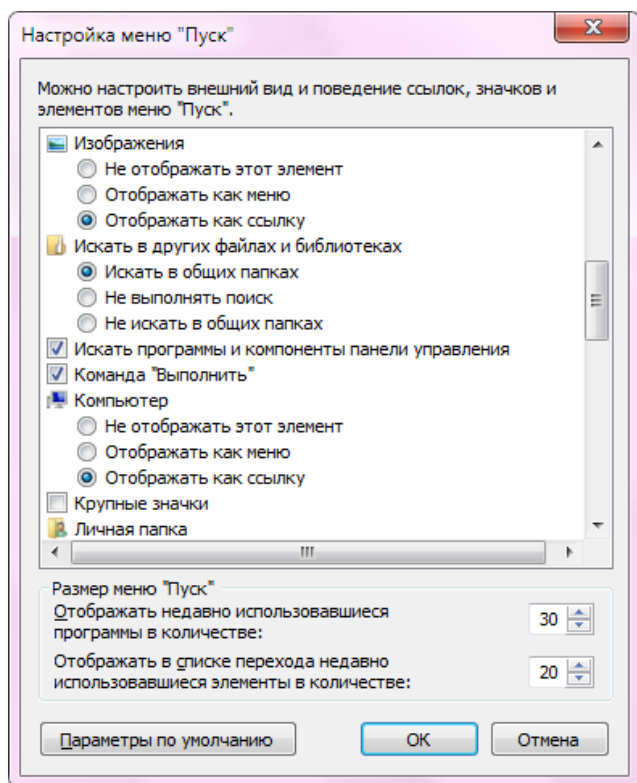


Рисунок 1.7 – Окно настройки меню **Пуск**

Значки на Панели задач появляются для запущенных программ. Для тех программ, которыми Вы часто пользуетесь (в том числе и для **Провод-**

ника), полезно закрепить значок на **Панели задач** (в контекстном меню для значка программы команда *Закрепить программу в панели задач*). Это позволит не только быстро запускать нужные программы, но и сразу выбирать нужные файлы и папки для **Проводника** в **Списке переходов** (Jump List – список последних документов, например, для Word, или часто используемых папок, например, для **Проводника**), который появляется при щелчке правой кнопкой мыши на значке программы (рисунок 1.8). Необходимые строки списка можно закрепить как постоянные в **Списке переходов**. Видеодемонстрацию процесса использования **Списков переходов** можно посмотреть на сайте Microsoft (<http://windows.microsoft.com/ru-RU/windows7/Using-Jump-Lists-to-open-programs-and-items>).

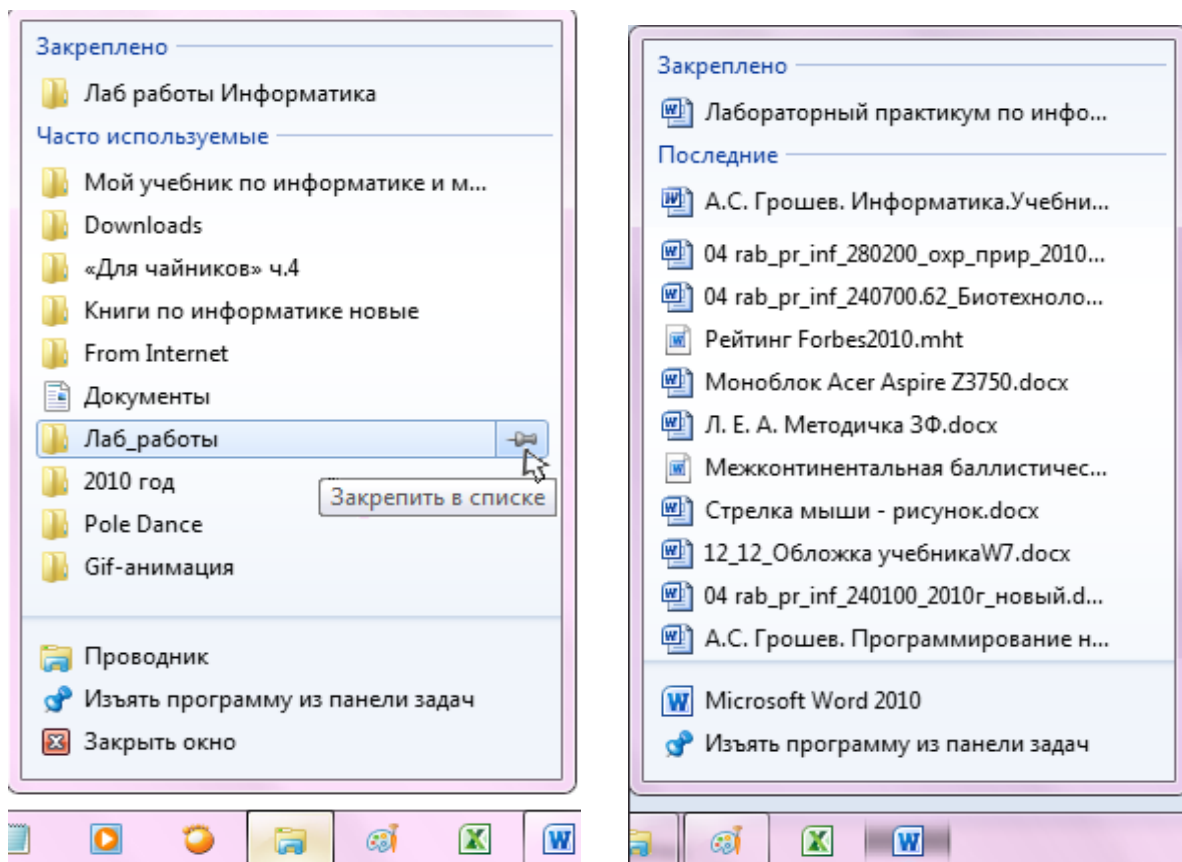


Рисунок 1.8 – Списки переходов для Проводника и для Word

Ярлыки программ (значки, содержащие ссылку на объект) появляются на Рабочем столе при инсталляции многих программ. Первоначально присутствует один ярлык – Корзина, в окне Персонализация (см. ранее рисунок 1.6) есть ссылка *Изменение значков рабочего стола*, которая позволяет разместить на Рабочем столе дополнительные ярлыки (с особыми свойствами!): **Компьютер**, **Файлы пользователя**, **Сеть**, **Панель управления**.

Гаджеты Рабочего стола используются обычно для показа некоторой информации: часы и календарь, загрузка процессоров и оперативной памяти.

ти, мониторинг работы сети, показ погоды, теле- и радиоканалов Интернета и пр. Среди них есть такие, которые работают только при наличии подключения к Интернету (погода, курсы валют, Интернет-телевидение и пр.). Новые гаджеты можно скачать и установить с сайта Microsoft (<http://gallery.live.com/>).

4 Настройка и использование Проводника. Операции с папками и файлами.

Проводник или **Windows Explorer** – стандартная программа, которая позволяет работать с папками и файлами Вашего компьютера и компьютеров локальной сети. Вид окна **Проводника** показан ранее на рисунках 1.3 и 1.4, однако возможности по его настройке очень большие [1, стр. 158 – 162].

Наиболее часто **Проводник** используется для нахождения нужных файлов и папок и выполнения с ними различных операций: открытия, создания новых файлов и папок, их копирования, перемещения, удаления и других операция. Список возможных операций для файлов и папок можно увидеть в **контекстном меню** (рисунок 1.9). Состав команд контекстного меню изменяется в зависимости от типа выбранных файлов и наличия прикладных программ, которые вписывают свои команды в контекстное меню, например, команды для программ ESET NOD32 и WinRar на рисунке 1.10. На том же рисунке показано подменю для пункта **Отправить**, которое содержит, в частности, команду архивации средствами Windows – **Сжатая ZIP-папка**.

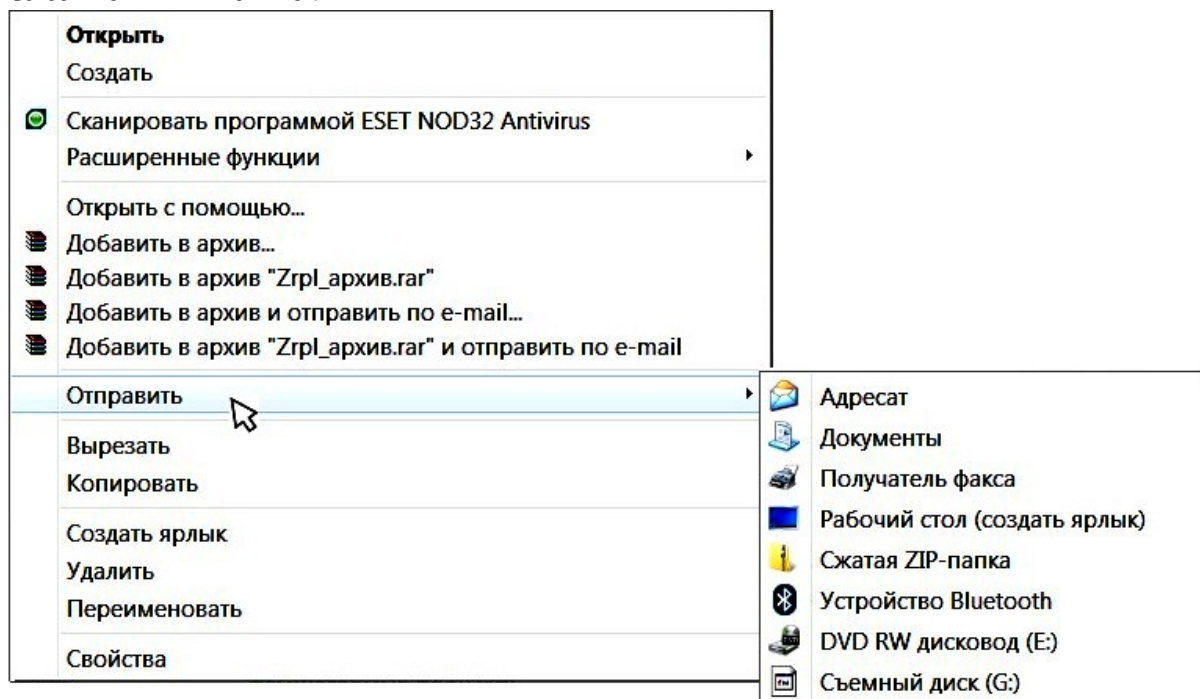


Рисунок 1.9 – Контекстное меню для файлов Проводника

Команду **Создать** контекстного меню (рисунок 1.10) удобно использовать для создания новых файлов определенного типа в выбранной папке. Для создания папок можно использовать кнопку **Новая папка** на панели команд **Проводника**.

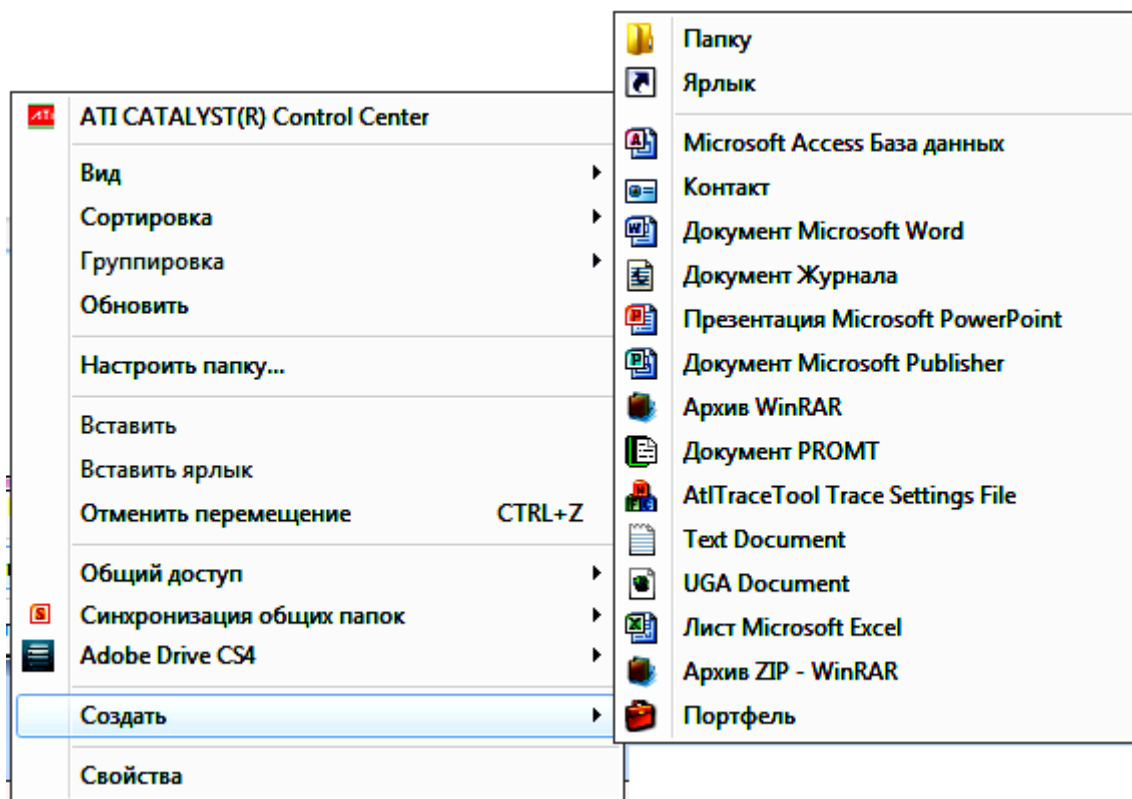


Рисунок 1.10 – Контекстное меню **Создать Проводника**

Для копирования или перемещения файлов из одной папки в другую, можно использовать как контекстное меню **Проводника** (команды Копировать и Вставить или Вырезать и Вставить), горячие клавиши CTRL+C и CTRL+V или CTRL+X и CTRL+V (другие горячие клавиши Windows см. в Приложении 1).

Другой способ, который часто используется – перетаскивание мышкой из одной папки в другую. При перетаскивании с нажатой левой кнопкой мыши у курсора появляется сообщение о виде операции (Копировать в ..., Переместить в ... или Создать ссылку в ...), как показано на рисунке 1.11. Вид операции меняется при нажатии кнопки CTRL или CTRL+SHIFT для последней операции. При перетаскивании с нажатой правой кнопкой мыши после выбора места, куда выполняется эта операция и отпускания кнопки мыши, появляется контекстное меню для выбора одной из трех названных выше операций и команда Отмена.

Выбор группы файлов для выполнения операций копирования, перемещения, архивации, удаления и пр. можно выполнить несколькими способами:

- 1) выделение с нажатой левой кнопкой мыши;

- 2) выделение первого и последнего файлов щелчком мыши с нажатой кнопкой SHIFT;
- 3) выделение любых файлов в **Проводнике** при нажатой кнопке CTRL.

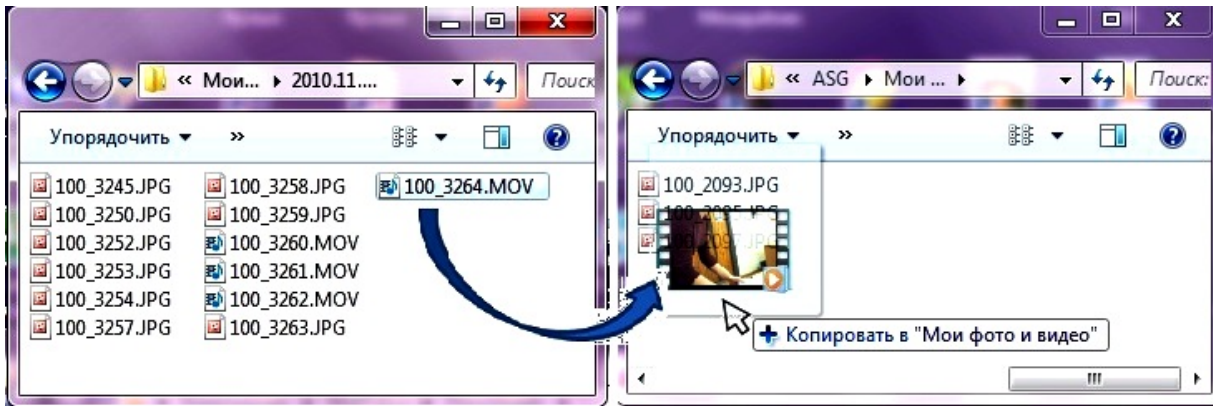


Рисунок 1.11 – Копирование файлов перетаскиванием мышкой

5 Способы запуска программ и открытия файлов данных

Запуск программ (приложений Windows) – операция, которую пользователь выполняет множество раз при работе на компьютере. Установленных в системе программ для работы с различными видами информации (с текстовой и числовой, с базами данных и информационными системами, с графической и конструкторской, с видео и Интернетом...) может быть несколько сотен. Поэтому нужно уметь пользоваться способами, которые позволяют наиболее быстро найти и запустить необходимую программу или открыть файл данных в программе по умолчанию для данного типа информации.

Способов запуска программ в операционной системе Windows достаточно много. Для этого можно использовать различные элементы **Рабочего стола**, о которых говорилось в предыдущем разделе:

- Меню кнопки **Пуск** (см. ранее рисунок 1.2) содержит:
 - на левой панели список наиболее часто используемых программ, для многих из них есть список переходов со ссылками на последние использованные файлы данных, и внизу кнопку **Все программы**, которая покажет на этой панели полный список установленных программ;
 - на правой панели список библиотек, Избранное, Недавние документы, команда **Выполнить**, которые также можно использовать для поиска данных и запуска программ;
- Ярлыки программ на **Рабочем столе**;
- Ярлыки на **Панели задач**, для многих из них могут существовать **Списки переходов**, а также файлы исполняемых программ или данных в Проводнике.

Обобщая, можно говорить о следующих основных способах:

- 1) Запуск программы с использованием ссылок на нее (в меню пуск, в ярлыках), в том числе с выбором файлов данных в **Списках переходов** ярлыков программ (см. ранее рисунок 1.8).
- 2) Открытие файла данных с использованием программы, заданной по умолчанию для этого типа данных, или с выбором программы с использованием команды Открыть с помощью (рисунок 1.12).
- 3) Запуск программы с использованием найденного исполняемого файла или с написанием его имени в окне команды Выполнить.

Для 3-го варианта можно использовать, например, имена программ Notepad, WordPad, Calc, Mspaint, Explorer, Iexplore, Winword, Excel и пр.

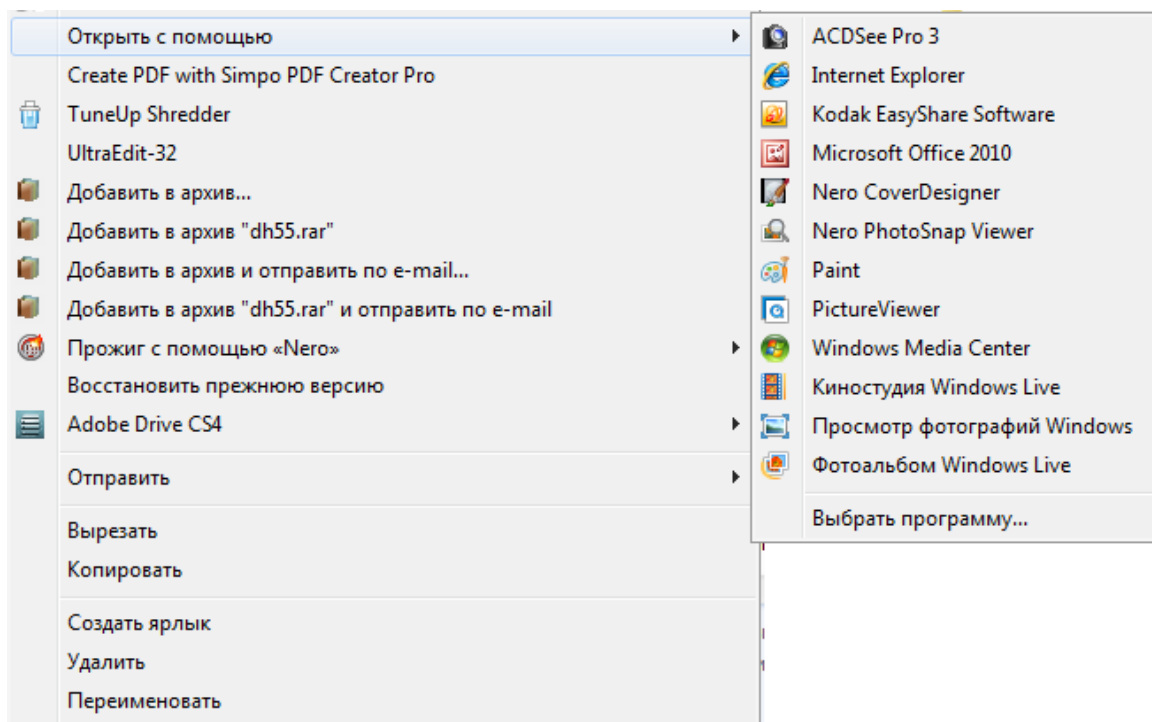


Рисунок 1.12 – Использование команды Открыть с помощью

Учебное задание

- 1) Войти в систему с использованием имени пользователя и пароля, выданных преподавателем.
- 2) Изучить состав папок профиля пользователя, скопировать с экрана окно этой папки и вставить в отчет о работе.
- 3) Закрепить на Панели задач значок Проводника.
- 4) Показать на Рабочем столе стандартные ярлыки Компьютер, Файлы пользователя и Сеть.

- 5) Установить гаджеты Часы, Календарь, Индикатор ЦП, прочие удалить с Рабочего стола.
- 6) Создать в папке Мои документы профиля пользователя папку с именем – *Ваша фамилия и инициалы* (например, Иванов П.С.).
- 7) В созданной папке создать новую папку с именем Архив.
- 8) Создать в своей папке с использованием контекстного меню пустые файлы следующих типов: текстовый документ, точечный рисунок, документ Word, лист Excel, база данных Access. Имя всех файлов – *Ваша фамилия*.
- 9) Выполнить копирование всех файлов разными способами в папку *Архив*.
- 10) Выполнить архивацию файлов в папке Архив командой контекстного меню Отправить → Сжатая Zip-папка.
- 11) Переименовать файл документа Word в РГР-*<Ваша Фамилия>*, базы данных в БД-*<Ваша Фамилия>*.

Лабораторная работа № 2. Использование стандартных программ операционной системы Windows 7

Цель работы

Изучение возможностей стандартных программ для работы с текстовой, числовой, изобразительной, мультимедийной информацией, с информацией Интернета. Выполнение архивации, записи лазерных дисков, обслуживания дисков средствами Windows.

Учебные вопросы

1. Записки, текстовые редакторы Блокнот и WordPad
2. Калькулятор, его режимы работы
3. Графический редактор Paint
4. Проигрыватель Windows Media
5. Браузер Internet Explorer
6. Запись информации на лазерные диски
7. Служебные программы проверки, дефрагментации и очистки дисков

Краткое изложение учебного материала по теме

1 Записки, текстовые редакторы Блокнот и WordPad

Записки (стикеры, Sticky Notes, файл stikynot.exe) – новое стандартное приложение, которое позволяет создавать небольшие текстовые записи в отдельных окнах, размещаемых на Рабочем столе. Подобно бумажным стикерам, **Записки** Windows 7 могут использоваться как листочки для быстрых, коротких записей. **Записки** поддерживают ввод текста с обычной клавиатуры, а также рукописный ввод при наличии сенсорных устройств ввода. При работе с фрагментами текста можно использовать обычные горячие клавиши (см. Приложение 1), а также множество других горячих клавиш, показанных на рисунке 2.1.

Закрывать **Записки** можно нажатием горячих клавиш *ALT+F4*, при этом их содержимое сохраняется в файле *StickyNotes.snt*, расположенном в профиле пользователя. Чтобы удалить **Записку**, достаточно нажать **X** в её правом верхнем углу или *Ctrl+D*. Свернуть **Записки** можно, дважды щелкнув по значку программы на панели задач. При выходе из Windows текст **Записок** сохраняется. Вывести на принтер текст прямо из Записки нельзя.

Записки автоматически распознают Интернет-адреса после введения *http://* или *www*. Ссылки в записках традиционно отображаются синим цветом и подчеркиваются. Чтобы открыть веб-страницу, нужно щелкнуть по ссылке левой кнопкой мыши, удерживая клавишу *Ctrl*.

В **Записках** используется шрифт *Segoe Print* и изменить его нельзя. Однако если вставить в записку текст из любого текстового редактора, то исходные параметры шрифта и форматирование текста сохраняются.



Рисунок 2.1 – Пример использования **Записок** Windows 7

Простейший текстовый редактор Блокнот (notepad.exe) показан на рисунке 2.2.

В **Блокноте** можно задать только параметры страницы для печати документа на принтере (размер бумаги, ориентация, поля, колонтитулы) и шрифт всего документа, которым будет показан текст в окне **Блокнота**, и которым он будет напечатан.

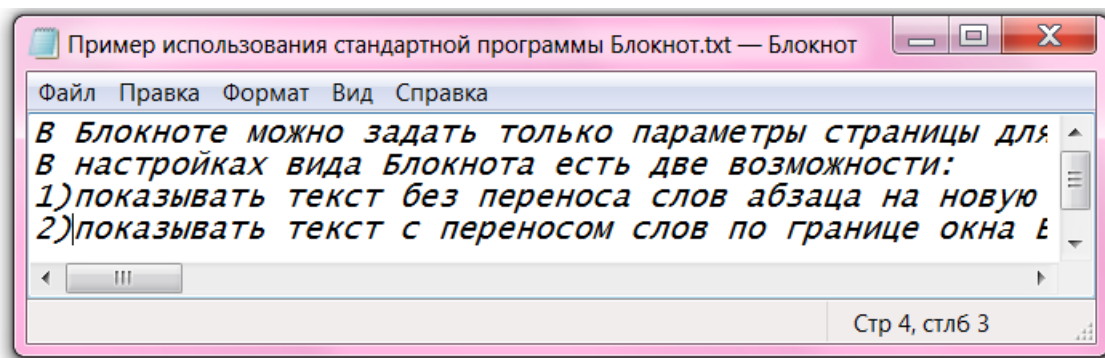


Рисунок 2.2 – Текстовый редактор Блокнот

В настройках вида **Блокнота** есть две возможности:

- 1) показывать текст без переноса слов абзаца на новую строку при достижении границы окна (в меню **Формат** – **Перенос по словам**), при этом можно в пункте меню **Вид** задать параметр **Строка состояния** (показывает в нижней части экрана позицию курсора в тексте – номер строки и символа в ней); в этом случае, как и обычно в текстовом редакторе;
- 2) показывать текст с переносом слов по границе окна **Блокнота**, в этом случае **Строку состояния** показать нельзя.

В том и в другом случае при печати на бумаге разбивка текста на строки в абзаце будет выполняться с учетом заданных параметров страницы.

Текстовый редактор **WordPad** имеет намного больше возможностей, чем **Блокнот** (рисунок 2.3).

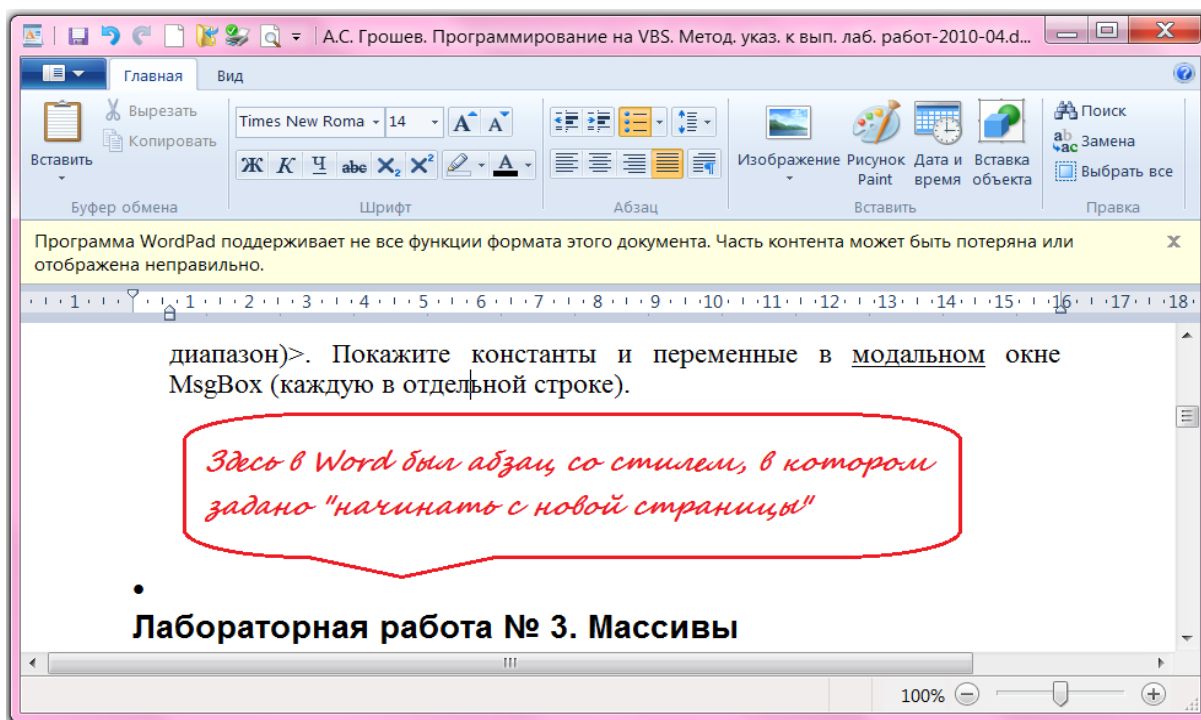


Рисунок 2.3 – Текстовый редактор WordPad

WordPad может работать с форматами RTF, Office Open XML (*.docx) и простым текстом в кодировках Windows, MS-DOS, Юникод и UTF-8. При загрузке в эту программу документа Microsoft Office Word могут возникнуть некоторые искажения при отображении шрифтов, рисунков и других внедренных объектов (см. рисунок 2.3).

В **WordPad** можно выполнять верстку сложных текстовых документов с заданием параметров шрифта, абзаца, с использованием нумерованных и маркированных одноуровневых списков, вставлять в текст картинки файлов графических форматов, рисунок **Paint** (запускается из **WordPad**, нарисованная картинка появляется в **WordPad**'е после закрытия программы **Paint**). Можно вставить текущую дату в выбранном Вами формате и объекты, имеющиеся в операционной системе. Если на компьютере установлена система **Microsoft Office**, становятся доступны объекты **Equation 3.0** для работы с формулами и **Graph** для работы с диаграммами, а также вставка в **WordPad** документов, электронных таблиц и презентаций **Office**.

2 Калькулятор, его режимы работы

Калькулятор Windows 7 имеет несколько режимов работы:

- Обычный
- Инженерный
- Программист
- Статистика,

в каждом из этих режимов можно использовать дополнительно вид калькулятора с журналом вычислений, с панелями для преобразования единиц, операций с датами и дополнительными листами расчетов (ипотека, автолизинг, экономия топлива).

Самый простой вид – *Обычный* (рисунок 2.4) можно использовать для выполнения операций сложения, вычитания, умножения и деления, при этом также есть возможность работать с памятью (кнопки MC – очистить память; MR – вывести на экран содержимое памяти; MS – занести в память содержимое индикатора, M+ – прибавить значение, отображенное на индикаторе, к содержимому памяти; M- – вычесть значение, отображенное на индикаторе, из содержимого памяти), можно вычислить корень квадратный, процент и 1/x. При использовании журнала можно вернуться к

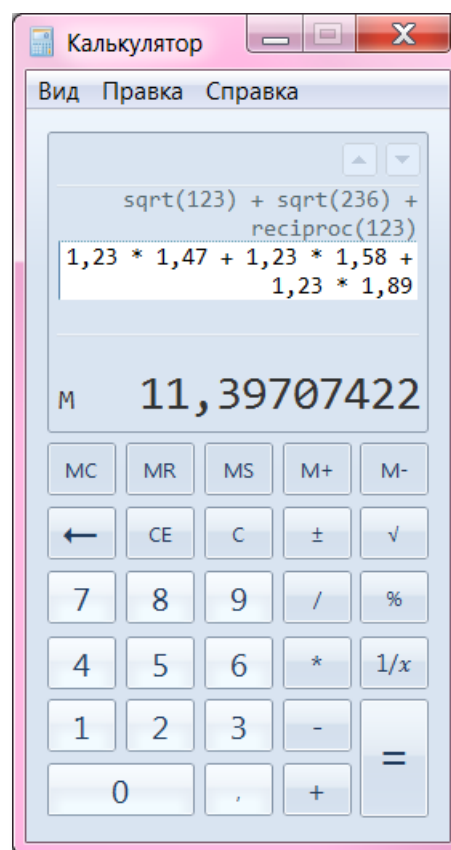


Рисунок 2.4 – Калькулятор, режим Обычный с Журналом

ранее выполненным вычислениям и внести в них исправления.

Математическое выражение, выделенное в журнале, или число, выбранное в индикаторе **Калькулятора**, можно скопировать в буфер обмена Windows (например, нажатием CTRL+C) для его последующего использования в других программах.

Большими возможностями обладает режим **Калькулятора Инженерный** (рисунок 2.5). Здесь есть тригонометрические функции, логарифмы, число Пи, возведение в любую степень, операции со скобками и прочее.

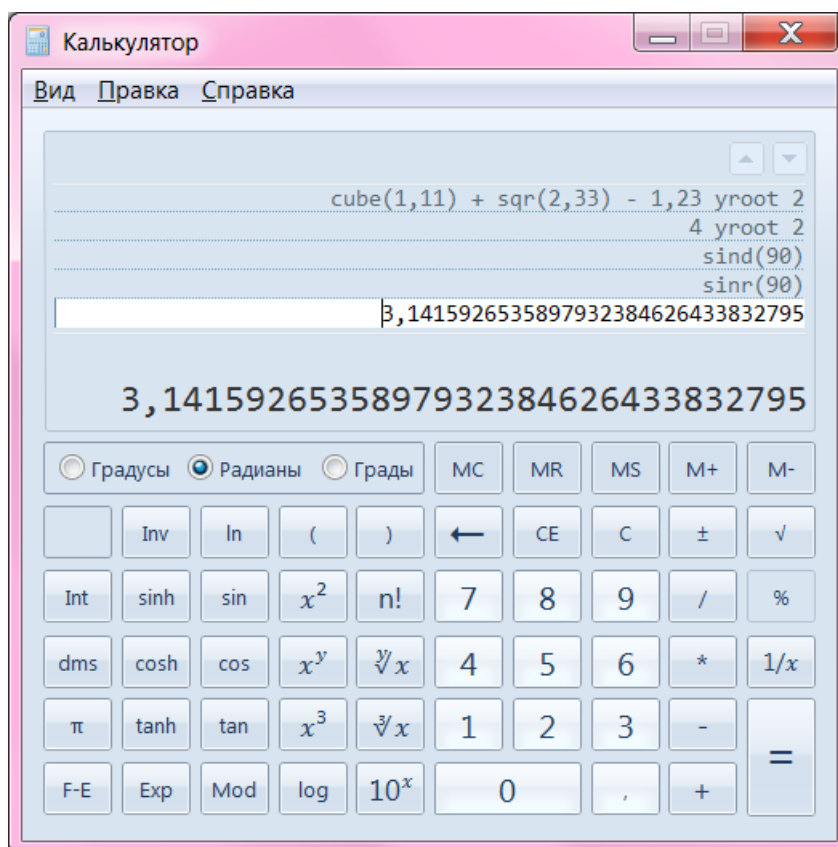


Рисунок 2.5 – Калькулятор, режим Инженерный с Журналом

В режиме *Программист* можно выполнять вычисления с целыми числами в шестнадцатеричной (Hex), десятичной (Dec), восьмеричной (Oct) и двоичной (Bin) системах счисления, при переключении систем число на индикаторе будет преобразовываться из одной системы в другую.

В режиме *Статистика*, который может использоваться для обработки результатов наблюдений (расчет среднеквадратического отклонения и пр.) **Калькулятор** может быть полезен инженерным и научным работникам, хотя для этих целей лучше использовать электронные таблицы.

3 Графический редактор Paint

Графический редактор **Paint** позволяет создавать растровые изображения с сохранением их в форматах PNG, JPEG, BMP, GIF, TIFF.

Для работы могут использоваться 9 различных видов кистей, 23 вида

фигур, 6 видов заливки фигур, новый режим работы с текстом с возможностью задания параметров шрифта для каждой буквы (см. примеры на рисунке 2.6)

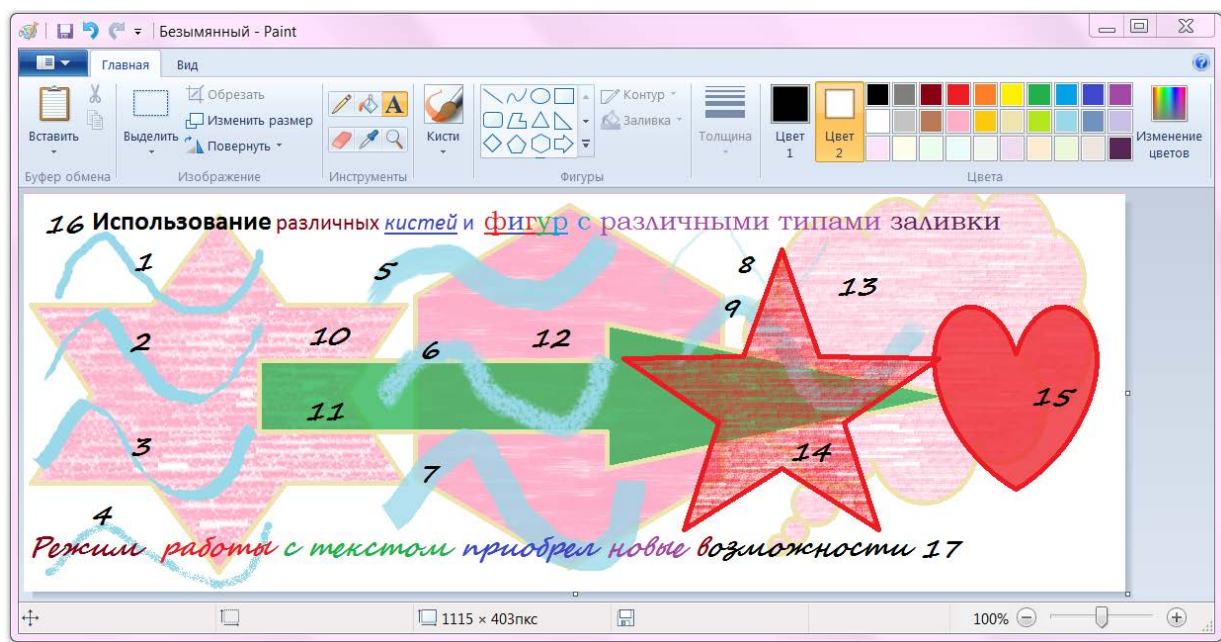


Рисунок 2.6 – Графический редактор **Paint** (1 – кисть, 2 – каллиграфическая кисть 1, 3 – каллиграфическая кисть 2, 4 – распылитель, 5 – кисть для масла, 6 – пастель, 7 – маркер, 8 – текстурный карандаш, 9 – кисть для акварели; 10 – шестиконечная звезда с заливкой *пастель*, 11 – стрелка вправо с заливкой *маркер*, 12 – шестиугольник с заливкой *масло*, 13 – выноска-облако с заливкой *текстурный карандаш*, 14 – пятиконечная звезда с заливкой *акварель*, 15 – сердце с заливкой *маркер*; 16 и 17 – текст с различными параметрами шрифта)

4 Проигрыватель Windows Media

Проигрыватель Windows Media (Windows Media Player, WMP, wmpplayer.exe) – универсальный проигрыватель, который предназначен для прослушивания аудиофайлов и просмотра видеофайлов большинства популярных форматов (рисунок 2.7).

Проигрыватель Windows Media можно также использовать для просмотра файлов изображений (JPG, BMP, GIF, TIFF), создания коллекций мультимедиа, записи компакт-дисков, копирования музыки с компакт-дисков, синхронизации мультимедийных файлов с переносными устройствами, для покупки цифрового содержимого мультимедиа в интернет-магазинах.

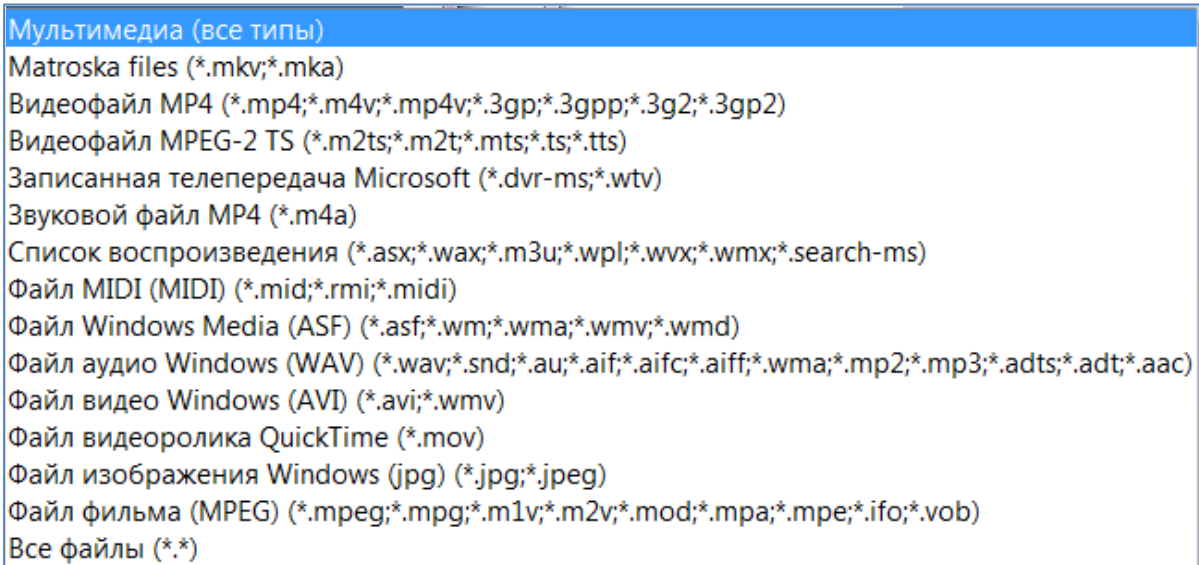


Рисунок 2.7 – Форматы файлов в диалоге WMP (*Файл – Открыть*)

При воспроизведении файлов **Проигрыватель** может иметь *Режим обложки* или *Режим текущего списка воспроизведения* (рисунок 2.8).



Рисунок 2.8 – *Режим текущего списка воспроизведения* (показ фильма)

При работе с библиотеками **Проигрыватель** имеет вид, показанный на рисунке 2.9.

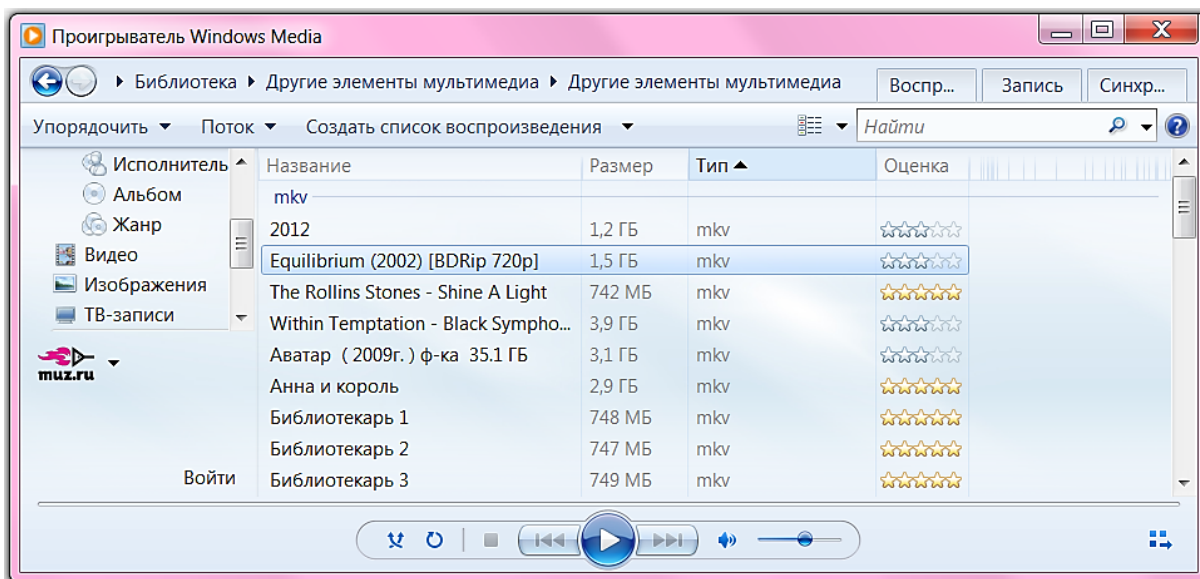


Рисунок 2.9 – Режим библиотеки

Пункт меню *Параметры* кнопки *Упорядочить* открывает диалоговое окно с множеством настроек работы **Проигрывателя**.

При работе с **Проводником**, содержащим файлы мультимедиа, кроме открытия таких файлов в **Проигрывателе** двойным щелчком можно использовать контекстную кнопку, появляющуюся на панели **Проводника**, *Воспроизвести все* или *Воспроизвести выделенное*.

5 Браузер Internet Explorer

Internet Explorer – программа корпорации Microsoft для работы с Интернетом. Его 9-я версия выглядит, как показано на рисунке 2.10.

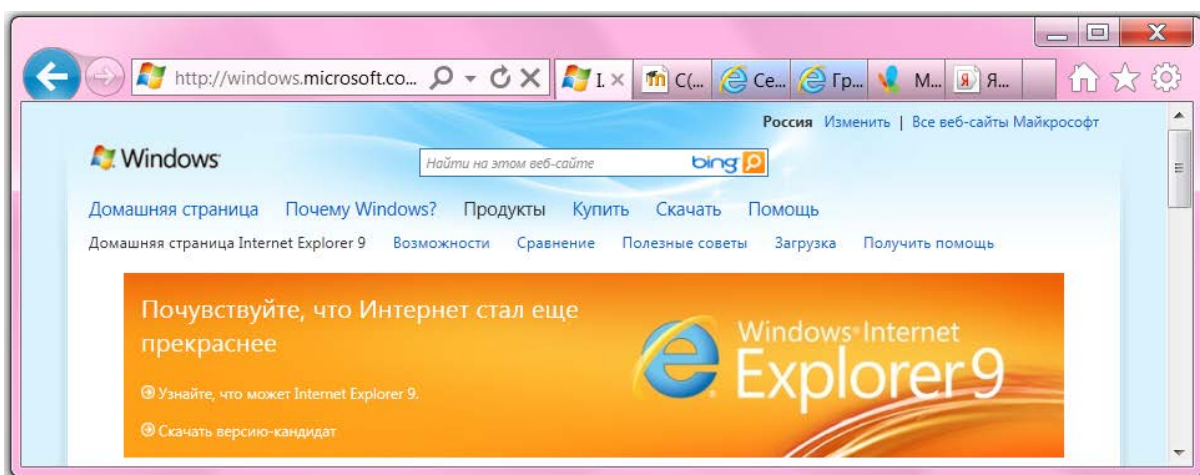


Рисунок 2.10 – Internet Explorer 9

Интерфейс браузера в новой версии упрощен, в верхней части окна присутствует многофункциональная адресная строка, ярлычки открытых страниц и новой страницы, справа 3 кнопки – *Домой*, *Избранное* и *Сервис*. По лаконичности интерфейса **Internet Explorer 9** в новой версии стал подо-

бен браузеру Google Chrome, однако по скорости работы несколько уступает последнему.

В адресной строке браузера, кроме написания в ней ссылки на сайт, можно написать строку поиска и нажать клавишу Enter, или выбрать в появившейся ниже панельке поставщика поиска (справа кнопка добавления поставщиков), как показано на рисунке 2.11. После этого откроется страница поискового сервера Google или другого.

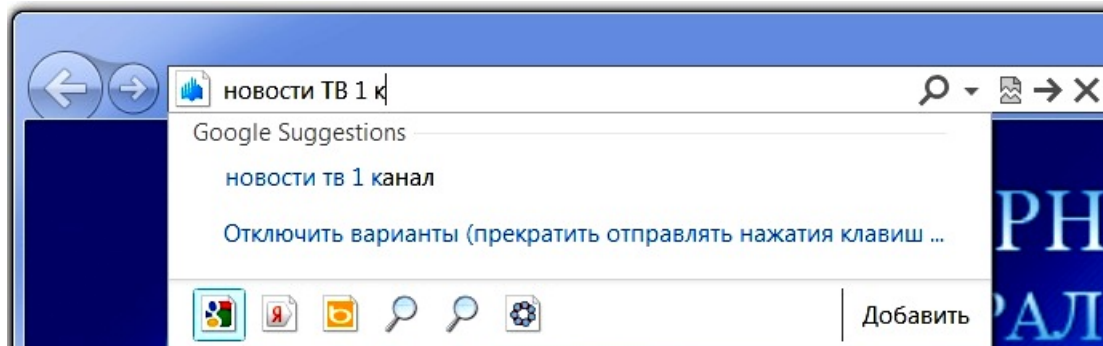


Рисунок 2.11 – Строка поиска в адресной строке браузера

При нажатии на стрелочку ▼ (справа от символа поиска) раскрывается список последних адресов и адресов в журнале, количество строк в котором можно увеличить, щелкнув на соответствующих кнопочках показа дополнительных результатов (рисунок 2.12).

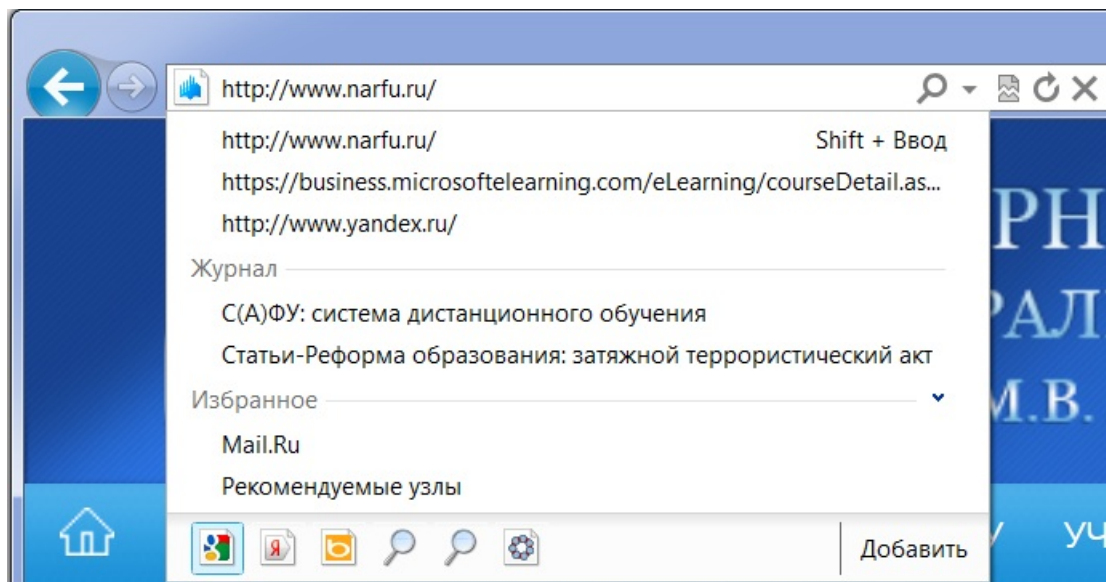



Рисунок 2.12 – Список адресов, с которыми ранее работали

Другой способ открыть страницы, которые Вы наиболее часто посещаете, – нажать на крайний справа ярлычок *Создать вкладку*, на пустой странице будут показаны миниатюры сайтов с полоской-индикатором их активности, щелчок на одной из них открывает выбранный сайт (рисунок 2.13).

Полезная функция браузера **Internet Explorer** – использование так

называемых ускорителей – команд контекстного меню для выделенных фрагментов текста на web-странице (появляется при щелчке мышкой на значке , рисунок 2.14).

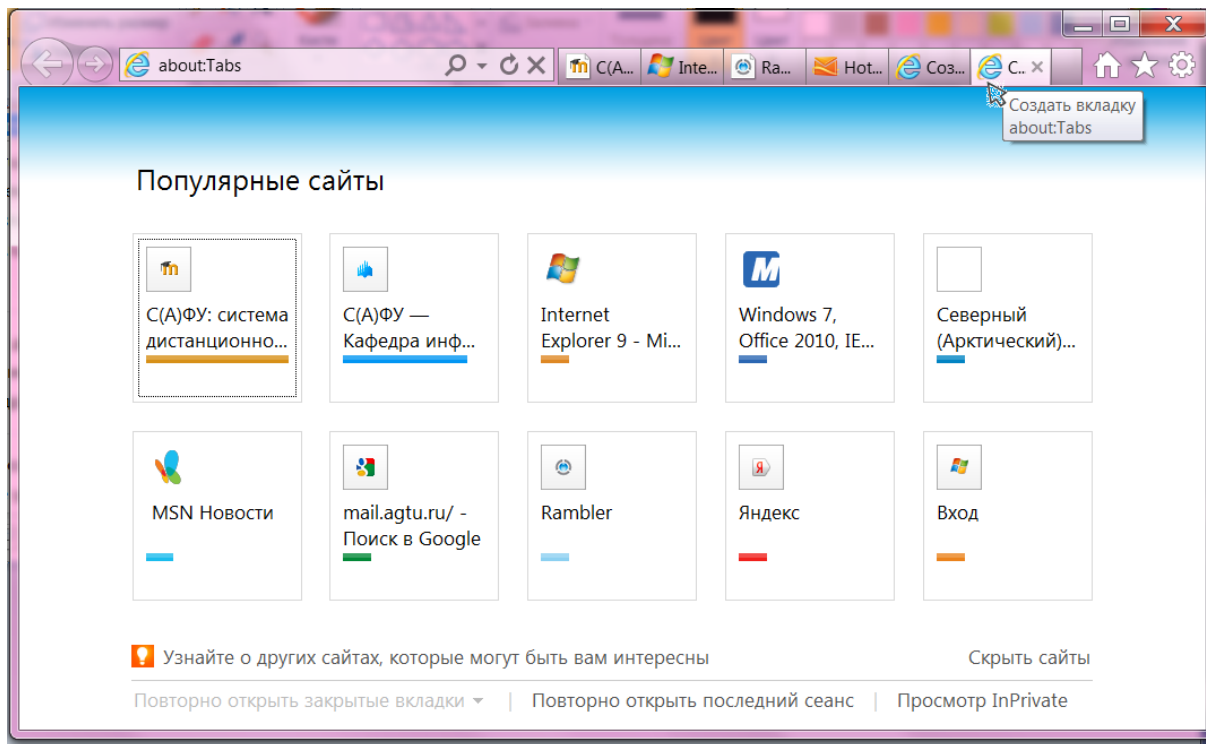


Рисунок 2.13 – Страница браузера после выбора ярлычка *Создать вкладку*

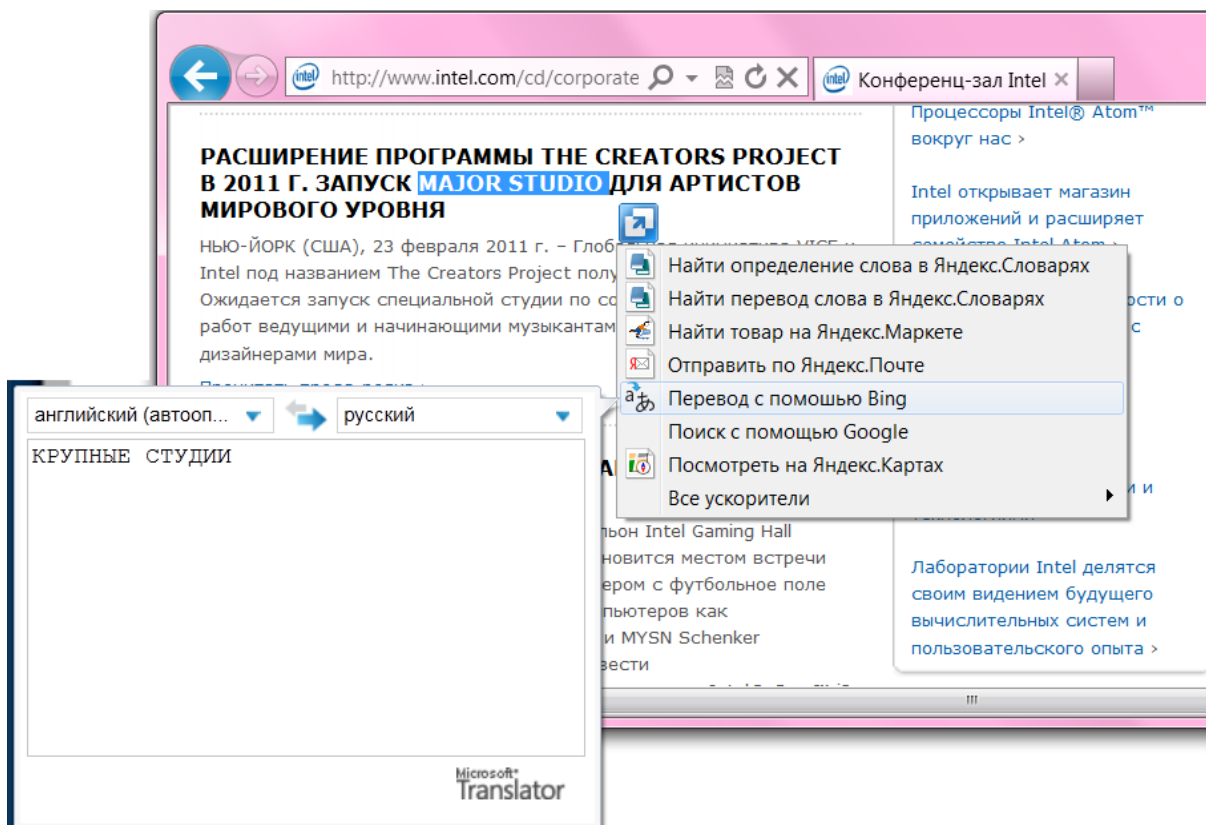


Рисунок 2.14 – Использование *Ускорителей* в браузере

6 Запись информации на лазерные диски

Запись информации на лазерные диски может выполняться средствами Windows, причем в системе можно использовать два формата работы с дисками. Если вставить в дисковод чистый диск, откроется диалоговое окно, показанное на рисунке 2.15. Если автозапуск отключен, для появления этого окна нужно дважды щелкнуть мышкой в Проводнике на значке CD/DVD привода.

В окне предлагается 2 варианта записи диска.

- 1) файловая система **LFS** (Live File System, также обозначается как **UDF**),
- 2) система **Mastered** (также обозначается как **ISO 9660** или **CDFS** – Compact Disc File System).

Формат **LFS** (**UDF**) позволяет копировать на диск и удалять на перезаписываемых дисках каждый файл по отдельности, чего нельзя сделать для формата **Mastered**.

Формат **Mastered** не позволяет записывать на диск файлы с размером более 2 Гб.

LFS (**UDF**) имеет несколько версий: 1.02, 1.50, 2.00, 2.01, 2.5, 2.6. Последние 2 версии предназначена только для Windows Vista и более поздних ОС. Версии 1.02, 1.5 и 2.00 можно использовать для записи видео на диски, используемые на бытовых DVD-плеерах (выбор версии можно делать в программе Nero, Windows форматирует диск в последней версии).

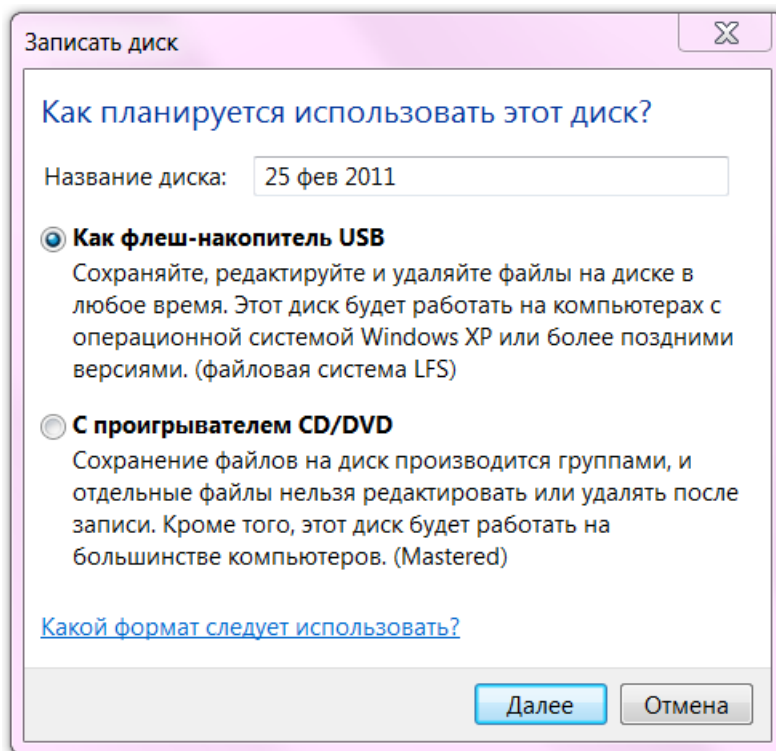


Рисунок 2.15 – Окно выбора режима работы с новым диском

Для записи файлов на диск **LFS (UDF)** достаточно, как обычно, выполнить их копирование в папку диска.

Для записи файлов на диск **Mastered (CDFS)** необходимо вначале выполнить их копирование в папку диска, затем нажать на кнопку *Запись на компакт-диск*, присутствующую в этом случае в **Проводнике**. Можно также не записывать подготовленные к записи файлы, если нажать на кнопку *Удалить временные файлы* или как обычно удалить файлы, которые Вы раздумали записывать (эти файлы будут выделены в Проводнике в группу *Подготовленные для записи на диск файлы*).

7 Служебные программы проверки, дефрагментации и очистки дисков

Служебная программа проверки диска на наличие ошибок файловой системы и на наличие плохих секторов может быть запущена, если открыть из контекстного меню окно свойств диска (рисунок 2.16). Из этого же окна можно запустить программу дефрагментацию диска – процесс оптимизации размещения фрагментов одного файла на диске, все фрагменты собираются в последовательно идущие кластеры. Последняя программа может быть запущена и из меню кнопки Пуск (Стандартные – Служебные).

Для программы проверки диска после ее выбора в окне свойств появляется окно задания параметров проверки, как показано на рисунке 2.16.

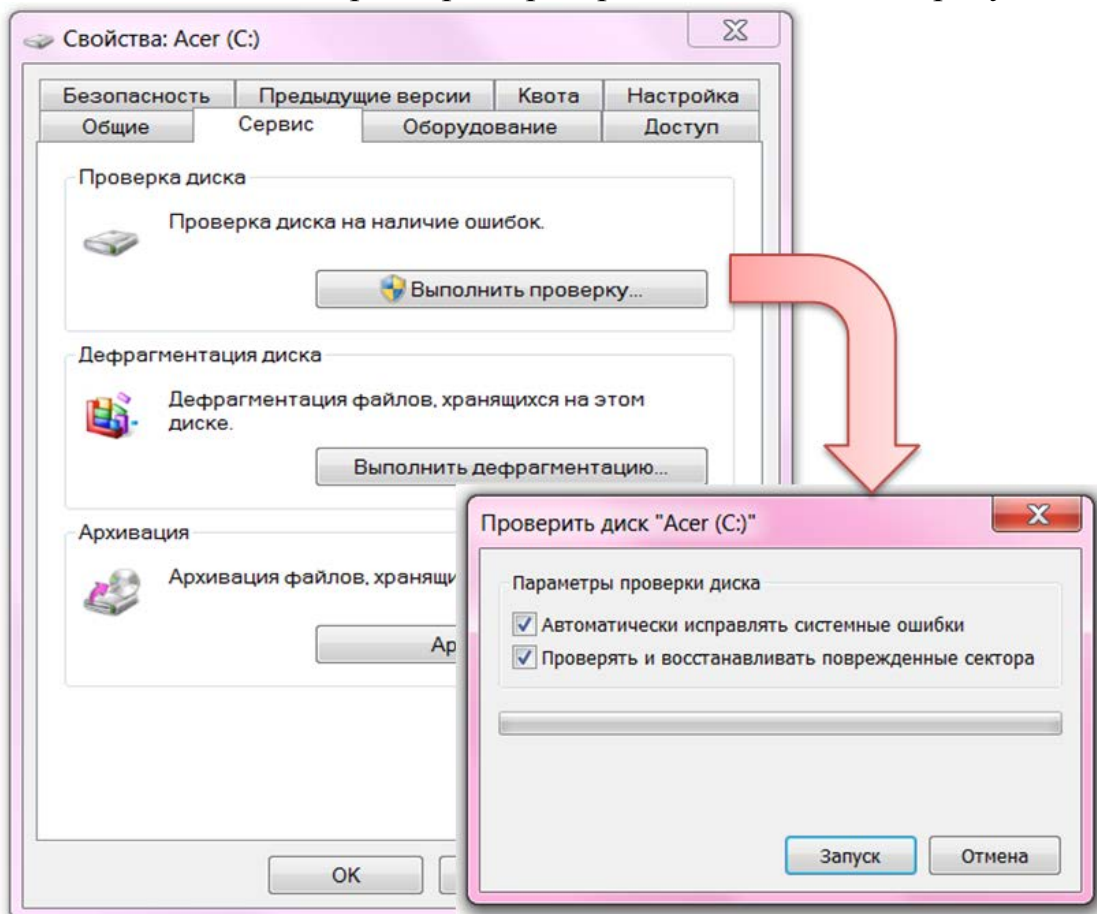


Рисунок 2.16 – Окна свойств диска C: и параметров проверки диска

Программа дефрагментации в Windows 7 запускается автоматически по умолчанию один раз в неделю. Расписание можно настроить или отменить автоматический запуск. Первый этап работы программы – анализ диска, который сообщает процент фрагментации, после чего можно сделать вывод о необходимости выполнения дефрагментации. Если программа ранее запускалась, окно программы сообщает о степени фрагментации дисков.

Программа очистки дисков запускается из меню кнопки Пуск (Стандартные – Служебные). Появляется окно выбора диска для проверки, после чего в новом окне будут показаны результаты анализа (размер временных и прочих файлов – Корзина, автономные веб-страницы, файлы отчетов об ошибках и пр.). Вы можете принять решение об их удалении или нет.

Учебное задание

- 1) Создать несколько **Записок** с текстом, задать их цвет и размер, научиться сворачивать, закрывать и удалять **Записки**.
- 2) В программе **Блокнот** открыть созданный в Лабораторной работе № 1 текстовый файл и записать в него:
 - а) Список стандартных программ операционной системы Windows 7.
 - б) Способы копирования файлов с использованием Проводника.
 - в) Способы выделения всех файлов, группы файлов от 1-го до n-го и произвольного набора файлов в Проводнике.
 - г) Способы запуска программ и открытия файлов данных.
- 3) Открыть в программе **WordPad** созданный в **Блокноте** документ. Переоформить списки пунктов 2.1 – 2.4 в виде нумерованных и маркированных списков. Вставить в текст:
 - а) Картинку проводника с Вашей папкой.
 - б) Нарисовать в **WordPad**'е картинку – настольный компьютер.
 - в) Познакомиться, какие еще объекты можно вставить в текст.
 - г) Сохранить файл в формате RTF.
- 4) Запустить **Калькулятор**, выполнить вычисления в режимах Обычный, Инженерный, Статистика. Скопировать все вычисления и их результаты в свой текстовый файл с заголовком «*Расчеты в Калькуляторе*», сохранить файл.
- 5) В графическом редакторе **Paint** продемонстрировать использование различных кистей, фигур с разным типом заливки, текста с разами параметрами шрифта. Сохранить рисунок в различных форматах и сравнить размер полученных файлов и вставить его в текстовый файл.
- 6) Открыть в **Проигрывателе Windows Media** аудио файл, затем видео

файл. Открыть в **Проигрывателе** эквалайзер, посмотреть варианты его стандартных настроек. Открыть из *Дополнительных возможностей* контекстного меню *Эффекты SRS WOW*, включить TruBass и настроить эффект объемного звучания WOW.

- 7) Открыть в браузере **Internet Explorer** сайт университета, найти информацию о своем институте и специальности.
- 8) Изучите разницу между форматами записи оптических дисков **UDF** и **CDFS**.
- 9) Открыть окно свойств одного из дисков и найти программы проверки диска и дефрагментации. Запустить программу очистки диска и посмотреть список предлагаемых к удалению файлов.

Лабораторная работа № 3. Microsoft Word 2010. Настройка стилей. Работа с текстом

Цель работы

Освоение приемов подготовки текстовых документов в соответствии с требованиями стандарта «Правила оформления студенческих работ» или других нормативных документов.

Учебные вопросы

1. Настройка стилей
2. Написание основного текста документа

Краткое изложение учебного материала по теме

Настройка стилей

При оформлении студенческих работ (рефераты, отчеты о лабораторных работах, расчетно-графические работы, курсовые работы и проекты, дипломные работы и диссертации), при подготовке статей для научных журналов и отчетов о научно-исследовательских работах предъявляются достаточно строгие требования к написанию заголовков, основного текста, таблиц, рисунков, формул и пр.

При правильном использовании стилей в системе Microsoft Word легко выполнить одинаковое оформление перечисленных выше элементов во всем документе, а также переоформить документ при необходимости простым изменением параметров используемых стилей.

Стиль – именованный набор параметров оформления абзаца, знака или таблицы.

На вкладке *Главная* ленты Word 2010 присутствует группа кнопок *Стили* (рисунок 3.1) – это так называемый *Список экспресс-стилей*. Этот список можно изменить (добавить или удалить стили).

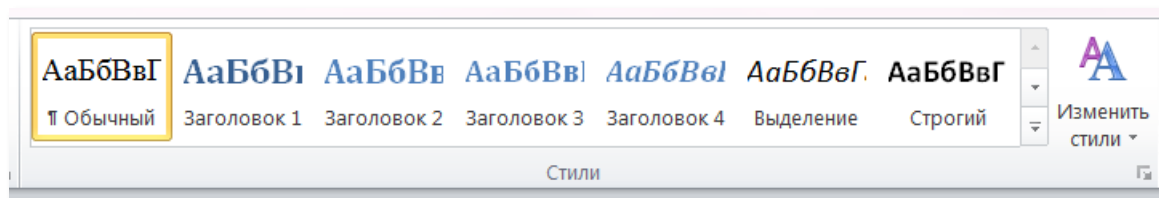



Рисунок 3.1 – Список экспресс-стилей на вкладке *Главная* ленты

Microsoft Word имеет более 100 предварительно настроенных стилей (рисунок 3.2), которые можно увидеть, если нажать на маленькую кнопку  в правом нижнем углу группы кнопок *Стили*. Следует учитывать, что список стилей в окне *Стили* может быть неполным, т. к. предусмотре-

ны режимы (настраиваемые в ссылке *Параметры...*): *Рекомендованные*, *Используемые*, *В текущем документе*, *Все* (рисунок 3.3). По умолчанию окно *Стили* открывается в режиме *Рекомендованные*, который при настройке стилей следует заменить на *Все*, а после настройки и использования всех необходимых в документе стилей – на *Используемые*.

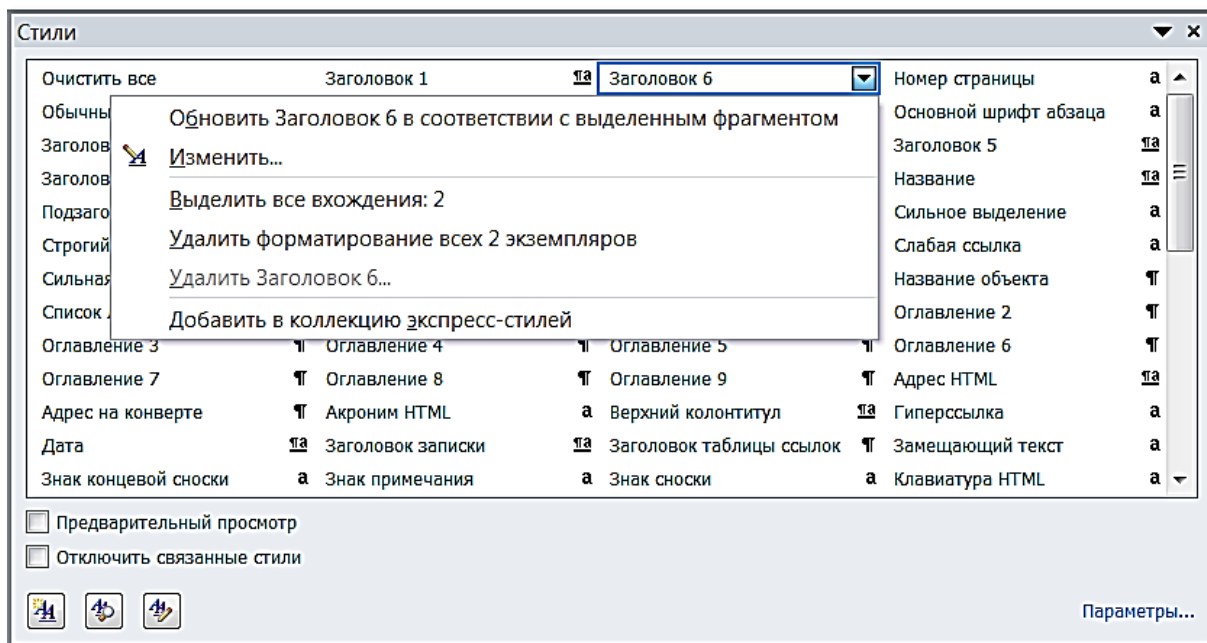


Рисунок 3.2 – Окно *Стили*

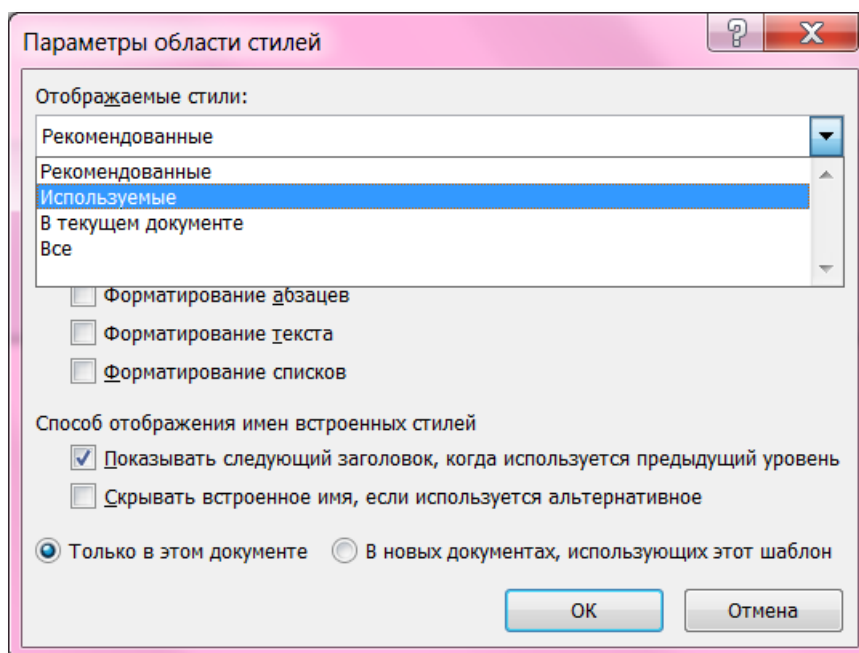


Рисунок 3.3 – Окно *Параметры области стилей*

Любой абзац, написанный в редакторе Word (в том числе рисунки, формулы, текст в таблицах), всегда оформлен каким-либо стилем из их набора.

Базовым является стиль с названием **Обычный**, на нем основано множество других стилей. Если изменить настройки стиля **Обычный**, могут измениться и настройки тех стилей, основой которых он является.

Многие из стилей имеют настройки параметров, не соответствующие действующим в нашей стране стандартам и правилам оформления документов, поэтому их параметры перед использованием нужно переопределить.

Чтобы отредактировать параметры стиля, следует щелкнуть правой кнопкой мыши на названии стиля в *Списке экспресс-стилей* (см. ранее рисунок 3.1) или в окне *Стили* (см. ранее рисунок 3.2) и в открывшемся списке выбрать команду *Изменить*. После чего откроется окно, показанное на рисунке 3.4.

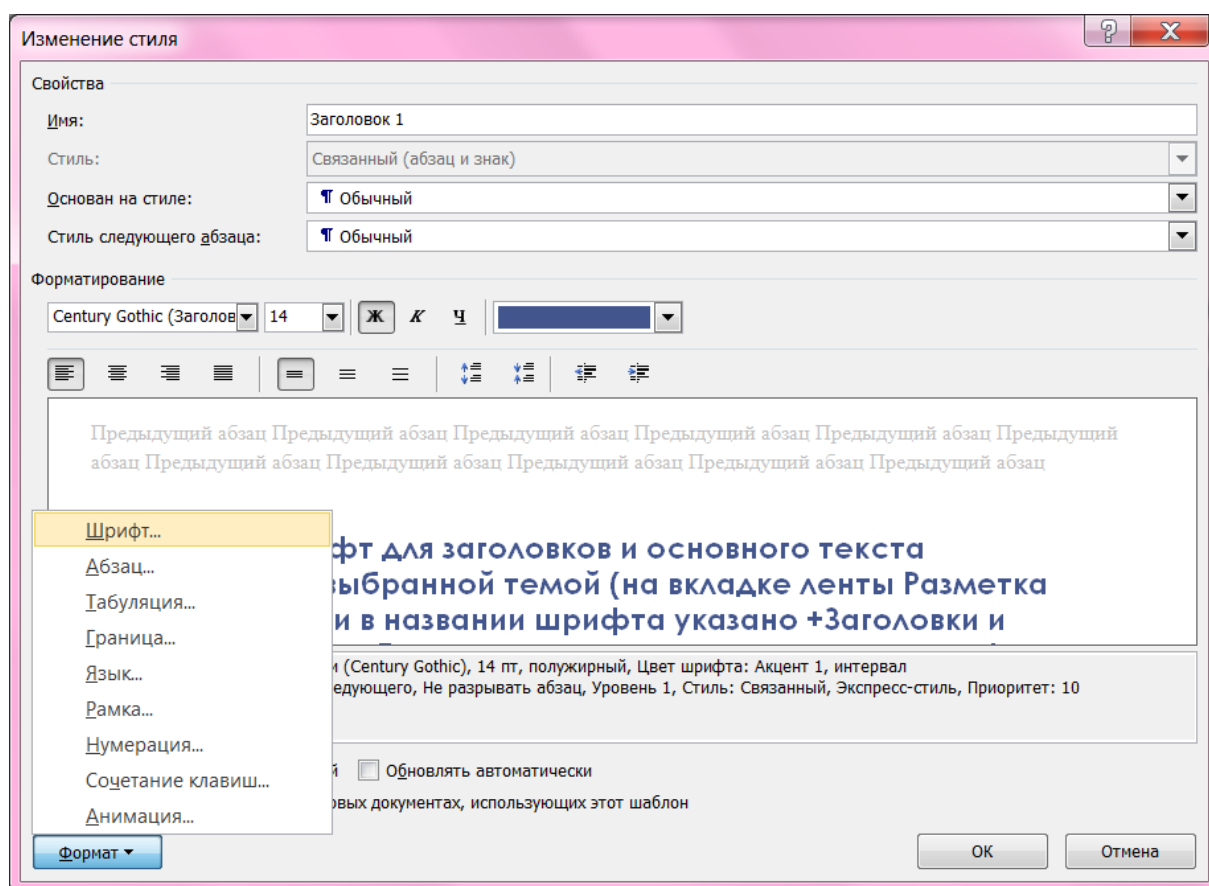


Рисунок 3.4 – Окно настройки стилей

Каждый стиль *Абзаца* имеет имя, название стиля, от которого он является производным и стиля следующего абзаца, пункты меню кнопки *Формат* задают параметры шрифта, абзаца, табуляции, границ, языка, рамки, нумерации, сочетания клавиш и анимации (см. рисунок 3.4).

Цвет и вид шрифт для заголовков и основного текста определяется выбранной темой (на вкладке ленты *Разметка страницы*), если в названии шрифта указано *+Заголовки* и *+Основной текст*. Если задать конкретные названия шрифтов и их цвет, эти параметры не будут изменяться

при изменении темы.

В системе Word 2010 можно создать собственный набор экспресс-стилей для документа или использовать готовые наборы (рисунок 3.5). Все цвета и форматы в пределах одного стандартного набора стилей гармонично сочетаются друг с другом, позволяя создавать профессионально оформленные документы.

Набор экспресс-стилей Word 2010 может включать стили основного текста, нескольких уровней заголовков, цитат, выделений и пр.

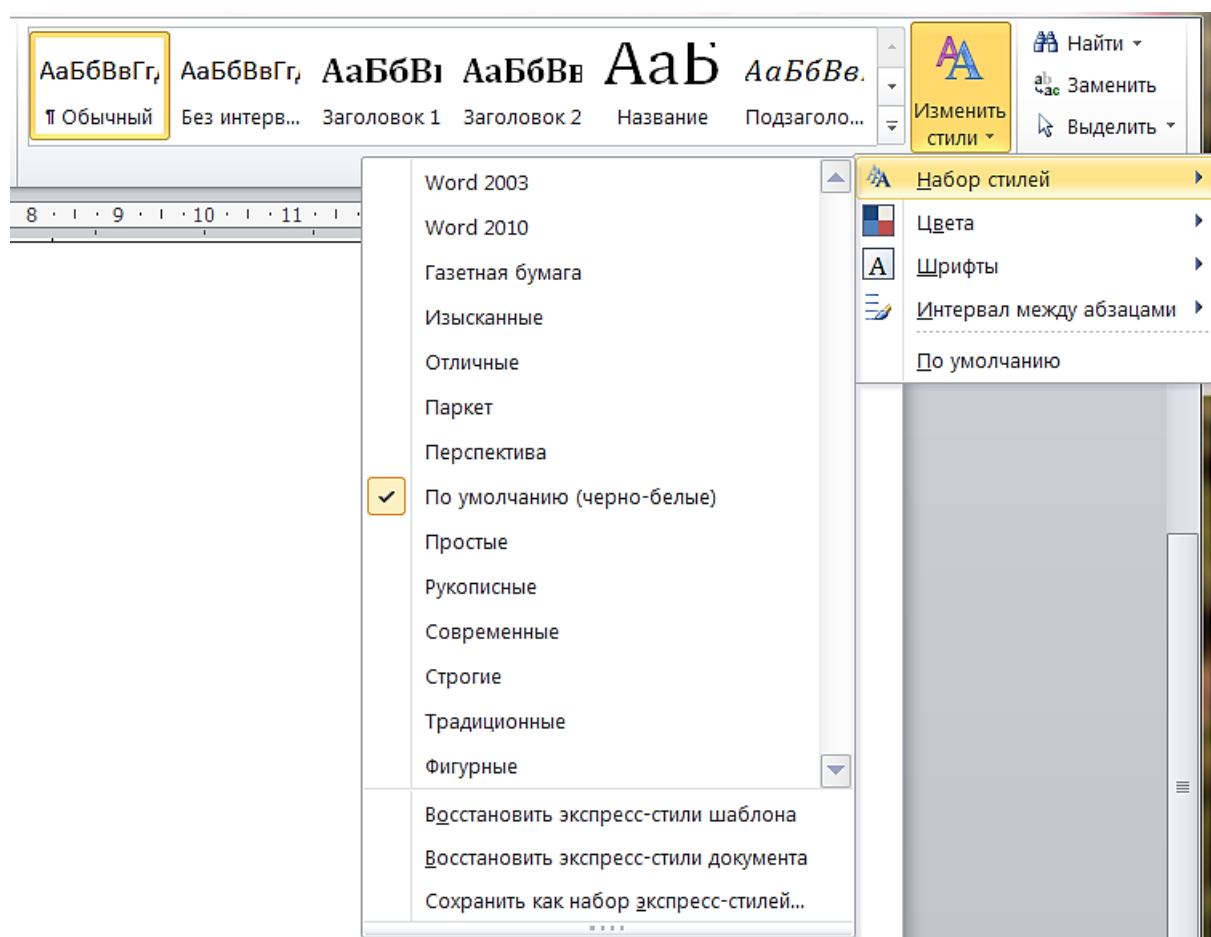


Рисунок 3.5 – Наборы экспресс-стилей Word 2010

Выберем для использования набор стилей *По умолчанию (черно-белые)*, настроим набор и параметры его стилей для оформления студенческих работ, после чего сохраним его, как новый набор с именем *Студенческий*.

Для документа, соответствующего стандарту оформления студенческих работ, можно определить следующие стили, которые следует включить в набор:

- 1) Основной текст
- 2) Заголовок раздела (первого уровня)
- 3) Заголовок подраздела (второго уровня)
- 4) Заголовок пункта (третьего уровня)
- 5) Рисунок

- 6) Подпись рисунка
- 7) Таблица
- 8) Название таблицы.

Итак, откроем СТО «Работы студентов. Общие требования и правила оформления», раздел «Правила оформления пояснительной записки». Здесь (после параметров страницы) написано следующее: «документы выполняются одним из следующих способов: ... – с использованием компьютера и принтера: гарнитура, предпочтительно, Times New Roman; кегль (размер шрифта) – 13; междустрочный интервал 1,5; выравнивание – по ширине; цвет шрифта – чёрный. Абзацы в тексте начинают отступом 12,5 мм. Расстояние между заголовками раздела и подраздела, а также заголовком и текстом – одна пустая строка».

Так как базовым для многих других является стиль *Обычный*, начнем настройку параметров именно с него. Этот стиль будем в дальнейшем использовать для основного текста документов.

Зададим параметры шрифта и абзаца, как показано на рисунках 3.6 и 3.7.

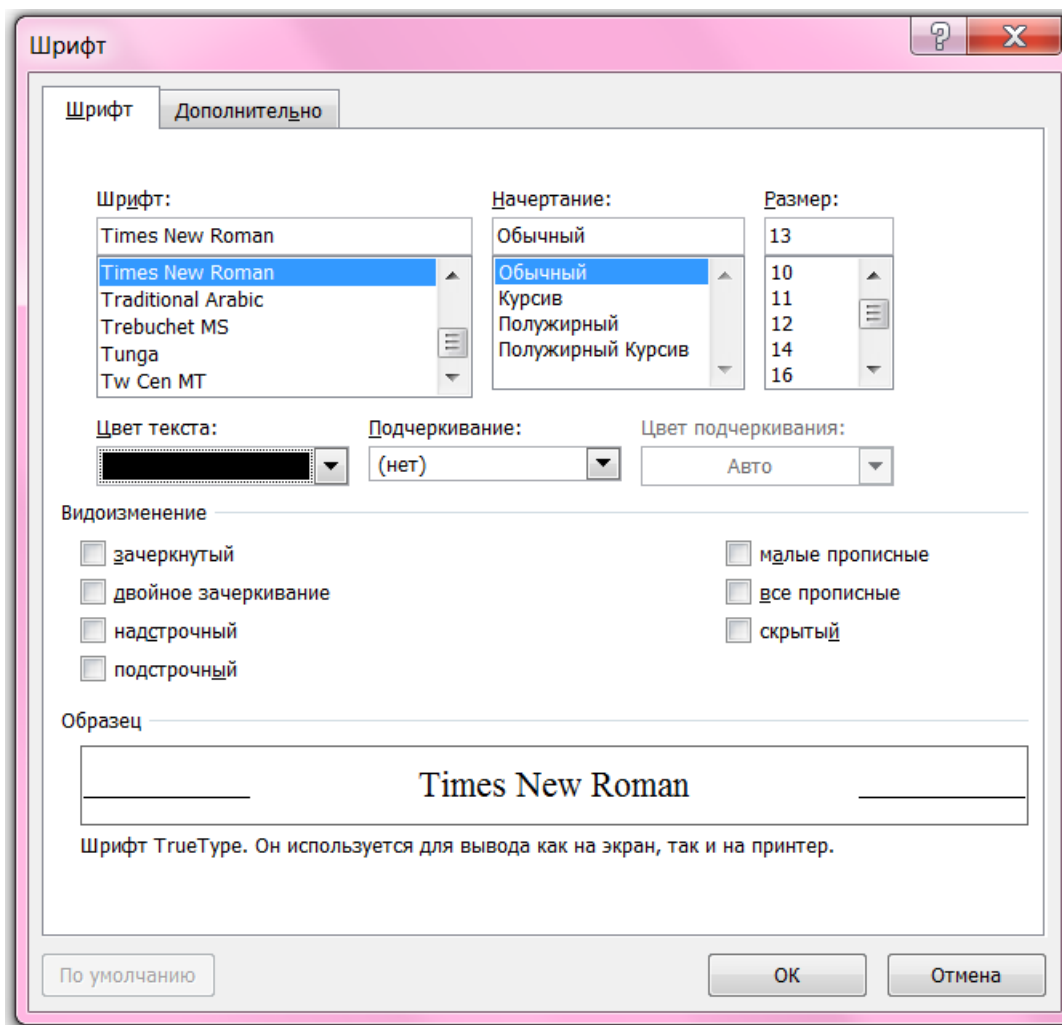


Рисунок 3.6 – Параметры шрифта для стиля *Обычный*

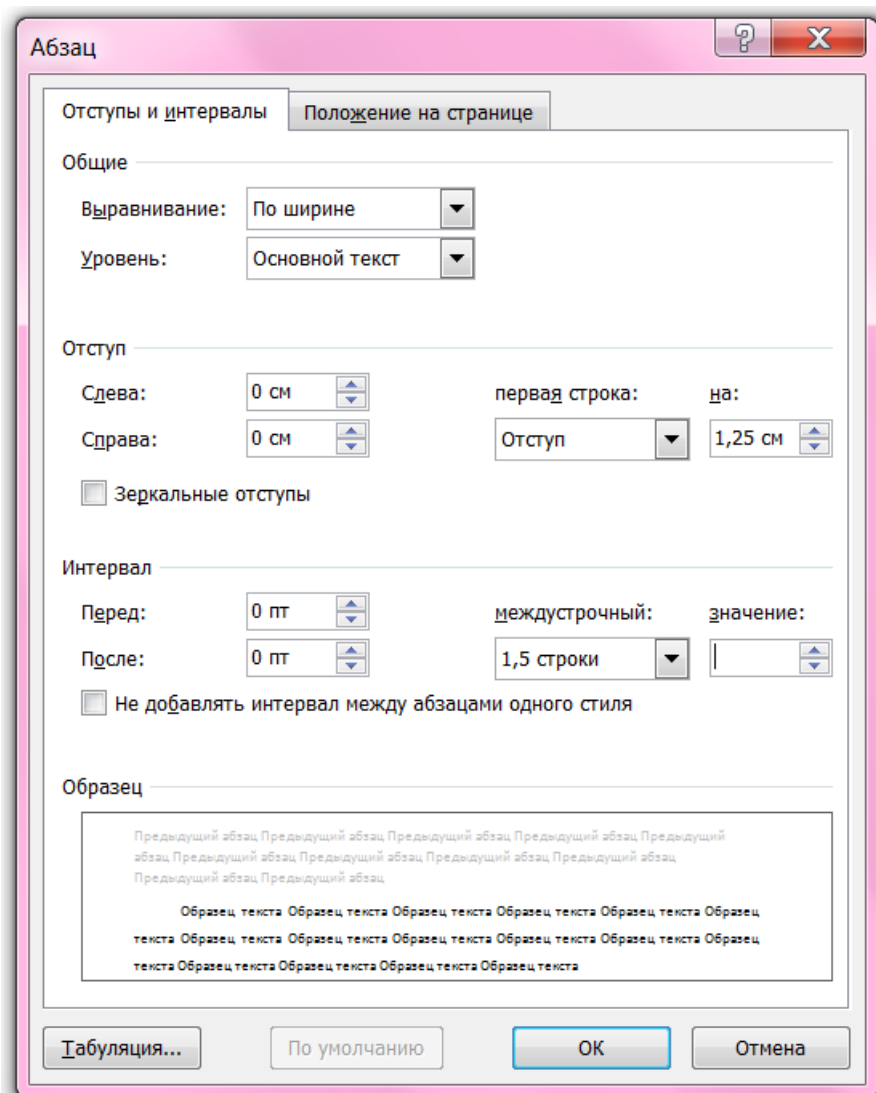


Рисунок 3.7 – Параметры абзаца для стиля *Обычный*

Описание параметров стиля **для основного текста** может выглядеть следующим образом (для отчета о лабораторной работы):

Стиль основного абзаца:

имя стиля – *Обычный*,

шрифт – Times New Roman, обычный, размер 13. цвет – черный;

абзац: выравнивание – по ширине, отступы: слева 0 см, справа 0 см, первой строки 1, 25 см; интервалы: перед 0 пт, после 0 пт, междустрочный 1,5 строки.

Параметры абзаца заголовков и шрифта заголовка раздела будут другими (шрифт заголовков подразделов и пунктов имеет те же параметры, что и основной текст).

В СТО написано следующее: «Заголовки разделов следует оформлять прописными буквами, без разрядки, без подчёркивания. Не допускаются переносы в словах, а также отрыв предлога или союза от относящего-

ся к нему слова». «Максимальная длина текста в строке заголовка раздела должна быть меньше на 10 мм, чем в основном тексте. Вторая и последующие строки заголовка раздела начинаются с абзацного отступа». «Каждый раздел рекомендуется начинать с новой страницы, при этом верхнее поле составляет 60 мм».

Для заголовка раздела выберем стиль *Заголовок 1* и зададим его параметры в соответствии с СТО, как показано на рисунках 3.8 – 3.10.

Для заголовков подразделов и пунктов выберем стили *Заголовок 2* и *Заголовок 3* и настроим их параметры аналогично стилю *Заголовок 1* за исключением следующих пунктов:

- 1) убрать галочку у пункта *все прописные* (см. рисунок 3.8);
- 2) интервал перед абзацем равен 13 пт (см. рисунок 3.9);
- 3) убрать галочку у пункта *с новой страницы* (см. рисунок 3.10).

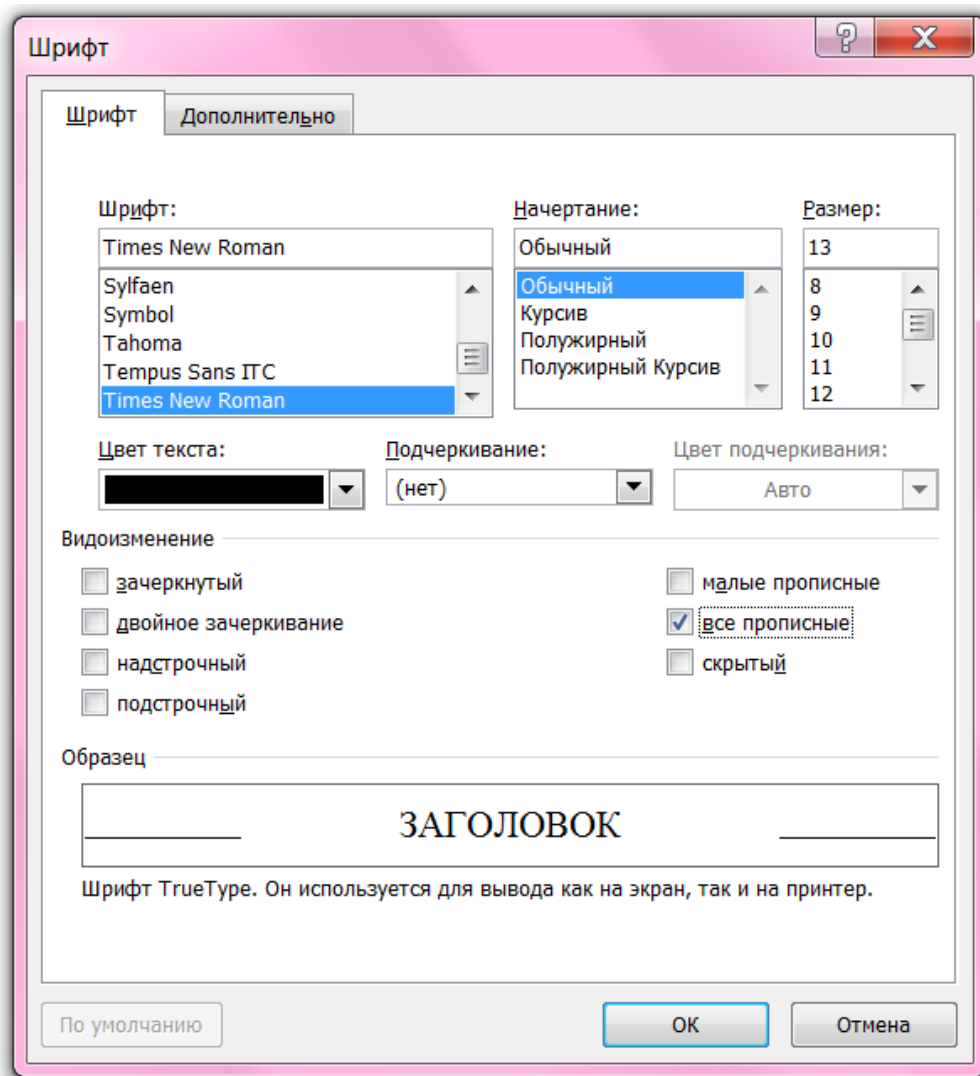


Рисунок 3.8 – Параметры шрифта для стиля *Заголовок 1*

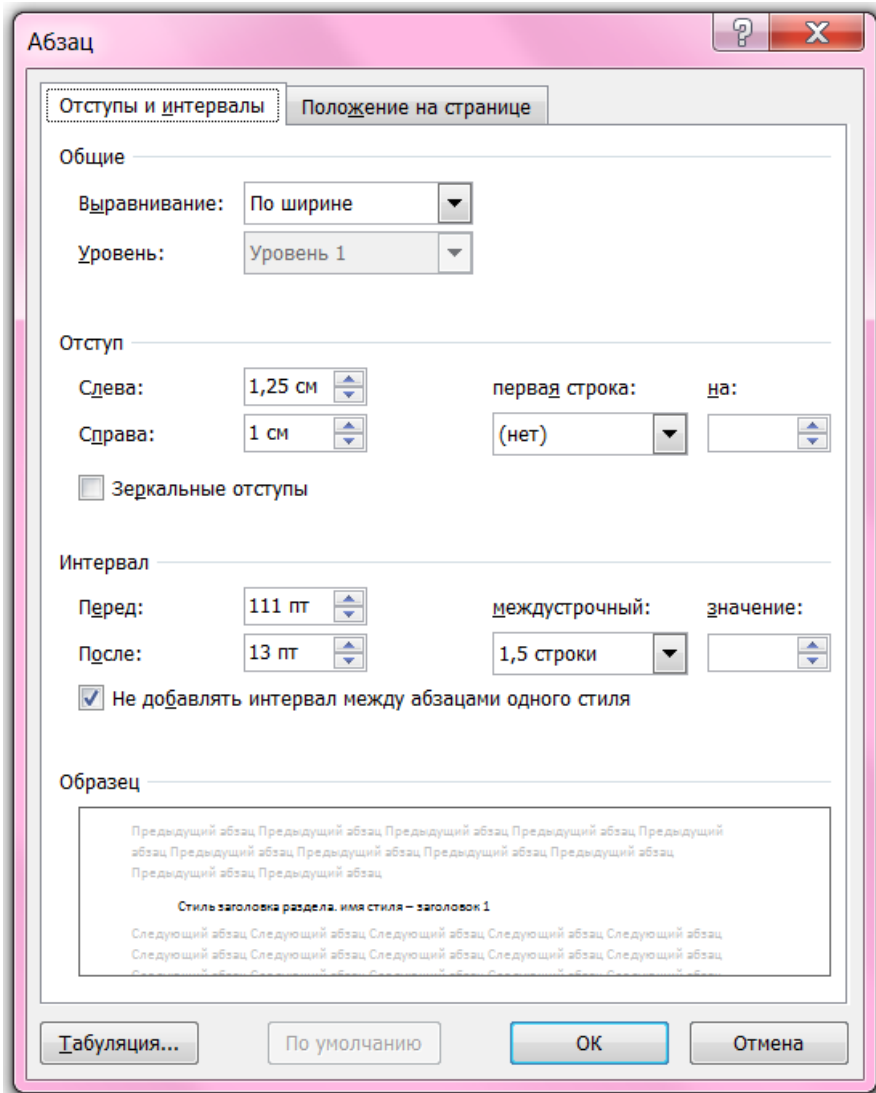


Рисунок 3.9 – Параметры абзаца для стиля *Заголовок 1*. Отступы и интервалы

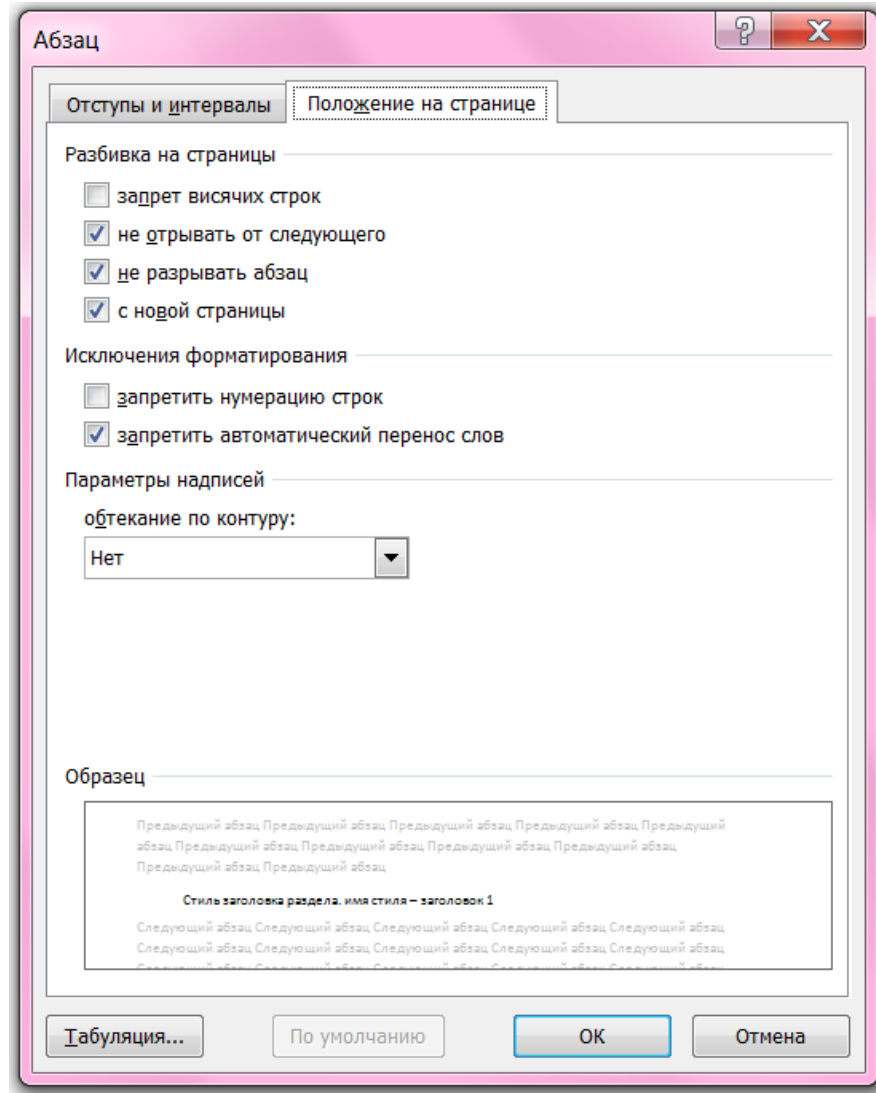



Рисунок 3.10 – Параметры абзаца для стиля *Заголовок 1*. Положение на странице

Описание параметров стиля **для заголовка раздела** будет выглядеть следующим образом:

Стиль **заголовка раздела**:

имя стиля – *Заголовок 1*,
шрифт – Times New Roman, обычный, размер 13. цвет – черный, все прописные буквы;
абзац: выравнивание – по ширине, отступы: слева 1, 25 см, справа 1 см, первой строки 0 см; интервалы: перед 111 пт, после 13 пт, междустрочный 1,5 строки, с новой страницы, запретить автоматический перенос слов.

Для оформления рисунков создадим новый стиль с именем *Рисунок*. Для этого вначале выберем в окне *Стили* (см. рисунок 3.2) стиль *Обычный*, затем щелкнем на кнопке *Создать стиль* . Откроется окно *Создание стиля* (рисунок 3.11).

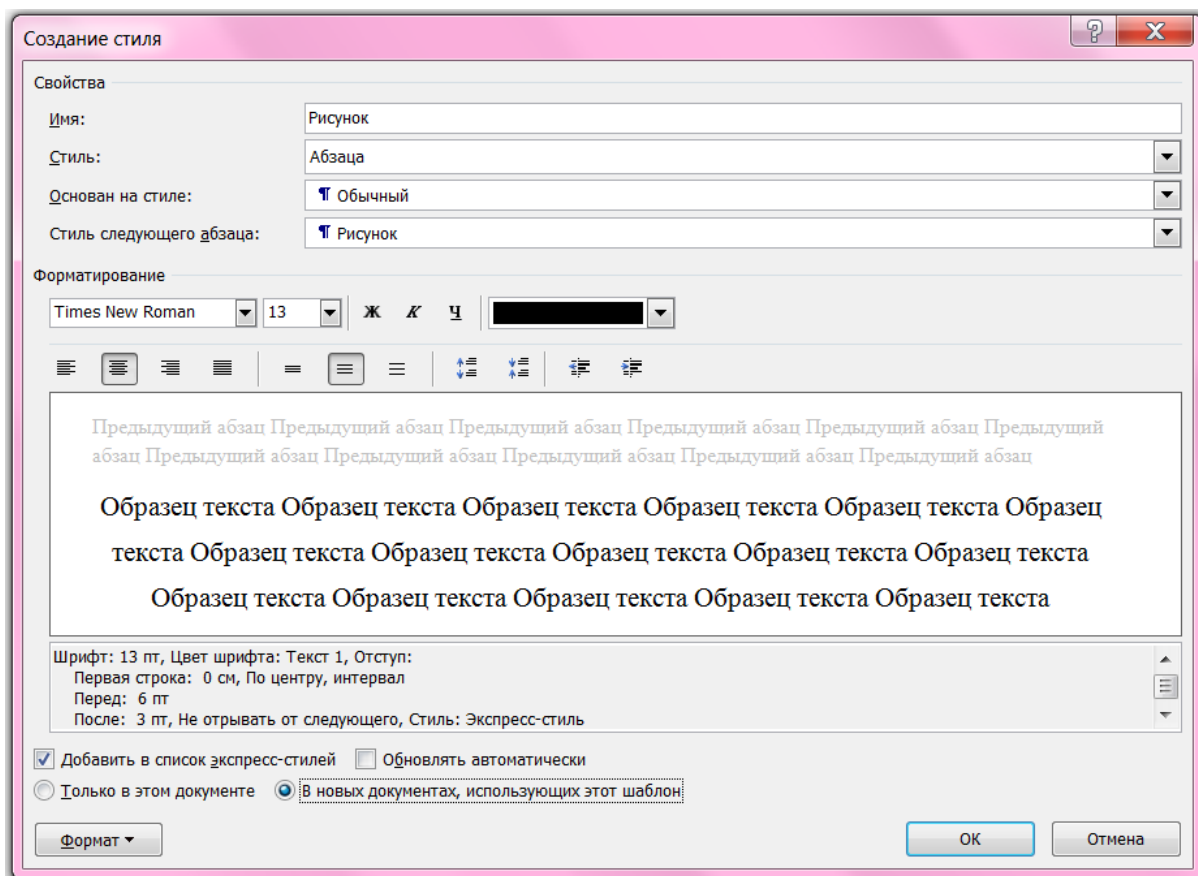


Рисунок 3.11 – Создание стиля *Рисунок*

В соответствии с примерами оформления, приведенными в СТО, рисунки и другие иллюстрации рекомендуется размещать с центрированием по горизонтали, до рисунка следует установить увеличенный интервал, после рисунка следует размещать подрисуночную подпись и название, которые нельзя отрывать от рисунка. Настроим эти параметры для нового сти-

ля с именем *Рисунок*.

Для названий рисунков аналогично предыдущему на основе шрифта *Обычный* создадим стиль с именем *Название рисунка*, для которого в параметрах абзаца зададим отступ первой строки 0, интервал после абзаца 6 и выравнивание по центру.

Далее создадим стили *Таблица* и *Название таблицы*. Для стиля *Таблица* следует задать межстрочный интервал одинарный и отступ первой строки 0. Для *Названия таблицы* также нужно задать отступ первой строки 0, выравнивание – по левому краю, интервал перед абзацем – 6, не отрывая от следующего. Для стиля *Таблица* следует задать следующий стиль – *Название таблицы*.

После редактирования списка экспресс-стилей группа кнопок *Стили* будет выглядеть, как показано на рисунке 3.12.

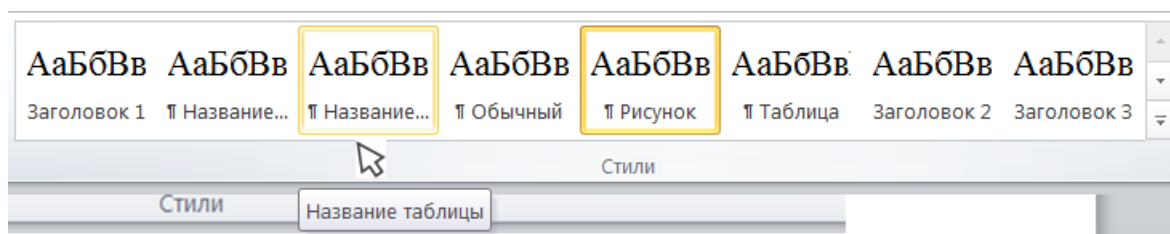


Рисунок 3.12 – Отредактированный Список экспресс-стилей

Сохраним этот набор стилей с именем *Студенческий*, для чего на вкладке ленты *Главная* в группе *Стили* нажмем кнопку *Изменить стили* (см. рисунок 3.5), в списке строки *Набор стилей* выберем команду *Сохранить как набор экспресс стилей...* и при сохранении выберем предложенную системой папку в профиле пользователя (<диск>:\Users\<>пользователь>\AppData\Roaming\Microsoft\QuickStyles). Сохраненный набор появится в списке *Набор стилей*. В последующем, если Вы планируете использовать этот набор при работе с документами на другом компьютере, следует скопировать *Студенческий.dotx* в аналогичную папку профиля пользователя.

2 Написание основного текста документа

После настройки стилей можно приступать к набору основного текста. При этом перед написанием текста заголовков, названий таблиц и рисунков, перед созданием таблиц и диаграмм, перед вставкой рисунков следует выбрать соответствующий стиль в группе экспресс стилей.

При написании текста следует помнить следующие основные правила:

- 1) Клавиша *Enter* – конец абзаца. Если вы хотите выполнить переход на новую строку внутри абзаца, следует нажать *Shift+Enter*.
- 2) Вставить код перехода на новую страницу можно нажатием клавиш

Ctrl+Enter.

- 3) Интервалы между абзацами при необходимости должны задаваться в параметрах стиля или абзаца, но не вставкой пустых строк.
- 4) Между словами следует вставлять только один пробел.
- 5) Мягкий перенос в слове (т. е. рекомендуемое место переноса) можно вставить нажатием клавиш CTRL+дефис на основной клавиатуре (но см. пункт 9 – CTRL+минус на числовой клавиатуре – тире!).
- 6) Знак пробела следует всегда вставлять после знаков препинания, но не перед ними.
- 7) Нерастяжимый пробел можно вставить сочетанием клавиш Ctrl+Shift+Пробел, в этом случае размер пробела между словами не будет увеличиваться при выравнивании по ширине и такие слова не будут отрываться друг от друга при переносе. Например, А. С. Пушкин (здесь между инициалами и фамилией вставлены 2 нерастяжимых пробела). В соответствии с СТО нерастяжимый пробел следует использовать в заголовках разделов (глав) между предложением или союзом и словом, к которому он относится.
- 8) Неразрывный дефис (использование аналогично п. 7) можно вставить клавишами Ctrl+Shift+дефис.
- 9) Знак тире должен иметь слева и справа от себя пробел. При правильном написании и использовании короткого дефиса Word производит автозамену на более длинную черточку. Этот же знак появится, если нажать клавиши CTRL+минус на числовой клавиатуре.
- 10) Слова, заключенные в кавычки или скобки, не должны отделяться от них пробелами, например: (текст), но не (текст).
- 11) Для ввода римских цифр используются приписные латинские буквы I, V, X, L, C, D.
- 12) В тексте следует использовать автоматическую расстановку переносов, что задается в группе *Параметры страницы* на вкладке *Разметка страницы*.
- 13) В параметрах Word обычно задано «Автоматически проверять орфографию» и «Автоматически проверять грамматику». Набор правил по умолчанию задан «Для деловой переписки». При необходимости можно выбрать другой набор правил. Орфографические ошибки в тексте Word подчеркивает волнистой красной линией, грамматические – зеленой. Щелчок правой кнопкой мыши на таком выделении позволяет при необходимости исправить ошибку или прочитать замечание по грамматике.
- 14) Масштаб документа на экране можно изменять колесиком мышки при нажатой клавише Ctrl.
- 15) Окно Word для одного документа можно разделить на два по вертикали использованием кнопочки с черточкой в правом верхнем углу

рабочей области.

- 16) Интервал между страницами можно скрыть двойным щелчком мыши на нем.
- 17) Абзац или выделенный фрагмент текста при необходимости могут иметь границы, нарисованные линиями разных типов и заливку разными цветами (кнопки в группе *Абзац* вкладки ленты *Главная*).
- 18) Для подсчета количества знаков в тексте или в выделенном фрагменте можно использовать кнопку *Статистика* в группе *Правописание* на вкладке ленты *Рецензирование*.

Учебное задание

1. Начать набор основного текста для выданного преподавателем документа с оформлением в соответствии с правилами для студенческих работ.

Лабораторная работа № 4. Microsoft Word 2010. Работа с таблицами, формулами, диаграммами, рисунками

Цель работы

Освоение приемов работы с таблицами, формулами и диаграммами.

Учебные вопросы

1. Оформление таблиц
2. Вставка формул
3. Построение диаграмм
4. Рисование и вставка рисунков
5. Автоматическое формирование оглавления

Краткое изложение учебного материала по теме

1 Оформление таблиц

Таблица в документе в соответствии с правилами для студенческих работ должна иметь следующую структуру и оформление:



Рисунок 4.1 – Элементы таблицы

Главный принцип при оформлении таблиц – поясняющая надпись в столбце заголовков и соответствующая ей информация должны располагаться в одной строке, каждое числовое значение в таблице должно находиться в отдельной клетке, независимо от того, как разлинована таблица (т. е. линий между строками и колонками мы можем не видеть).

Создание таблицы следует начинать с подсчета в ней количества столбцов и строк. Например, на рисунке 4.2 показана таблица, в которой 9 столбцов и 10 строк (оформление ее имеет много недостатков, главный – нет центрирования текста в ячейках шапки по горизонтали и по вертикали).

Чтобы оформить такую таблицу выберем место ее размещения в тек-

сте, зададим стиль абзаца *Таблица* (в этом стиле междустрочный интервал одинарный и нет отступа первой строки!), перейдем на вкладку ленты *Вставка* и нажмем на кнопку *Таблица*. Откроется панель, показанная на рисунке 4.3.

Время работы инструмента после заточки, Т, мин.	Множитель a_p для процессов							
	Пиления				Фрезерования цилиндрического продольного, поперечного	Точения продольного, сверления	Долбления фрезерной цепочкой	
	Рамное	Ленточное	Круглой пилой продольное	Круглой пилой поперечное			При осевой подаче	При боковой подаче
0	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
60	1,17	1,20	1,30	1,10	1,20	1,20	1,24	1,30
120	1,32	1,40	1,50	1,15	1,30	1,30	1,40	1,52
180	1,42	1,60	1,80	1,20	1,42	1,35	1,55	1,73
240	1,50	1,80	2,20	1,30	1,50	1,40	1,65	1,88
300	-	2,00	2,50	1,40	1,57	1,45	1,78	2,00
360	-	2,20	2,80	1,50	1,60	1,50	1,86	2,10

Рисунок 4.2 – Пример таблицы в журнальной статье

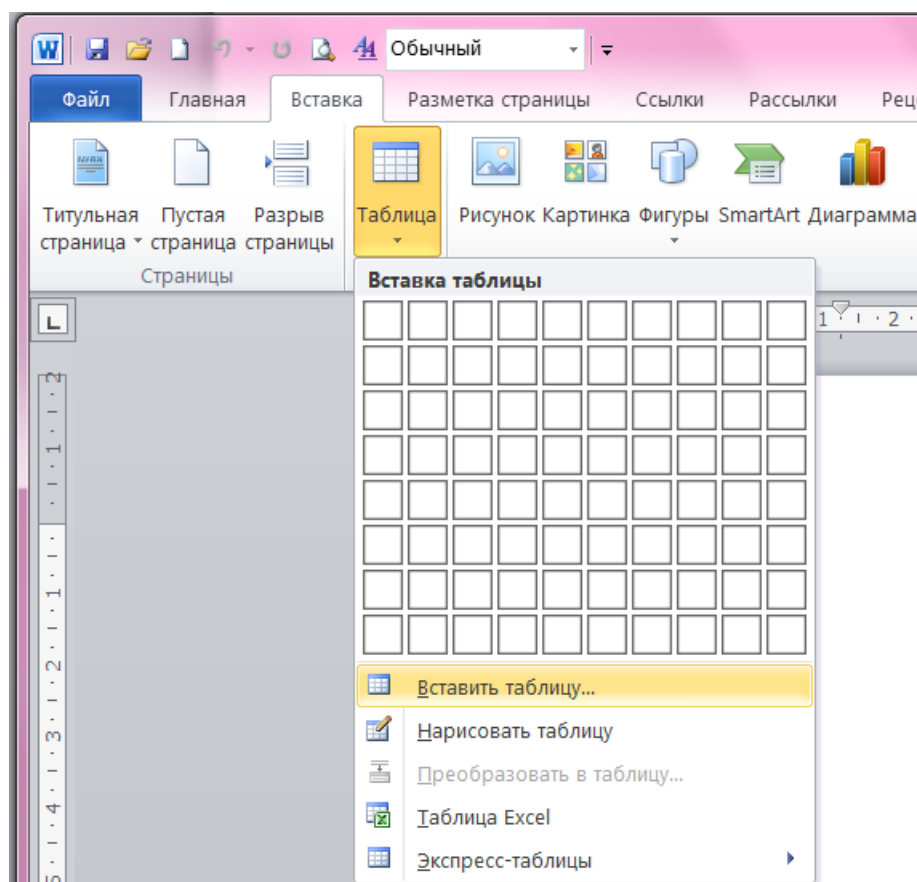


Рисунок 4.3 – Панель кнопки *Таблица* вкладки *Вставка*

Если оформляемая таблица имеет не более 10 столбцов и 8 строк, для ее создания можно выбрать необходимое количество клеточек в верхней части панели. Если размер больше, следует выбрать команду *Вставить таблицу...* и в появившемся окне *Вставка таблицы* (рисунок 4.4) задать ее размер.

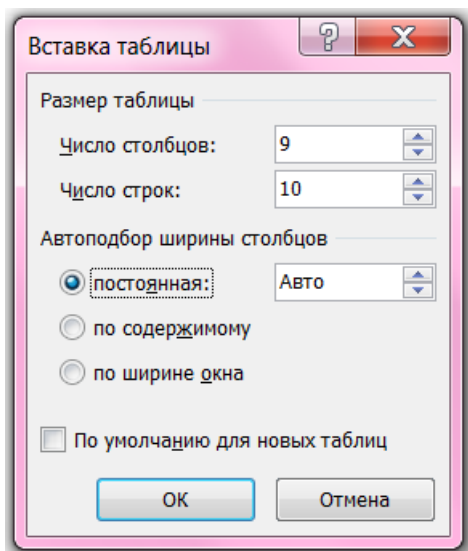


Рисунок 4.4 – Окно задания размеров таблицы и ширины её столбцов

при этом включить режим *Отобразить сетку* – кнопка слева на вкладке *Макет* (рисунок 4.6).

Если в этом окне оставить параметр *Автоподбор ширины столбцов – постоянная: Авто*, таблица по ширине будет занимать 100% рабочей области листа с одинаковой шириной всех колонок (рисунок 4.5).

Далее при оформлении таблицы следует изменить ширину столбцов так, чтобы все надписи в шапке таблицы и информация ее основной части были размещены наиболее рационально, высота таблицы была минимальна, но числа в таблице были правильно написаны.

При работе с таблицей на ленте присутствует контекстные вкладки *Конструктор* и *Макет* с большими возможностями по оформлению таблиц. Целесообразно

Рисунок 4.5 – Первоначальный вид таблицы

Далее следует оформить шапку таблицы. Для сложной шапки, как на рисунке 4.2, следует пользоваться командой *Объединить ячейки*. Для этого сначала нужно выделить группу ячеек, затем выбрать эту кнопку на вкладке *Макет* или аналогичную команду в контекстном меню. Не рекомендуется при оформлении таблицы пользоваться командой *Разделить ячейки*, т. к. это может привести к непредсказуемым последствиям при из-

менении формата таблицы и при вставке новых строк или столбцов.

После объединения ячеек в шапке, ввода текста и цифр мы получим таблицу 4.1. При оформлении таблицы можно уменьшить размер шрифта до 10. Поля ячеек (кнопка на вкладке *Макет*) можно уменьшить при необходимости до 0,1 см или до 0. Выравнивание в шапке таблицы обычно задается по центру по горизонтали и по вертикали. Выравнивание чисел в ячейках таблицы следует выполнять по правому краю, при необходимости можно увеличить поле справа у таких ячеек.

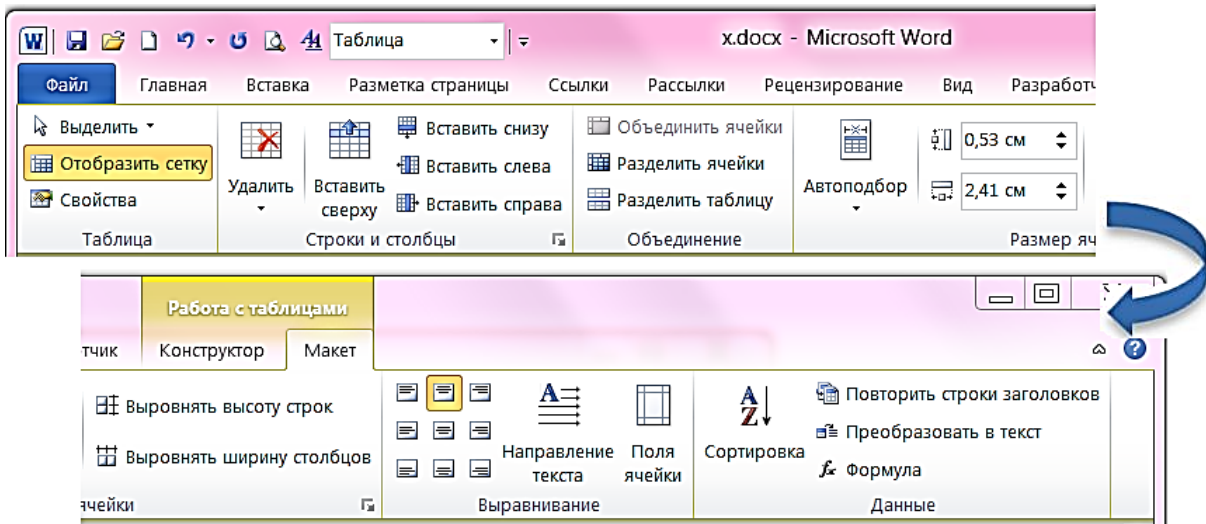


Рисунок 4.6 – Контекстная вкладка ленты *Макет* работы с таблицами

Таблица 4.1 – Пример оформления таблицы

Время работы инструмента после заточки, T, мин.	Множитель a_p для процессов							
	пиления				фрезерования цилиндрического продольного, поперечного	точения продольного, сверления	долбления фрезерной цепочкой	
	рамное	ленточное	круглой пилой продольное	круглой пилой поперечное			при осевой подаче	при боковой подаче
0	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
60	1,17	1,20	1,30	1,10	1,20	1,20	1,24	1,30
120	1,32	1,40	1,50	1,15	1,30	1,30	1,40	1,52
180	1,42	1,60	1,80	1,20	1,42	1,35	1,55	1,73
240	1,50	1,80	2,20	1,30	1,50	1,40	1,65	1,88
300	-	2,00	2,50	1,40	1,57	1,45	1,78	2,00
360	-	2,20	2,80	1,50	1,60	1,50	1,86	2,10

После заполнения информацией шапки таблицы следует настроить ширину столбцов, перемещая мышкой границы столбцов.

Для правильного переноса слов следует использовать вставку мягкого переноса (Ctrl+дефис основной части клавиатуры).

Разлиновать таблицу можно линиями разного вида и толщины. В соответствии с правилами оформления студенческих работ горизонтальные и

вертикальные линии, разграничивающие строки таблицы, допускается не проводить, если их отсутствие не затрудняет пользование таблицей.

2 Вставка формул

Для оформления формул в документе может быть использована кнопка *Формула* на вкладке *Вставка*. При этом можно выбрать за основу предложенные образцы и затем отредактировать их с использованием контекстной вкладки *Конструктор* или вставить новую формулу (рисунок 4.7). Следует помнить, что при сохранении документа в старом формате Word 2003 такие формулы превращаются в рисунки, однако могут снова редактироваться, как формулы, при обратном сохранении этого документа в формате Word 2010.

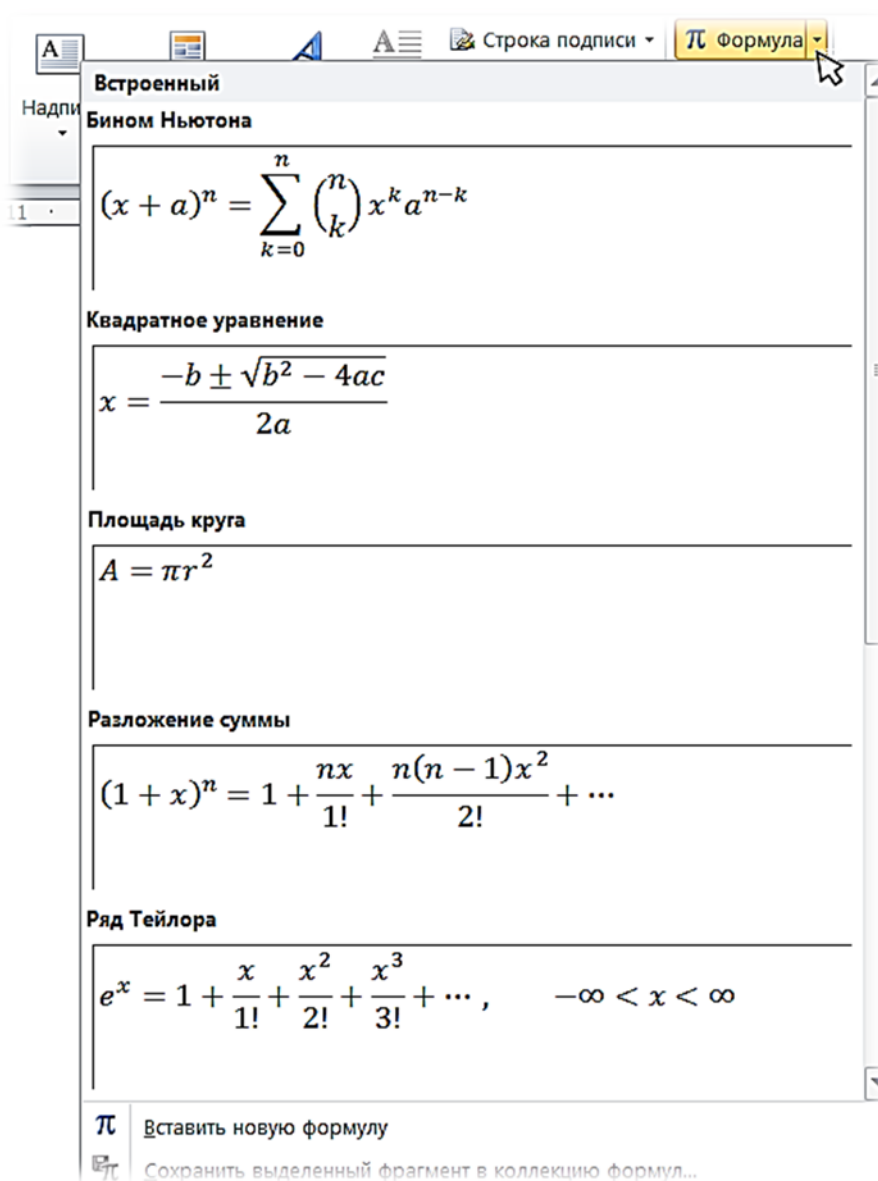


Рисунок 4.7 – Панель кнопки *Формула*

Word 2010 сохранил также возможность работать с редактором фор-

мул предыдущих версий (Microsoft Equation 3.0). Чтобы создать новую формулу этим способом, следует использовать команду *Вставить объект* на вкладке *Вставка*.

В новом режиме построения формул на контекстной вкладке Конструктор присутствуют группы шаблонов Символы и Структуры для построения различных фрагментов формул (рисунок 4.8).

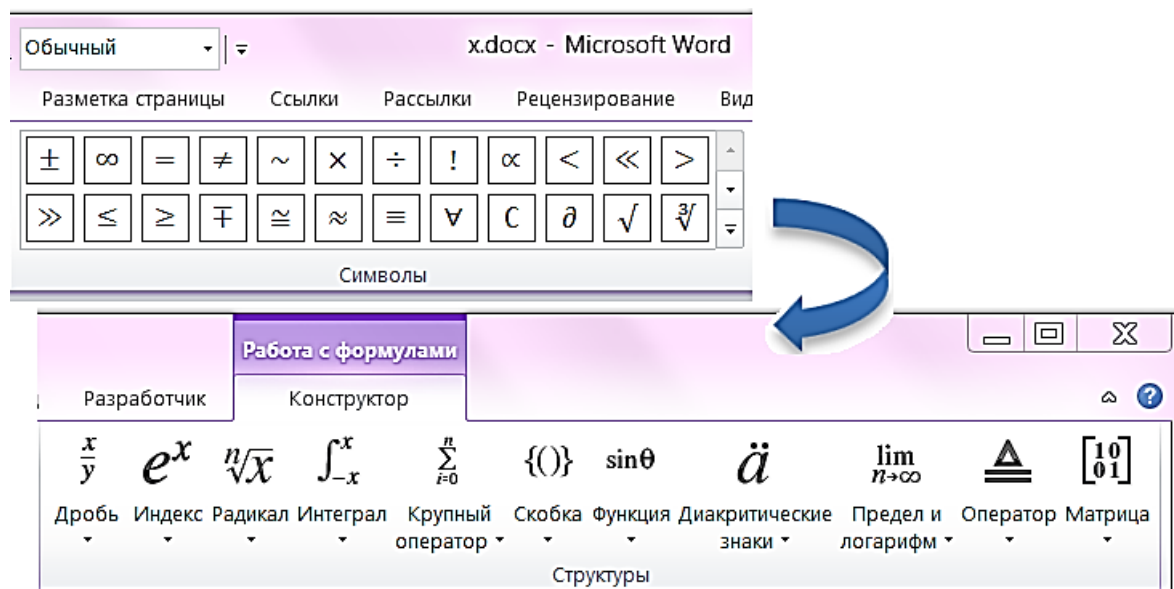


Рисунок 4.8 – Контекстная вкладка *Конструктор* для работы с формулами

При выборе любой кнопки в группе *Структуры* открывается панель с вариантами шаблонов для выбранного типа структуры.

Предположим, нам нужно создать формулу, показанную на рисунке 4.9.

$$\sigma = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n}}$$

Рисунок 4.9 – Рисунок формулы

Для этого нажмем на кнопку *Формула*, появится графический объект с надписью внутри него *Место для формулы*. Найдем в группе *Символы* знак σ , нажмем на клавиатуре символ = и выберем в группе *Структуры* символ квадратного корня. Получится следующая часть формулы:

$$\sigma = \sqrt{\square}$$

Далее выберем шаблон дроби, в числителе вставим из шаблонов значок суммы, напишем с использованием шаблонов индексов и символов x , i , 2 на клавиатуре все выражение числителя и букву n знаменателя. Буква x с

черточкой сверху находится в шаблоне *Диакритические знаки*. Чтобы добавить к выражению в скобках показатель степени, следует выделить этот фрагмент и в шаблонах выбрать верхний индекс. На вкладке *Главная* можно задать размер букв для формулы. Получим следующую формулу:

$$\sigma = \sqrt{\frac{\sum(x_i - \bar{x})^2}{n}}$$

3 Построение диаграмм

Построение диаграмм с заданием для них числовой информации можно выполнить непосредственно в системе Word 2010. Для этого на вкладке *Вставка* следует нажать на кнопку *Диаграмма* (в группе *Иллюстрации*), после чего откроется окно выбора типа диаграммы (рисунок 4.10).

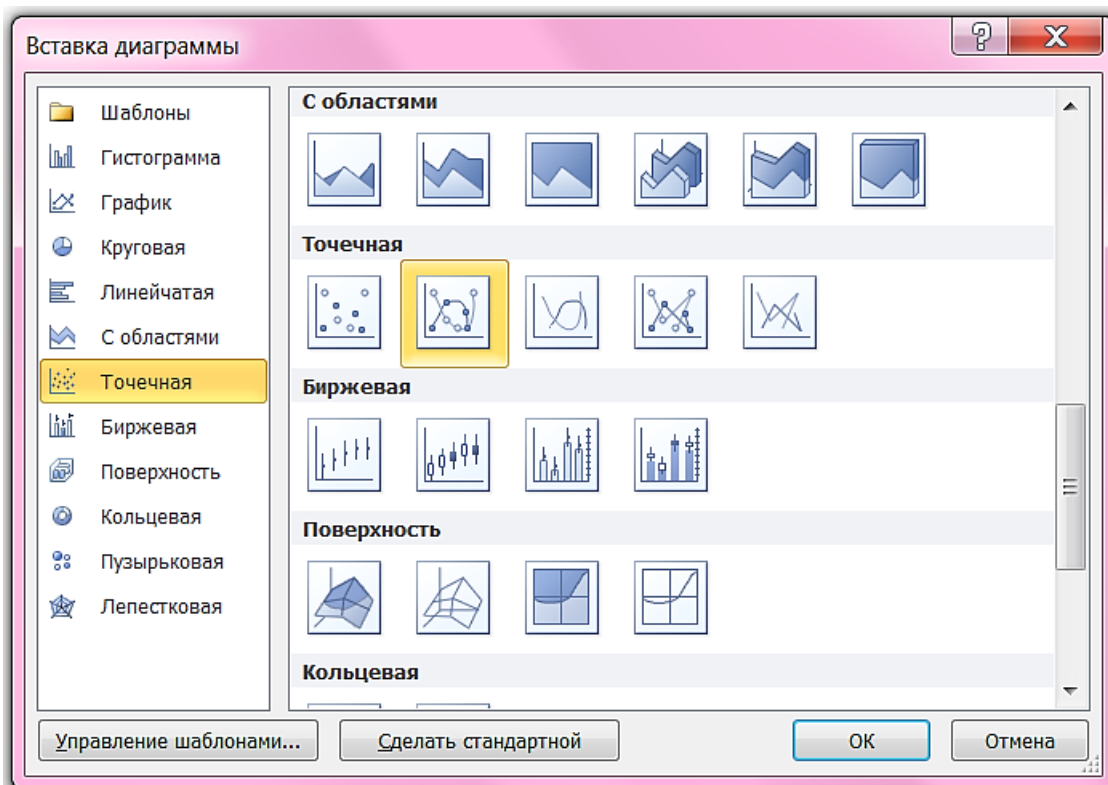


Рисунок 4.10 – Окно выбора типа диаграммы

В то же время в системе Word 2010 присутствует и старая возможность построения графиков с использованием объекта *Диаграмма Microsoft Graph*.

Предположим, у нас есть данные, приведенные в таблице 4.1. Для построения графиков зависимостей таблицы 4.1 выделяем все клетки таблицы, содержащие цифры и копируем в буфер обмена. Затем выбираем

место, где будет размещена диаграмма, на вкладке *Вставка* нажимаем на кнопку *Диаграмма* и выбираем тип *Точечная – Точечная с маркерами*. Открывается окно системы Excel, в котором присутствуют некоторые данные, как образец. Мы выбираем на листе клетку A2 и выполняем вставку данных из буфера, после чего на листе Word сразу видим диаграмму, на которой показаны данные одной зависимости $Y = F(X)$. Вид таблицы показан на рисунке 4.11.

Таблица 4.1 – Зависимость выхода продуктов крекинга от глубины превращения сырья

Глубина превращения	Газойль	Бензин	Газ
0	0,000	0	0
0,1	0,020	0,01	0
0,2	0,170	0,03	0
0,3	0,250	0,05	0
0,4	0,300	0,09	0,01
0,5	0,325	0,12	0,055
0,6	0,340	0,18	0,08
0,7	0,310	0,23	0,16
0,8	0,260	0,33	0,21
0,9	0,180	0,36	0,3
0,98	0,080	0,33	0,5

	A	B	C	D
1	Значения X	Значения Y		
2	0	0	0	0
3	0,1	0,02	0,01	0
4	0,2	0,17	0,03	0
5	0,3	0,25	0,05	0
6	0,4	0,3	0,09	0,01
7	0,5	0,325	0,12	0,055
8	0,6	0,34	0,18	0,08
9	0,7	0,31	0,23	0,16
10	0,8	0,26	0,33	0,21
11	0,9	0,18	0,36	0,3
12	0,98	0,08	0,33	0,5
13	Для изменения диапазона данных диаграммы перетащите правый нижний угол диапазона.			

Рисунок 4.11 – Данные диаграммы в окне Excel

Чтобы получить на диаграмме все зависимости, перетащим маркер в

правом нижнем углу синей линии вправо до столбца D, после чего увидим диаграмму, приведенную на рисунке 4.12.

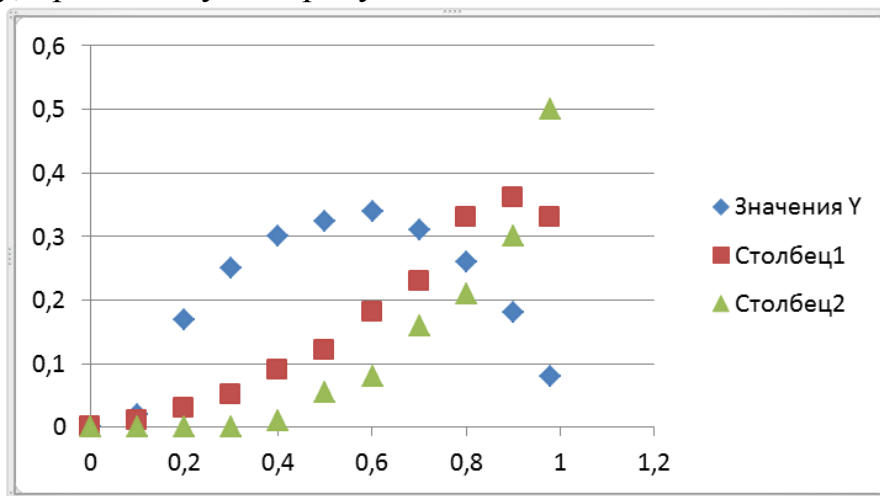


Рисунок 4.12 – Первоначальный вид диаграммы

Диаграмма и все ее элементы имеют большие возможности по настройке, представленные на контекстных вкладках ленты группы *Работа с диаграммами: Конструктор, Макет и Формат*. К изменению параметров элементов диаграммы можно также перейти двойным щелчком мыши на выбранном элементе или выбором в контекстном меню пункта *Формат ...* (после слова *Формат* идет название выбранного элемента).

Проведем через точки экспериментальных данных линии тренда с аппроксимацией экспериментальных данных полиномами 3-ей или 4-ой степени (полиномы 4-ой и более степеней следует использовать крайне осторожно, так как на них может присутствовать много экстремумов). Вид диаграммы с линиями тренда (после задания некоторых параметров этих линий – толщина, цвет, показ коэффициента аппроксимации) приведен на рисунке 4.13. Размер маркеров на диаграмме уменьшен, легенда (надписи для кривых) удалена.

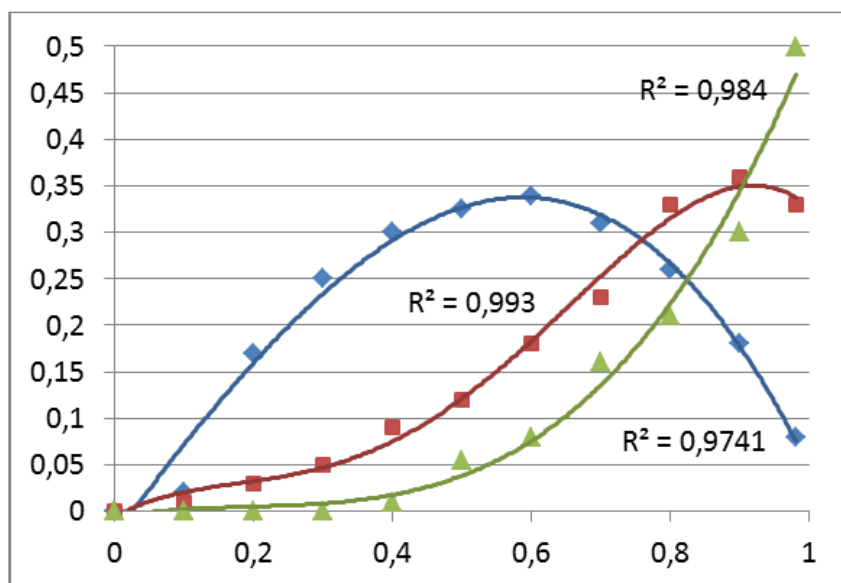
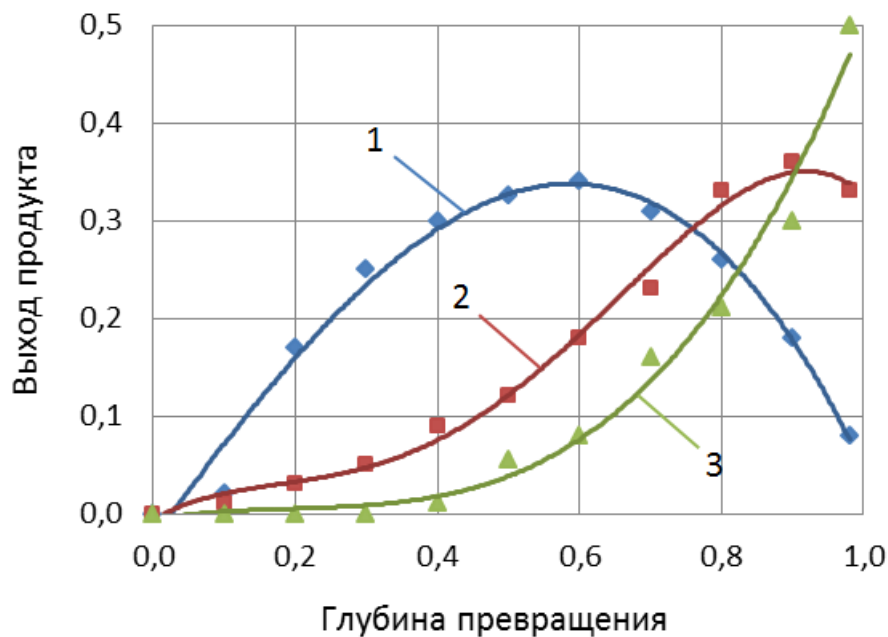


Рисунок 4.13 – Диаграмма с линиями тренда

Для окончательного оформления диаграммы покажем вертикальную сетку, риски сетки (наружу) уберем, оформим надписи в соответствии с правилами оформления. Линии-выноски и надписи с номерами кривых добавляем на диаграмму с использованием вкладки ленты *Вставка – Фигуры*. Для вертикальной оси зададим шаг сетки 0,1. Для чисел на осях зададим формат числовой с одним в дробной части. Уберем рамку вокруг диаграммы.

Окончательный вид диаграммы показан на рисунке 4.14.



1 – газойль; 2 – бензин; 3 – газ

Рисунок 4.14 – Окончательный вид диаграммы

4 Рисование и вставка рисунков

Рисование в текстовом редакторе Word 2010 можно выполнить с использованием набора векторных фигур кнопки *Фигуры* группы *Иллюстрации* на вкладке *Вставка* (рисунок 4.15).

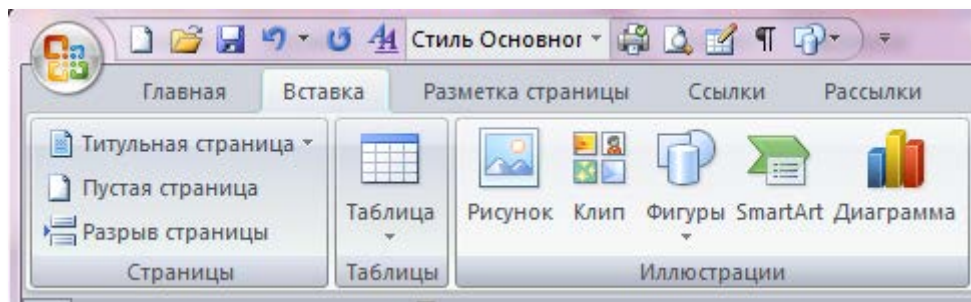


Рисунок 4.15 – Группа *Иллюстрации* вкладки *Вставка*

При нажатии на кнопки **Фигуры** появляется панель, содержащая набор готовых фигур, разбитых на категории (рисунок 4.16). Если рисунок будет состоять из группы связанных между собой фигур, внизу панели можно выбрать команду **Новое полотно**. После этого в документе появляется пустая область для рисования, а на ленте появляется новая панель **Формат**, которая включает в себя ряд инструментов по форматированию только что созданных фигур.

Если рисование выполняется без использования **Полотна**, несколько объектов можно объединить в единый рисунок с использованием группы **Упорядочить** на вкладке **Формат** или командами контекстного меню. Фигуры можно **Группировать**, **Разгруппировать** и **Перегруппировать**, в зависимости от поставленной задачи (рисунок 4.17).

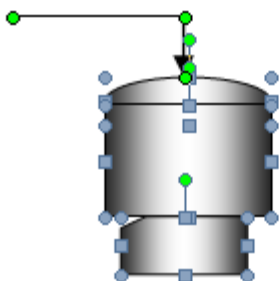
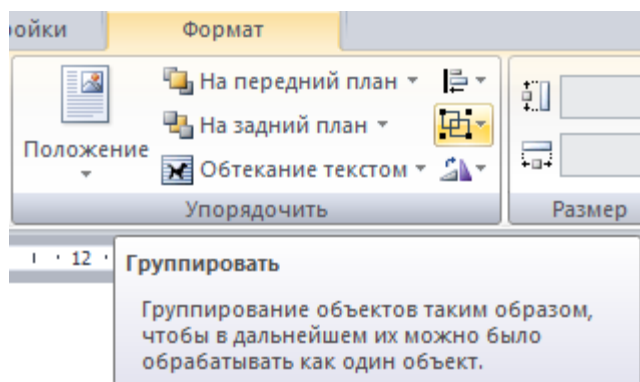


Рисунок 4.17 – Группировка фигур

Для группировки объектов необходимо их выбрать с помощью команды **Выделить** на вкладке ленты **Главная** ► **Редактирование** или «мышкой» при нажатой кнопке **SHIFT** или **CTRL**.

Параметры графических объектов задаются с помощью контекстной

Рисунок 4.16 – **Фигуры** для векторного рисунка

вкладки **Средства рисования – Формат** или с использованием диалогового окна **Формат фигуры**, которое можно вызвать из контекстного меню. Здесь задаются параметры заливки, контур и эффекты фигур, размеры, поворот и масштаб, положение в тексте или обтекание, поля и выравнивание надписи (текст можно вставить в прямоугольники, окружности и др. фигуры) и прочее (рисунок 4.18).

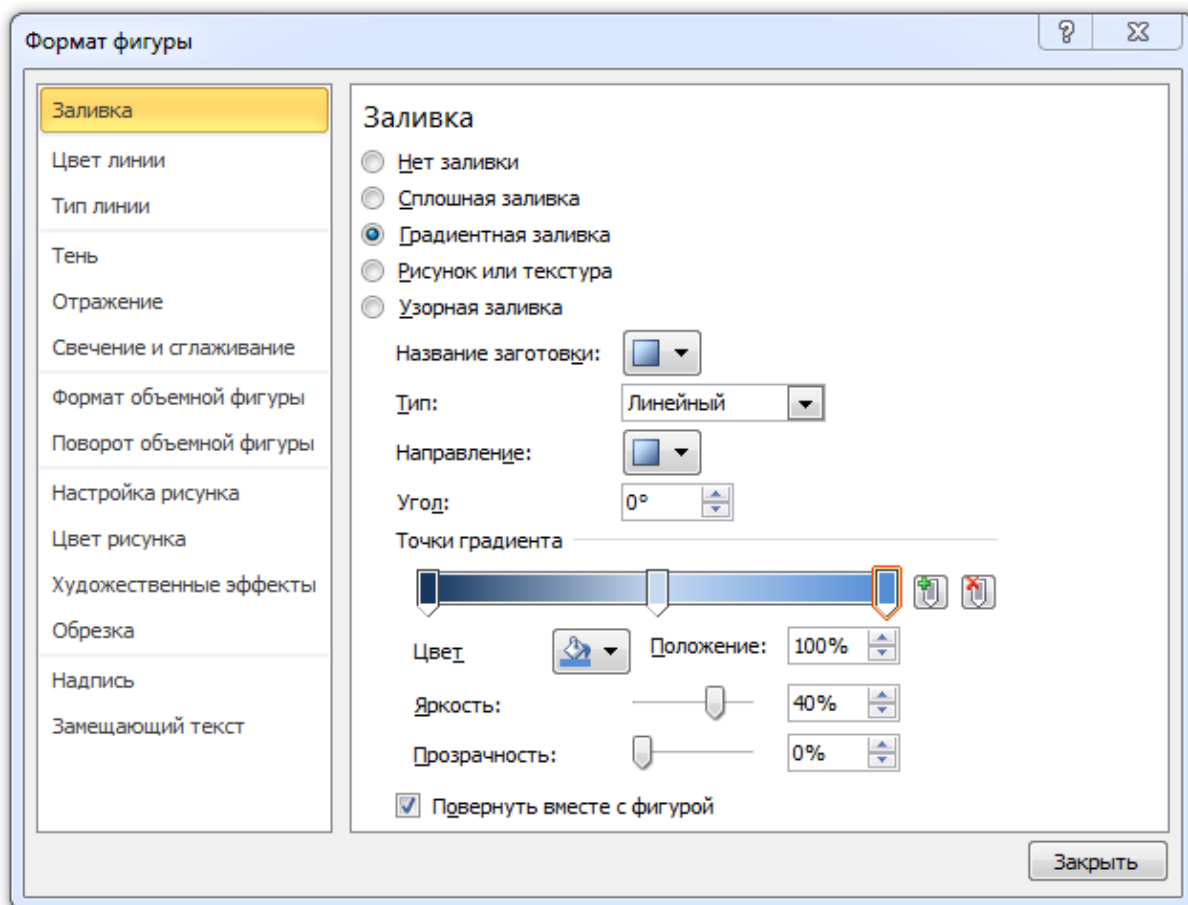


Рисунок 4.18 – Окно **Формат фигуры**

При создании рисунка или вставке в документ рисунка другого файла его можно расположить в слое текста (**в тексте**) или вне слоя текста (**с обтеканием текстом**). Рисунок, расположенный **в тексте**, вставляется непосредственно в позицию курсора, ведет себя как обычный текстовый абзац и требует для своего размещения отдельное место в текстовом пространстве. Рисунки **с обтеканием текстом** не требуют для себя отдельного места в текстовом пространстве. Это позволяет задать их точное положение на странице и поместить за слоем текста или перед ним, а также задать расположение относительно других объектов. Формат рисунка в данной версии офиса задается на контекстной вкладке работы с рисунком **Формат ► Положение** или **Обтекание**. Кнопка **Обтекание** позволяет задать варианты: *в тексте, вокруг рамки, по контуру, сквозное, сверху и снизу, за текстом, перед текстом*.

5 Автоматическое формирование оглавления

Автоматическое формирование оглавления – функция Word, которую следует использовать для сложного структурированного документа (состоящего из разделов, подразделов, пунктов). Для использования этого режима необходимо все заголовки, включаемые в оглавление, оформлять стилями соответствующего уровня (см. Лабораторную работу № 3).

В **Word 2010** оглавление формируется с помощью кнопки **Оглавление** на ленте **Ссылки**, которая позволяет сформировать авто-собираемое оглавление или создать ручное оглавление выбранного стиля (рисунок 4.19).

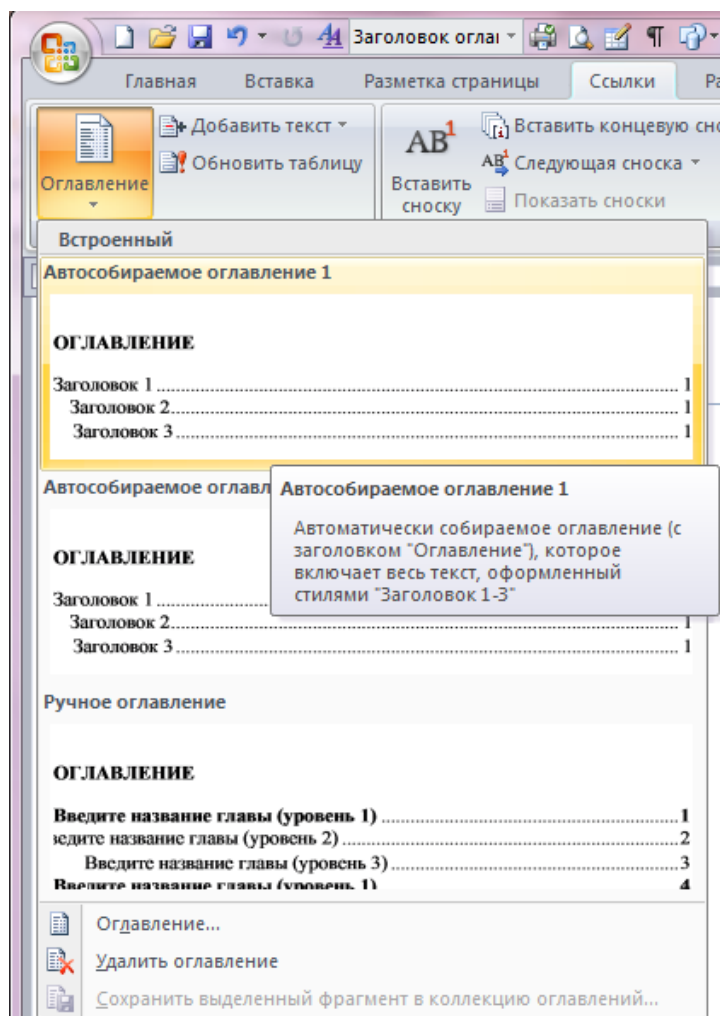


Рисунок 4.19 – Создание оглавления

Учебное задание

1. Для выданного преподавателем документа оформить таблицы, формулы и диаграммы в соответствии с правилами для студенческих работ.

Лабораторная работа № 5. Microsoft Excel 2010. Работа с формулами. Использование абсолютной и относительной адресации. Математические функции

Цель работы

Освоение основных приемов работы в электронной таблице. Занесение данных, использование автозаполнения и формул с относительными и абсолютными ссылками.

Учебные вопросы

1. Ячейки электронной таблицы и их адресация
2. Занесение данных в ячейки
3. Работа с формулами. Относительные и абсолютные ссылки
4. Использование математических функций и матричных операций

Краткое изложение учебного материала по теме

1 Ячейки электронной таблицы и их адресация

Основным элементом электронной таблицы является ячейка (или клетка) таблицы.

В каждую ячейку может быть помещена информация различного типа:

- **текстовая** длиной до 32767 символов;
- **числовая**;
- **дата**;
- **время**.

Полную техническую характеристику, возможности и ограничения Excel 2010 можно посмотреть на сайте Microsoft <http://office.microsoft.com/ru-ru/excel-help/HP010342495.aspx?CTT=1>.

На лист электронной таблицы можно также вставить объекты разных типов: диаграммы, векторные фигуры, рисунки, надписи в рамке, фигурный текст (WordArt) и пр.

Ячейки таблицы образуют строки и столбцы. Каждая ячейка имеет адрес, образованный обозначением столбца и строки (например, **A1**, **BB22**, **ССС333**), который может использоваться при написании формул (рисунок 5.1).

Стиль **A1** адреса ячейки можно изменить на другой – на числовое обозначение столбцов и колонок (вкладка **Файл** ► **Параметры**, в появившемся окне **Параметры Excel** в разделе **Формулы** поставить галочку у пункта **Стиль ссылок R1C1**). Те же ссылки, что и ранее, будут выглядеть,

как **R1C1**, **R22C54**, **R333C2109**, где **R** – row (ряд), **C** – column (колонка). Таблица с типом адресации *R1C1* показана на рисунке 5.2.

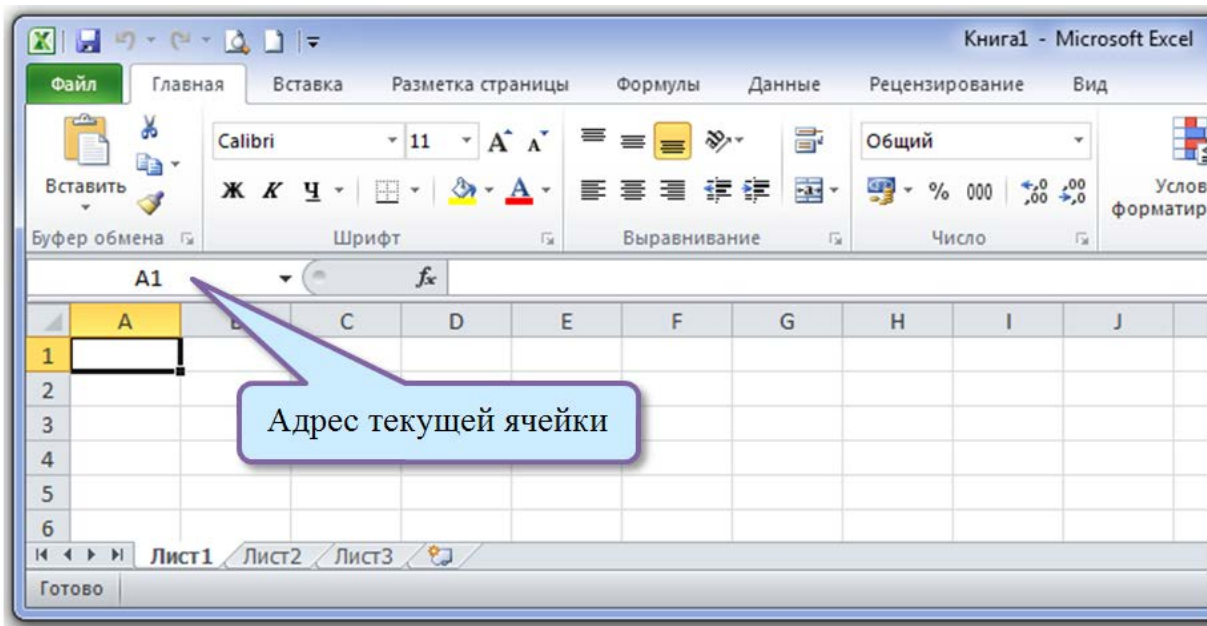


Рисунок 5.1 – Книга Microsoft Excel с типом адресации *A1*

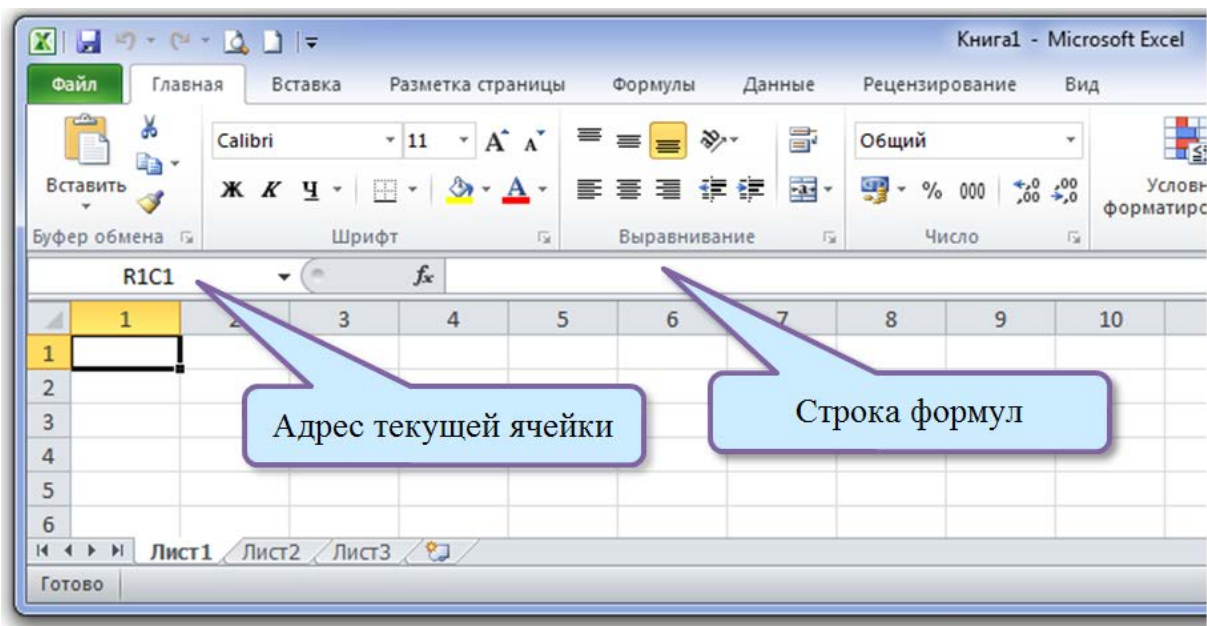


Рисунок 5.2 – Книга Microsoft Excel с типом адресации *R1C1*

Каждая ячейка электронной таблицы имеет множество свойств, которые определены по умолчанию для выбранного стиля ячейки. Свойства можно просмотреть и изменить на вкладке *Главная* ленты, либо с использованием традиционного диалогового окна *Формат ячеек*, которое показано на рисунке 5.3 (вызвать его можно из контекстного меню или маленькой кнопкой в правом нижнем углу любой группы кнопок вкладки *Главная*).

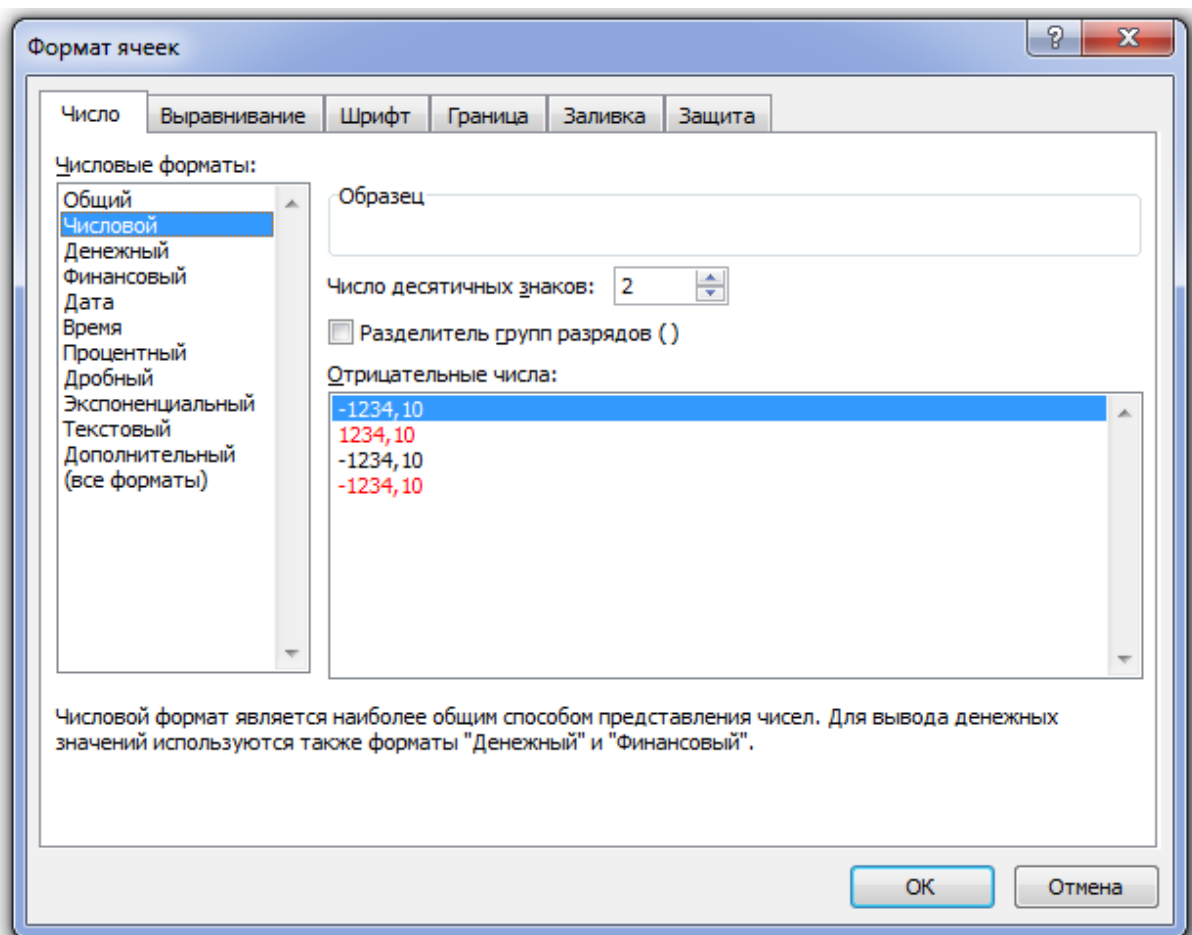


Рисунок 5.3 – Окно Формат ячеек

Ячейка электронной таблицы по умолчанию имеет стиль **Обычный** (лента *Главная*, группа *Стили*). Параметры стиля **Обычный** можно изменить, тогда все ячейки таблицы этого стиля будут иметь новые свойства. Можно создать новый стиль, если для ячейки задать новые свойства.

В соответствии с требованиями по оформлению студенческих работ зададим для стиля ячеек *Обычный* шрифт Times New Roman, обычный, размер 13, черный, как показано на рисунке 5.4.

2 занесение данных в ячейки

При размещении информации на листе следует предварительно задать параметры страницы (вкладка *Разметка страницы*), чтобы на листе появились пунктирные линии границ страницы. Перед вводом информации для ее однозначного толкования системой можно задать тип данных в ячейке в группе *Число* вкладки *Главная*.

Чтобы занести информацию в выбранную ячейку, нужно щелкнуть на ней левой кнопкой мыши и приступить к вводу данных с клавиатуры. При этом старая информация в ячейке, если она присутствовала, уничтожается.

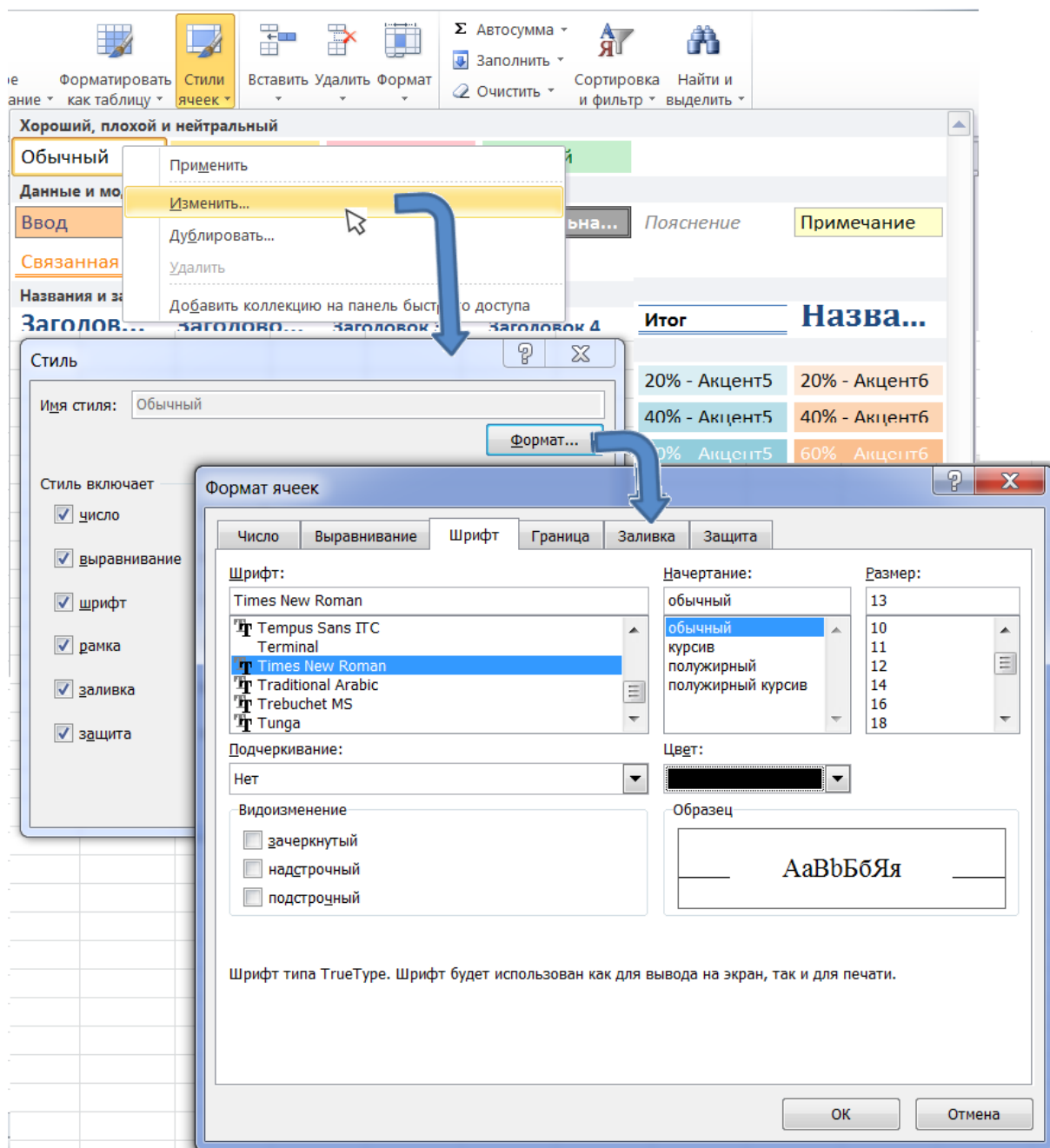


Рисунок 5.4 – Редактирование стиля ячеек *Обычный*

Редактирование существующей информации ячейки возможно в *Строке формул* (см. рисунок 5.2) или в ячейке после двойного щелчка на ней мышкой.

Следует помнить, что в Excel существует внутреннее представление информации и внешнее. Внешнее представление задается форматом ячейки. Внутреннее – 15 значащих цифр для чисел, 32767 символов при вводе текста в ячейку (вставить из буфера обмена можно больше). Внутреннее представление даты – целое число, которое отсчитывается от 1 января 1900 г., формат времени – число от 0 до 0,99999999 (количество секунд в сутках равно 86400, одна секунда в Excel = $1/86400 = 0,0000115740$).

При вводе длинного текста в ячейку этот текст виден справа от ячейки

ки, пока не встретит не пустую ячейку (на ней текст оборвется). Если задать параметр *Перенос текста* (кнопка в группе *Выравнивание* вкладки Главная) – текст будет располагаться в несколько строк в одной ячейке или в группе объединенных ячеек (кнопка *Объединить и поместить в центре*). Примеры размещения текста на листе электронной таблицы показаны на рисунке 5.5.

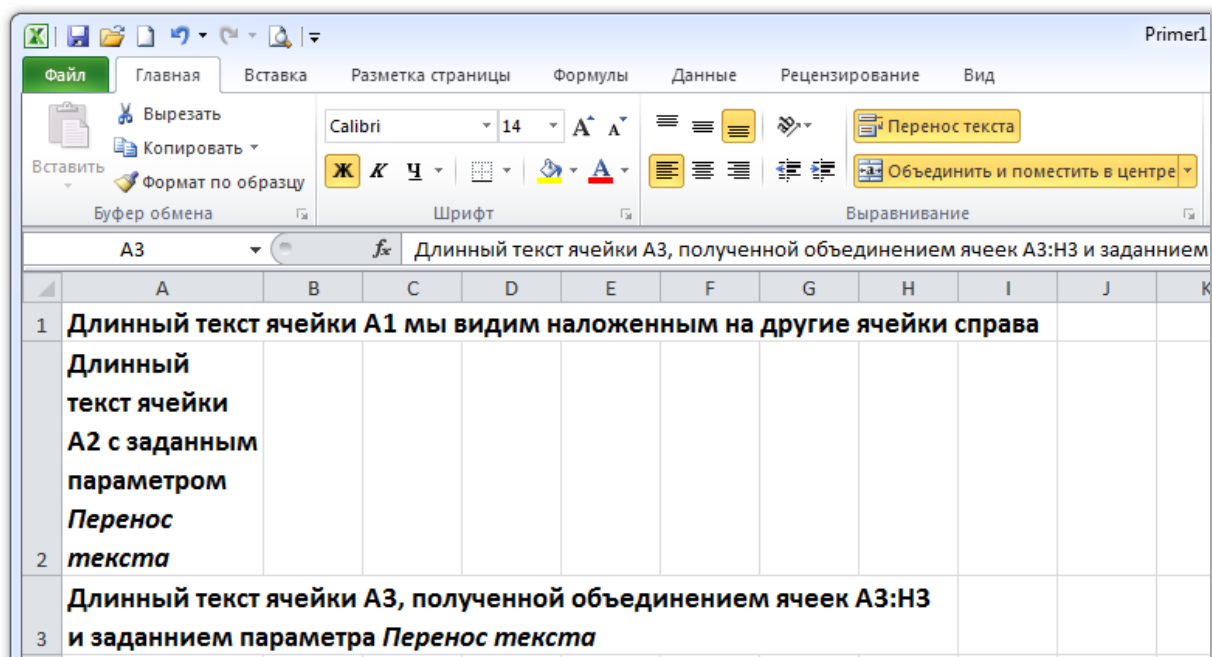


Рисунок 5.5 – Примеры работы с текстом в электронной таблице

При вводе чисел в ячейку следует учитывать следующее:

- разделителем целой и дробной части числа может быть точка или запятая. Вид разделителя определяется настройками Windows. **Поэтому следует всегда использовать цифры и точку (она является и запятой при соответствующей настройке) на числовой клавиатуре;**
- для ввода больших и маленьких чисел используется экспоненциальная форма с использованием латинской буквы **e**. Число **-1,23e-15** это **-1,23·10⁻¹⁵**;
- в формате чисел можно задать количество видимых знаков после запятой, для отрицательных чисел можно при необходимости задать формат **красные числа без знака**;
- для чисел можно задать разделитель групп разрядов (в соответствии с настройками Windows).

Тип данных **дата/время** наиболее сложен из-за наличия различных форм внешнего представления даты, времени или даты и времени одновременно. Формат дат и времени зависит от настроек Windows (*Язык и региональные стандарты*). Полный список всех стандартных форматов даты/времени можно просмотреть в окне *Формат ячеек* (см. рисунок 5.3),

кроме того можно определить собственные форматы представления данных.

При вводе данных может быть использовано автозаполнение для чисел, дат и определенных в системе Excel списков (рисунок 5.6).

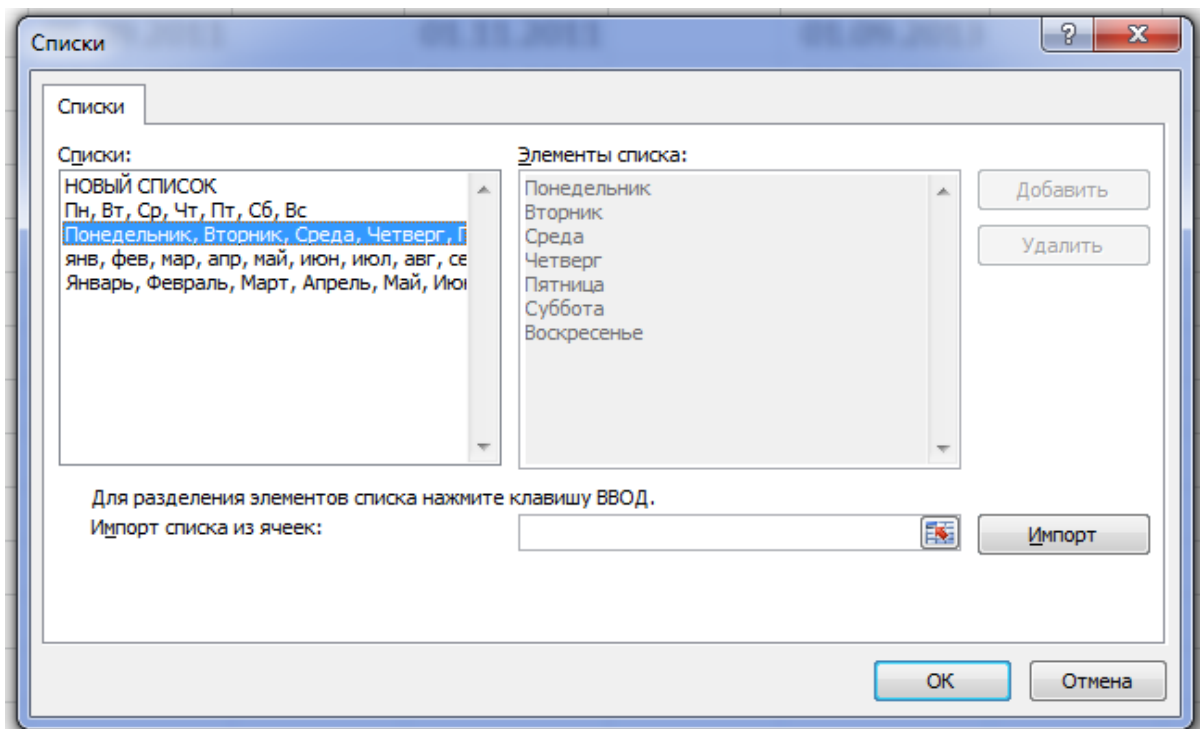


Рисунок 5.6 – Списки для автозаполнения системы Excel

Пример использования списков: пишем в ячейку **A2** **Пн**, затем с нажатой левой кнопкой мыши за маркер копирования/автозаполнения в правом нижнем углу ячейки **A2** расширяем выбор вниз до ячейки **A8**. Получаем список сокращенных названий дней недели (рисунок 5.7). Аналогично для других списков.

	A	B	C	D
1	Автозаполнение с использованием Списков Excel			
2	Пн	Понедельник	Янв	Январь
3	Вт	Вторник	Фев	Февраль
4	Ср	Среда	Мар	Март
5	Чт	Четверг	Апр	Апрель
6	Пт	Пятница	Май	Май
7	Сб	Суббота	Июн	Июнь
8	Вс	Воскресенье	Июл	Июль
9			Авг	Август
10				Сентябрь
11				Октябрь
12				Ноябрь
13			Дек	Декабрь

Маркер копирования/автозаполнения

Рисунок 5.7 – Использование Списков

При автозаполнении ячеек датами существует несколько вариантов: расширение выбора за маркер копирования/ автозаполнения левой кнопкой мыши – заполнение датами подряд по дням календаря; при растягивании за маркер копирования/автозаполнения правой кнопкой мыши – в контекстном меню появляются варианты: по дням, по рабочим дням (пятидневка), по месяцам, по годам (рисунок 5.8).

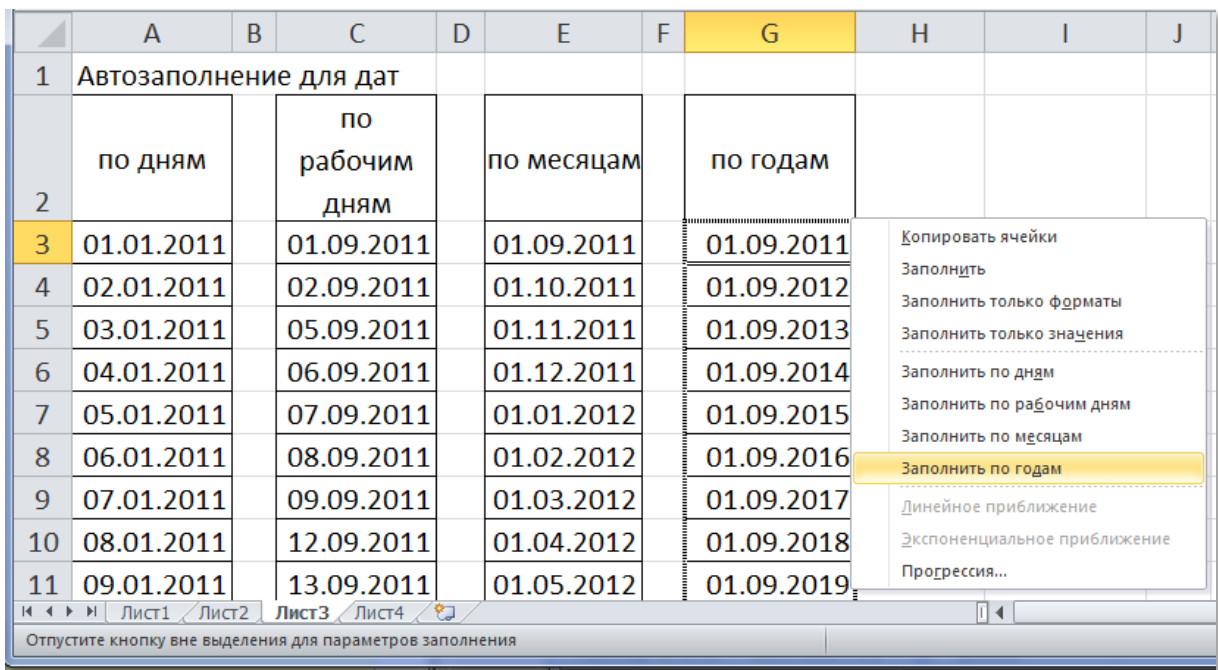


Рисунок 5.8 – Использование автозаполнения для дат

Автозаполнение для чисел при расширении выбора за маркер копирования/автозаполнения правой кнопкой мыши – после выбора в контекстном меню пункта *Прогрессия* появляется диалоговое окно для задания параметров арифметической или геометрической прогрессии (рисунок 5.9).

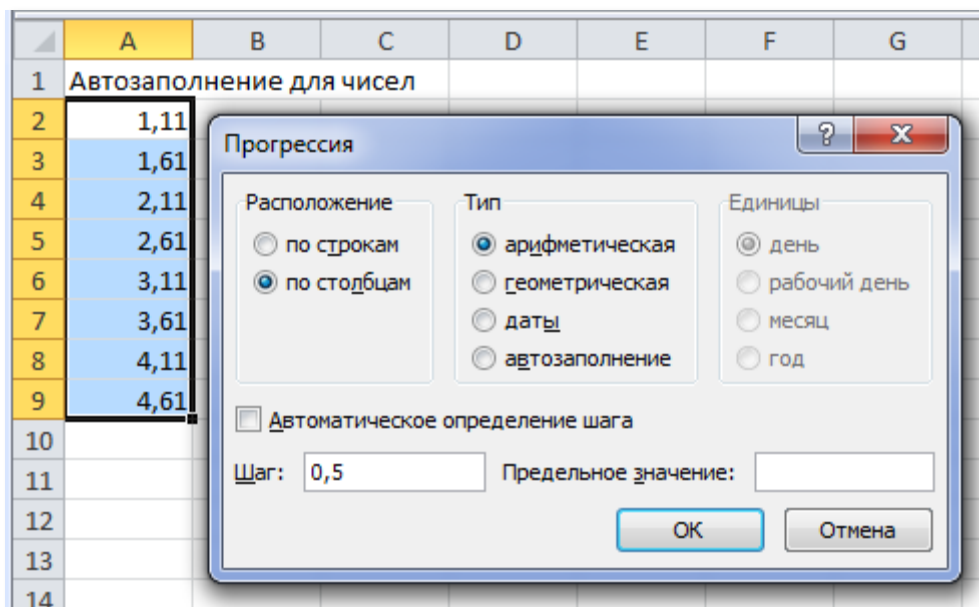


Рисунок 5.9 – Использование автозаполнения для чисел

Автозаполнение для текста, начинающегося или заканчивающегося числом – при копировании число в начале строки или в конце (при отсутствии числа в начале строки) автоматически увеличивается на 1 (рисунок 5.10).

	A	B	C	D	E	F	G
25	999 №		9 гр., п/гр. 1		текст 1,99		текст 1-99
26	1000 №		10 гр., п/гр. 1		текст 1,100		текст 1-100
27	1001 №		11 гр., п/гр. 1		текст 1,101		текст 1-101
28	1002 №		12 гр., п/гр. 1		текст 1,102		текст 1-102
29	1003 №		13 гр., п/гр. 1		текст 1,103		текст 1-103

Рисунок 5.10 – Использование автозаполнения для текста с числами

3 Работа с формулами. Относительные и абсолютные ссылки

Формулы – это выражения, с помощью которых можно выполнять вычисления, манипулировать содержимым других ячеек, проверять условия и пр.

Формула всегда начинается со знака равенства (=).

В Excel обычно задан режим показа результатов вычислений по формулам, но можно задать и режим показа формул (вкладка *Формулы* – *Показать формулы* в группе *Зависимости формул*).

Формулы в системе Excel могут состоять из постоянных значений (числа, текст, даты/время), символов математических операций (+, -, *, /, ^), круглых скобок для изменения порядка действий (система знает приоритет вычислений в математических формулах, принятый в математике), функций разного типа, ссылки на ячейки и имена ячеек.

Часто при вычислениях одинаковую операцию нужно выполнить с рядом данных. Простейший пример показан на рисунке 5.11.

	A	B	C	D
1	1,0	55	56,0	111,0
2	3,5	50	53,5	103,5
3	6,0	45	51,0	96,0
4	8,5	40	48,5	88,5
5	11,0	35	46,0	81,0

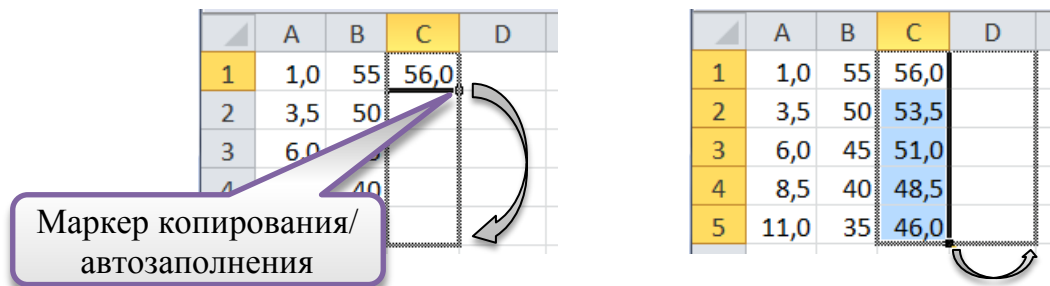
	A	B	C	D
1	1	55	=A1+B1	=B1+C1
2	3,5	50	=A2+B2	=B2+C2
3	6	45	=A3+B3	=B3+C3
4	8,5	40	=A4+B4	=B4+C4
5	11	35	=A5+B5	=B5+C5

Рисунок 5.11 – Расчет по формулам в Excel, слева – обычный режим с результатами расчета, справа – показ формул

Как видно из рисунка 5.11, одинаковую операцию сложения нужно выполнить с парами чисел колонок **A** и **B** в строках 1 – 5, затем с парами чисел колонок **B** и **C**.

Для организации подобных вычислений используется одна из главных особенностей электронных таблиц – изменение относительной адресации при копировании формул. Формула в колонке **C** была написана один раз в строке **1**, затем за маркер копирования/автозаполнения в правом

нижнем углу скопирована в остальные строки путем расширения выбора на строки 2 – 5, затем получившаяся выделенная колонка была скопирована в столбец D (рисунок 5.12).



Маркер копирования/
автозаполнения

Рисунок 5.12 – Копирование формул

При написании формулы не следует писать ссылки символами, а следует указывать на экране мышкой те ячейки, данные которых используются при вычислениях в формуле. При этом в формуле появляются ссылки на указанные ячейки.

Как видно из примера, при копировании формул ссылки изменяются: при копировании вниз формулы строки 1 =A1+B1, для строки 2 формула будет иметь вид =A2+B2 и т. д., при копировании вправо – формула для столбца C – =A1+B1, для столбца D – =B1+C1. Такие ссылки (A1, B1, C1) называются относительными.

Если при копировании формулы ссылка в ней или ее часть (обозначение столбца или строки) не должны изменяться, используется абсолютная и смешанная адресация. Для обозначения такого адреса используют знак доллар \$ перед неизменяющимся элементом (изменять тип адресации следует нажатием клавиши F4):

- 1) \$A\$1 при копировании не будут изменяться обозначение столбца и строки;
- 2) \$A1 при копировании не будет изменяться обозначение столбца;
- 3) A\$1 при копировании не будет изменяться обозначение строки.

Пример использования абсолютной и относительной адресации показан на рисунке 5.13.

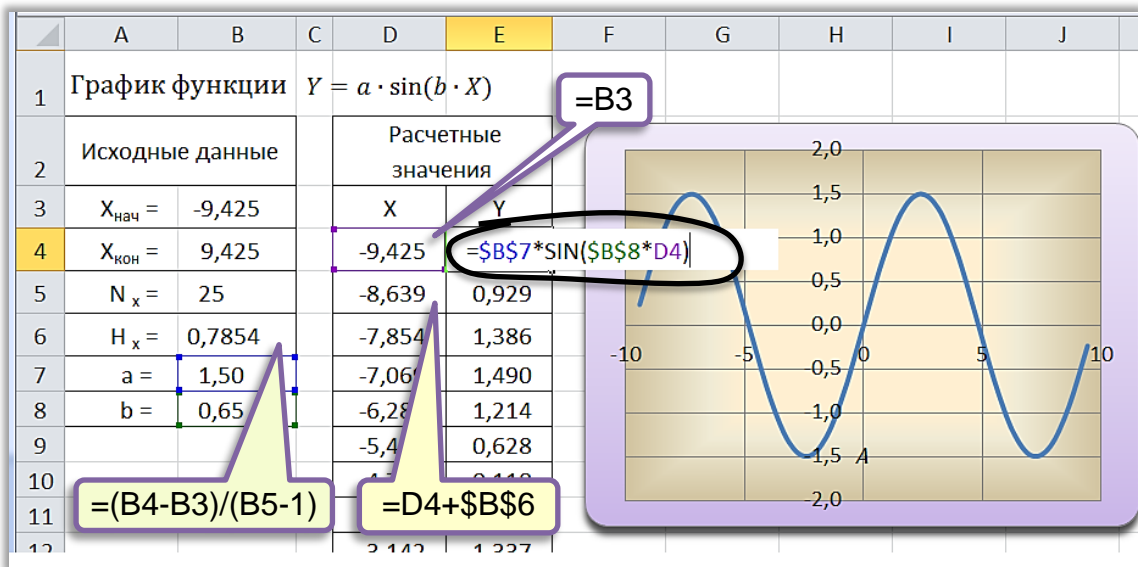


Рисунок 5.13 – Использование абсолютной и относительной адресации

Пример использования смешанной адресации показан на рисунке 5.14.

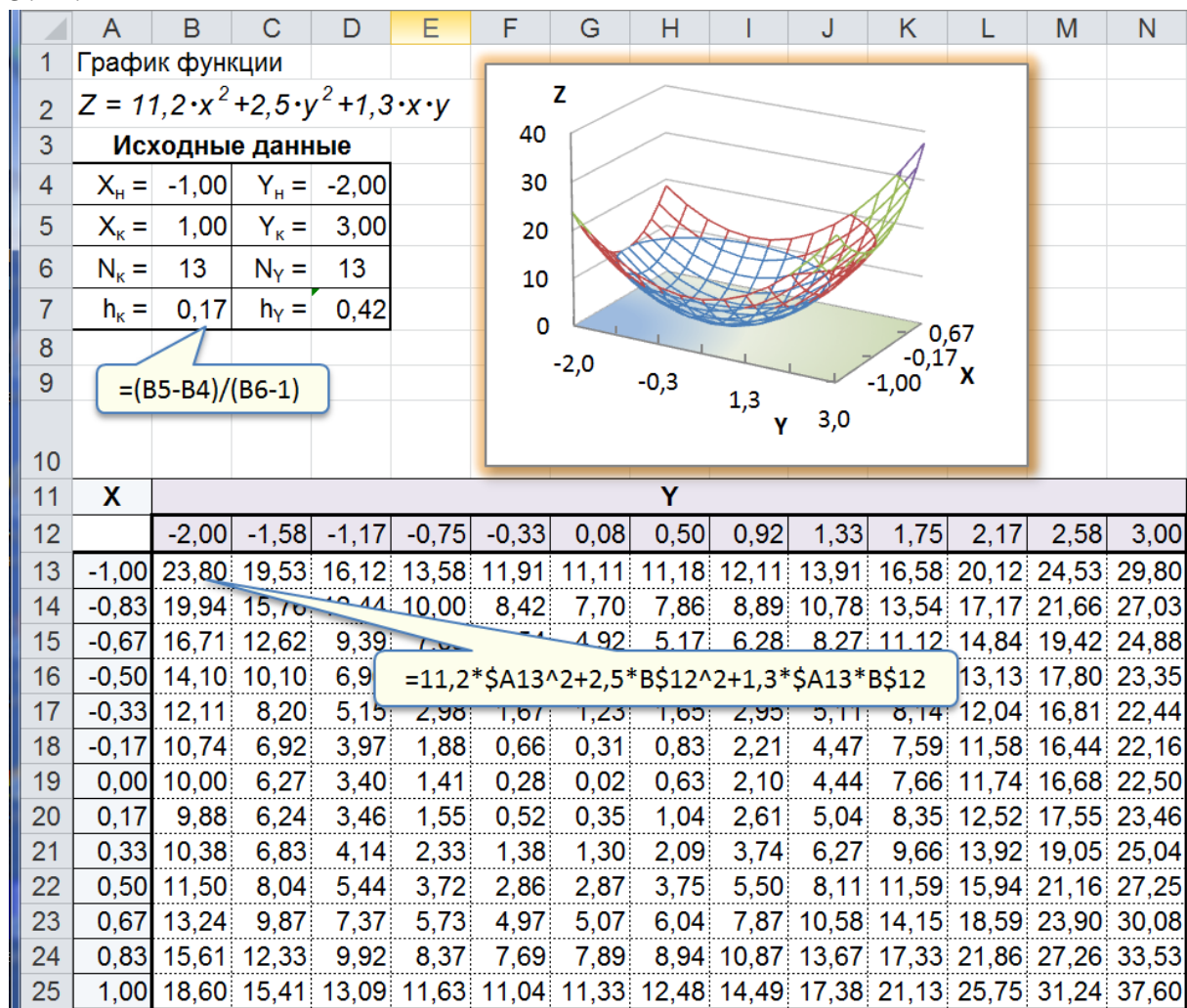


Рисунок 5.14 – Использование смешанной адресации

Система **Excel** имеет достаточно большой набор функций (см. вкладку **Формулы** на рисунке 5.15).

Функции сгруппированы по типу обрабатываемых ими данных:

- **финансовые** (БС, ВСД, КПЕР...);
- **логические** (ЕСЛИ, И, ИЛИ, НЕ, ИСТИНА ...);
- **текстовые** (ЗАМЕНИТЬ, КОДСИМВ, ЛЕВСИМВ, ПРАВСИМВ ...);
- **работы с датой и временем** (СЕГОДНЯ, ТДАТА, МИНУТЫ...);
- **ссылки и массивы** (АДРЕС, ВЫБОР, ТРАНСП ...);
- **математические** (ABS, ACOS, LN, LOG10...);
 - в том числе матричные МОБР, МОПРЕД, МУМНОЖ...;
- **статистические** (ДИСП, КВАДРОТКЛ, КОРРЕЛ...);
- **инженерные** (БЕССЕЛЬ.1, ВОСЬМ.В.ДВ, ДВ.В.ДЕС...) и пр.

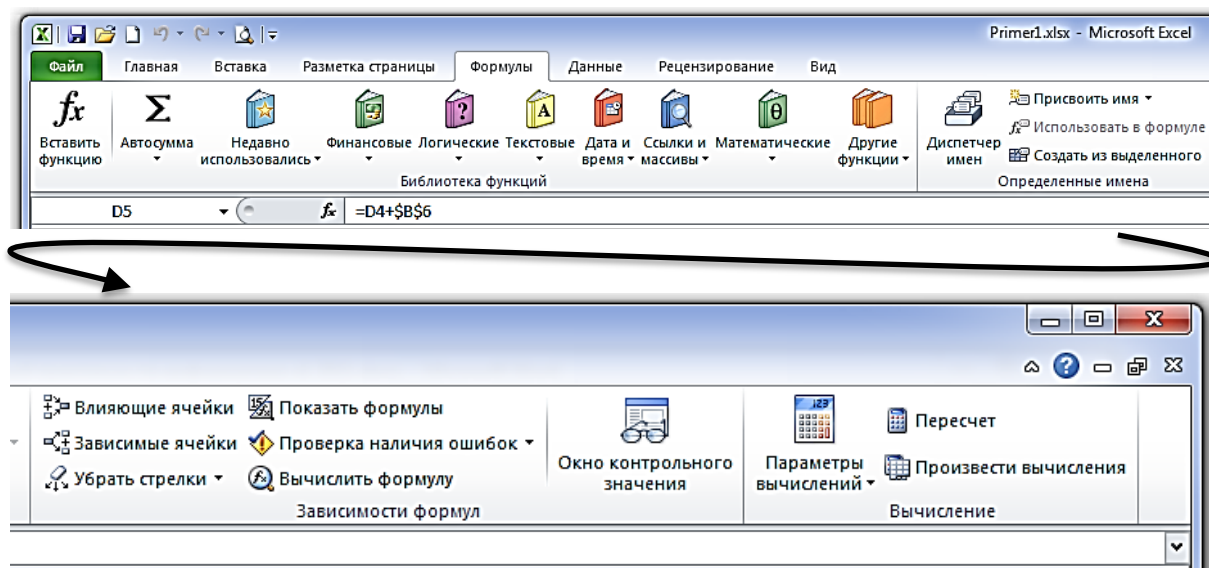


Рисунок 5.15 – Вкладка ленты **Формулы** Excel 2010 на широком экране

4 Использование математических функций и матричных операций

В системе Excel в категории **Математические функции** присутствует 63 наименования. Это функции тригонометрические, логарифмические, матричные, округления, преобразования, вычисления случайного числа, суммирования, перемножения и пр.

Кроме того в категории **Инженерные функции** присутствует несколько десятков наименований функций для работы с числами различных систем счисления, операций с комплексными числами и пр.

Названия функций для русской версии **Office** и для версий на других языках могут различаться. Поэтому при использовании функций всегда следует найти их в библиотеке системы (даже таких, как **SIN** и **LOG**).

Для этого в процессе написания формул можно использовать следующие варианты работы с функциями:

- 1) списки для кнопок на вкладке ленты **Формулы** (см. рисунок 5.15),
- 2) раскрывающийся список функций слева в строке формул,
- 3) значок f_x в той же строке,
- 4) контекстный список, появляющийся при написании первых букв функций.

В 1 – 3 варианте для выбранной функции появляется диалоговое окно задания аргументов функции (для варианта 3 при повторном выделении в формуле ранее написанной функции позволяет редактировать её в диалоговом окне), как показано на рисунке 5.16. Это же окно можно вызвать, если выбрать в контекстном списке функций (вариант 4) нужное название, а потом нажать на значок f_x в строке формул.

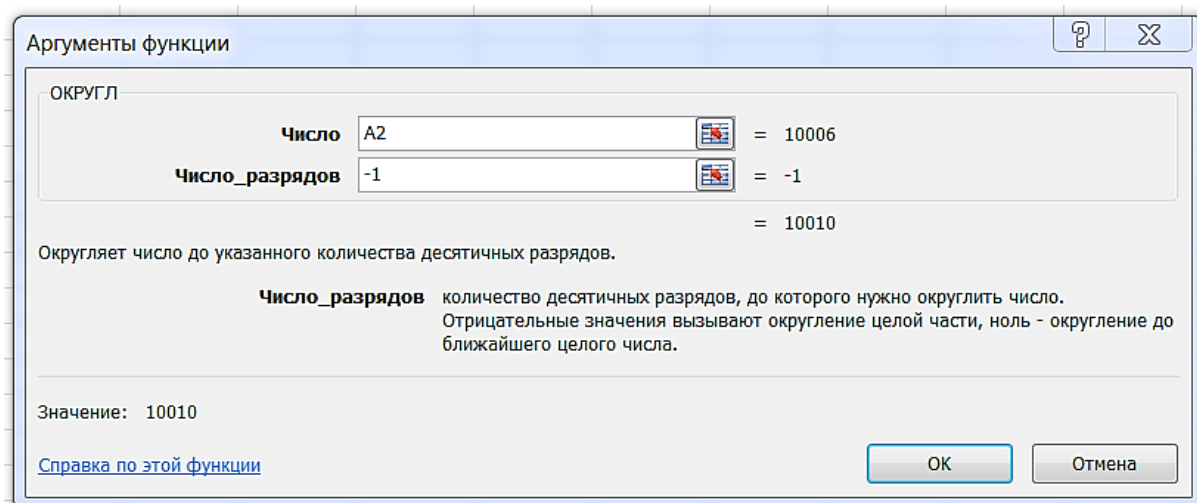


Рисунок 5.16 – Диалоговое окно задания аргументов для функции **ОКРУГЛ**

При использовании математических функций следует обращать внимание на наличие нескольких функций логарифмов с разным основанием и на единицы измерения углов в тригонометрических функциях (радианы).

Кроме основных функций с системой **Microsoft Office** поставляются надстройки для **Excel** – вспомогательные программы, служащие для использования дополнительных возможностей, они устанавливаются из окна **Параметры Excel**, раздел **Надстройки**.

Например, при проведении сложного статистического или инженерного анализа можно упростить процесс и сэкономить время, используя надстройку "Пакет анализа". Надстройку "Поиск решения" можно использовать для определения влияния ячеек на экстремальные значения зависимой ячейки.

Примеры использования математических функций

Пусть нам нужно вычислить **y** по математической формуле:

$$y = \frac{tg^2(2b + c)}{\sqrt{\left| \sqrt[3]{(a - b)^2} - \frac{e^{-b^2} \ln a}{a + \ln(2b + c)} \right|}}$$

причем в этой формуле значения **b** и **c** заданы в градусах.

В системе Excel для вычисления **y** следует в отдельных ячейках задать значения переменных **a**, **b** и **c**, т. к. эти значения встречаются в формуле несколько раз, а также с целью расчета по формуле с различными исходными данными. Кроме того эту сложную формулу во избежание ошибок следует разбить на части, например, следующим образом:

$$y1 = tg^2(2b + c)$$

$$y2 = \sqrt[3]{(a - b)^2}$$

$$y3 = \frac{e^{-b^2} \ln a}{a + \ln(2b + c)}$$

$$y = \frac{y1}{\sqrt{|y2 - y3|}}$$

$$y = \frac{tg^2(2b + c)}{\sqrt{\left| \sqrt[3]{(a - b)^2} - \frac{e^{-b^2} \ln a}{a + \ln(2b + c)} \right|}}$$

Расчетные формулы в системе Excel показаны на рисунке 5.17.

	A	B	C	D	E
1	Исходные данные				
2	a =	1,012			
3	b =	2,750	°		
4	c =	39,500	°		
6	y1 = TAN(РАДИАНЫ(2*B3+B4))^2 = 1,000				
7	y2 = ((B2-B3)^2)^(1/3) = 1,446				
8	y3 = EXP(-B3^2)*LN(B2)/(B2+LN(2*B3+B4)) = 4,764				
9	y = E6/КОРЕНЬ(ABS(E7-E8)) = 0,549				

Рисунок 5.17 – Пример расчета по сложной формуле

Достаточно часто в электронных таблицах используется функция суммирования чисел в столбце или строке (рисунок 5.18). Кроме обычной функции **СУММ** в системе Excel есть также:

- **СУММЕСЛИ** (вычисление суммы для заданного условия, пример использования показан на рисунке 5.18),
- **СУММЕСЛИМН** (суммирует числа в ячейках в соответствии с не-

- сколькими заданными условиями),
- **СУММКВ** (вычисляет сумму квадратов аргументов),
 - **СУММКВРАЗН** (вычисляет сумму квадратов разностей значений в двух массивах),
 - **СУММСУММКВ** (вычисляет сумму сумм квадратов соответствующих элементов двух массивов).

	A	B	C	D	E	F
1	Ведомость оборудования					
2	Группа	Наименование	=D3*E3	Цена, руб.	Количество, шт.	Сумма, руб.
3	1	Ноутбук Acer Aspire 3830TG-2414G50nbb i5-2410M/ 4Gb/500Gb/		28 871,00	3	86 613,00
4	1	Ноутбук Acer Aspire 4830T-2313G32Mnbb i3-2310M/3G/320Gb/		23 901,00	2	47 802,00
5	1	Ноутбук ASUS K52JU i3-380M/3Gb/320Gb/ATI MR 6370 512MB/D		16 990,00	5	84 950,00
6	1	Ноутбук ASUS K52JU i5-480M/4Gb/320Gb/HD6370 512Mb/DVDR		19 985,00	3	59 955,00
7	2	Системный блок (Intel Core i3 2100/ASRock H61M-VS/2GB/500GB/	=СУММ(F3:F12)	12 990,00	10	129 900,00
8	2	Системный блок (Intel Core i3 2100/ASRock H61M-VS/2GB/500GB/		12 990,00	5	64 950,00
9	2	Системный блок (Intel Core i5 2300/ASRock H61M-VS/4Gb/500GB/		16 500,00	3	49 500,00
10	3	ПЛАШЕТ Acer Iconia Tab A501 10.1" Tegra 250/1024Mb/32Gb/1.3		20 877,00	2	41 874,00
11	3	ПЛАШЕТ =СУММЕСЛИ(\$A\$3:\$A\$12;A15;\$F\$3:\$F\$12)		21 809,00	2	43 618,00
12	3	ПЛАШЕТ Lenovo K1-T0V32W Tegra2/ 10.1" / 1024Mb/ 32Gb/ 1.3		18 076,00	2	36 152,00
13	Итого					645 314,00
14	Итого по группе					
15	1					279 320,00
16	2					244 350,00
17	3					121 644,00

Рисунок 5.18 – Использование функций СУММ и СУММЕСЛИ

Особенность использования матричных операций: исходные данные и результат расчета – не одно число, а группа чисел. Исходные числа должны размещаться в соседних ячейках листа Excel, результирующие числа получают первоначально в одной ячейке, но затем разворачиваются на расчетную матрицу с использованием сочетания клавиш **CTRL+SHIFT+ENTER**, как показано на рисунке 5.19.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
2														
3														
4														
5														
6														
7														
8														
9														
10														
11														
12														
13														
14														
15														
16														
17														

$$\begin{cases} 2,00 \cdot X_1 - 4,00 \cdot X_2 - 2,00 \cdot X_3 = -2,00 \\ -0,12 \cdot X_1 - 2,00 \cdot X_2 + 3,00 \cdot X_3 = 3,00 \\ 3,00 \cdot X_1 + 1,50 \cdot X_2 - 5,00 \cdot X_3 = 4,00 \end{cases}$$

формат + 0,00; - 0,00

=МОБР(L3:N5)

Решение системы линейных уравнений по матричной формуле $X = A^{-1} \cdot B$

=МУМНОЖ(L8:N10;I3:I5)

Решение системы

$X_1 =$	4,21
$X_2 =$	1,51
$X_3 =$	2,18

Выделите диапазон, начиная с ячейки, содержащей матричную формулу. Нажмите клавишу F2, а затем нажмите клавиши CTRL+SHIFT+ENTER

Рисунок 5.19 – Использование матричных функций МОБР и МУМНОЖ

Учебное задание

а) с использованием автозаполнения занести на *Лист 1* электронной таблицы списки дней недели и месяцев (сокращенные обозначения и полные), дат по дням, по рабочим дням, по месяцам и по годам. Показать, как будет выглядеть при копировании текст: *Группа 1, 1-я пара, 1-й семестр, 1-я неделя.*

б) выполнить на *Листе 2* расчет значений аргумента и функции, как показано на рисунке 5.11, для варианта задания в таблице, соответствующего номеру Вашего компьютера:

Вариант	Функция	Диапазон изменения X		Значение констант	
		X _{нач.}	X _{кон.}	a	b
1	$Y = a^{-bx^2} / b^{-ax^2}$	-1,5	1,5	3,45	12,34
2	$Y = x^3 - 2a \cdot x - \lg(x) + b$	0,1	2,2	1,80	5,20
3	$Y = x^3 - 2,4 \cdot a \cdot x^2 + \lg(20,3 - x) - b$	-2	2	0,91	0,68
4	$Y = a \sqrt{\operatorname{tg}(bx)}$	0°	85°	0,90	0,15
5	$Y = \ln(x \cdot a) / e^{x+b}$	0,1	4	2,50	0,58
6	$Y = x^3 + a \cdot x^2 + b$	-3	3	1,50	1,11
7	$y = (a \cdot e^x) / (b \cdot e^{-x})$	-1	1	0,89	1,00
8	$Y = x^5 + b \cdot x^2 + \sqrt{x^2 + a}$	-1,2	1,0	3,71	0,50
9	$Y = \operatorname{tg}(x+a) \cdot \sin(x-b)$	-0,5 радиан	1,0 радиан	0,20	1,10
10	$Y = \lg(x^2 + a \cdot b)$	-1000	1000	1,61	5,39

в) выполнить на *Листе 3* расчет значения Y для варианта задания в таблице, соответствующего номеру Вашего компьютера. Значения **a**, **b** и **c** задать в таблице исходных данных. Единицы измерения угла в таблице заданы в радианах. Сравнить полученное значение с результатом в таблице:

Вариант	Выражение	Значение переменных			Результат
		a	b	c	
1)	$y = \sqrt{a(b^{1/3} + 17)} + \frac{a^5 + a^2 + \sqrt{c}}{2b}$	1,5	10,2	10,034	6,00
2)	$y = \sqrt{3(a+b)/(b^3+17)} + \frac{a^2 + \sqrt[3]{c}}{(5-a)}$	13,5	0,92	1,05	-20,00
3)	$y = \lg \frac{a + \sin(b)}{\cos 2a} + \sqrt[4]{5 + \sqrt{2+c}}$	0,785	1,5708	3,777	5,00
4)	$y = \operatorname{tg} \frac{a - 24\sqrt{2b+3}}{2a^{2/3} + \lg(3b)}$	9,01	7,7058	-	0,05
5)	$y = \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2+a}}} + \sin \frac{13-b}{c^5}}$	7,0	13,0	0,8	2,00
6)	$y = \sqrt[3]{3 + \sqrt[3]{3 + \frac{\operatorname{tg}^2(a)}{32+b}}} + \lg \frac{a^3}{a+b+2c}$	3,1415	96,6	18,4	1,00
7)	$y = \frac{\operatorname{tg}^2(a) - \operatorname{tg}^2(2b+c)}{\sqrt{34 - \frac{\lg(a)}{a + \lg(2b+c)}}$	1,5	1,725	5,2425	34,00
8)	$y = \sqrt{1 - \frac{\sin^2(a) + \cos^2(b)}{a+b+c}} \frac{1}{(a+b+c)^{2/3}}$	3,1416	1,5708	-1,884	0,50
9)	$y = \operatorname{tg} \frac{\lg^2(a+b+c) + \sin(a^2)}{\sqrt{2 + \frac{14-b}{2c}}}$	0,5	6,385	4,201	1,00
10)	$y = \frac{b^2 - 4ac}{\sin^2(a) + \sin^2\left(\frac{2\sqrt{a}}{b-c}\right)}$	9,5	-4,51	2,2093	-100,00

г) выполнить на *Листе 4* решение системы линейных уравнений в соответствии с примером, показанным на рисунке 5.19.

Лабораторная работа № 6. Microsoft Excel 2010. Использование функций для работы с данными типа Текст и Дата/время

Цель работы

Освоение приемов выполнения расчетов и оформления результатов в электронной таблице при работе с текстом, датами и временем.

Учебные вопросы

1. Использование текстовых функций
2. Работа с информацией типа дата/время

Краткое изложение учебного материала по теме

Наиболее знакомы всем математические функции, о которых говорилось в предыдущей лабораторной работе.

Однако пользователи должны уметь использовать и функции работы с текстом, датами и временем.

Использование текстовых функций

Система Excel 2010 содержит 24 функции в категории **Текстовые**. Среди них операции объединения в строку данных разного типа (можно использовать символ & или функцию СЦЕПИТЬ), функции преобразования типа (ЗНАЧЕН, КОДСИМВ, СИМВОЛ, ТЕКСТ), поиска подстроки в строке (ПОИСК, НАЙТИ), замены части строки на другую (ЗАМЕНИТЬ, ПОДСТАВИТЬ, СЖПРОБЕЛЫ), выделения подстроки из строки (ЛЕВСИМВ, ПРАВСИМВ, ПСТР), замены регистра букв (ПРОПИСН, ПРОПНАЧ, СТРОЧН) и пр. Описания функций и их аргументов можно увидеть в диалоговом окне задания аргументов функции, подобном примеру на рисунке 5.16.

Если Вы знаете имя функции, можно воспользоваться всплывающим списком, который появится, если после символа равно (=) написать первые буквы имени функции (рисунок 6.1).

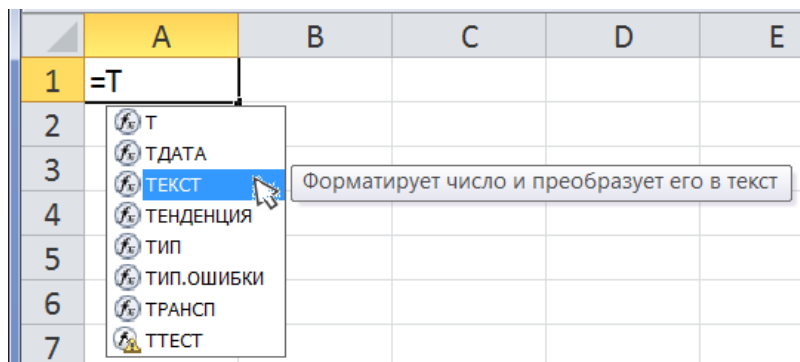



Рисунок 6.1 – Контекстный всплывающий список функций

После выбора имени функции (двойным щелчком мыши) можно нажать на кнопку  слева от строки формул для вызова диалогового окна задания аргументов функции.

Примеры использования некоторых текстовых функции приведены далее.

Чтобы объединить текст из нескольких ячеек в одно значение, можно воспользоваться символом конкатенации (амперсанд &) или функцией СЦЕПИТЬ (рисунок 6.2).

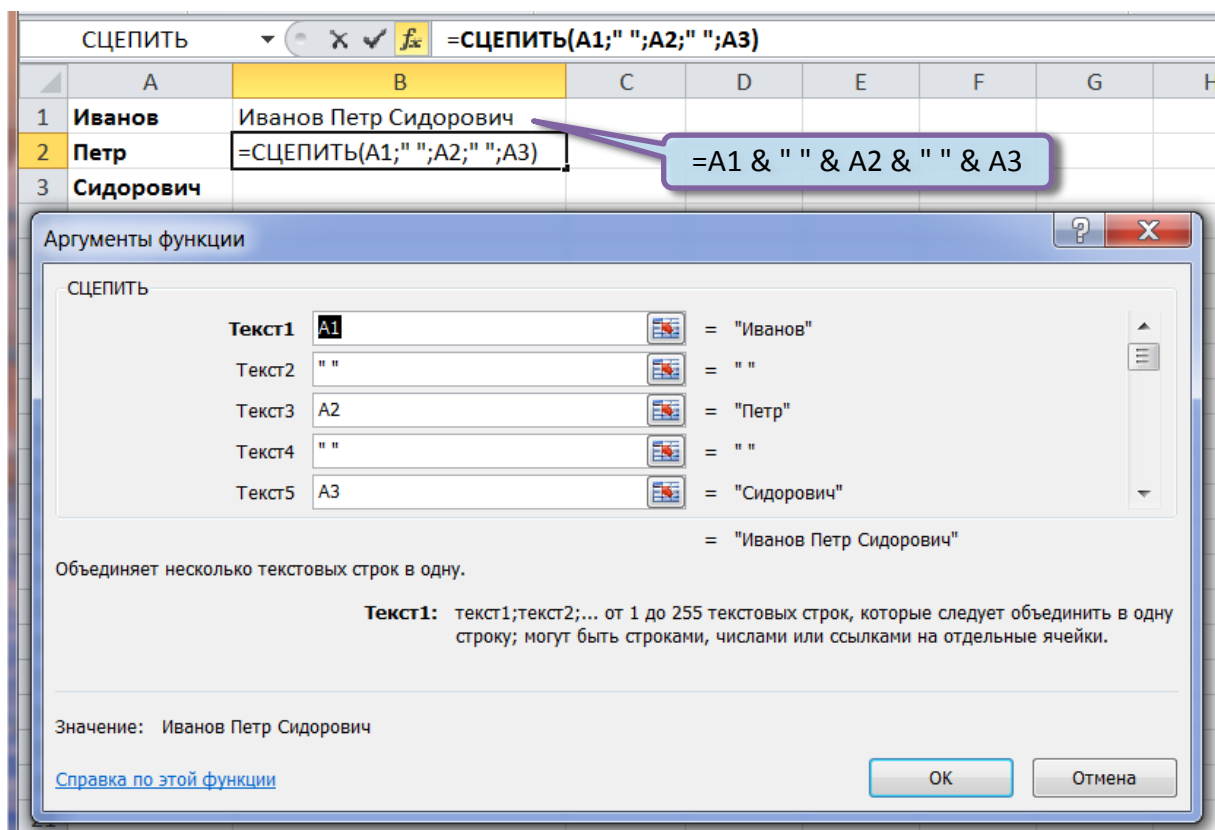


Рисунок 6.2 – Использование операции конкатенации и функции *СЦЕПИТЬ*

В первом случае функция в ячейке B1 будет выглядеть, как `=A1 & " " & A2 & " " & A3`, во втором случае функция в ячейке B2: `=СЦЕПИТЬ(А1;" "; А2;" "; А3)`.

Для преобразования данных разного типа при объединении их в текстовую строку следует пользоваться функцией `ТЕКСТ(<значение>; <формат>)`. Пример использования показан на рисунке 6.3.

	А	В	С	Д	Е	Ф	Г	Н	І
1	ФИО	Дата рождения							
2	Иванов И. И.	13.11.1991	Иванов И. И. родился в 1991 году						
3									
4	=ТДАТА()								
5	08.11.2011 18:09		Сегодня 08.11.2011, вторник						
6				Сейчас 18 час. 09 мин.					
7									
8									
9									
10									
11									
12									

Рисунок 6.3 – Использование функции ТЕКСТ

Другая задача, более сложная: предположим, в столбце **А** написаны фамилия, имя и отчество группы людей, а в столбце **В** нужно получить их фамилию и инициалы (рисунок 6.4).

Для решения этой задачи сначала в ячейку **С1** напишем формулу поиска первого пробела в тексте ячейки **А1**: =ПОИСК(" ";А1;1).

В ячейку **Д1** напишем формулу поиска второго пробела в строке ячейки **А1**: =ПОИСК(" ";А1;С1+1).

В ячейку **В1** напишем формулу получения фамилии с инициалами: =СЦЕПИТЬ(ПСТР(А1;1;С1); ПСТР(А1;С1+1; 1); ". ";ПСТР(А1;Д1+1; 1);".").

В этой формуле объединяются в одну строку фрагменты:

- 1) с первого символа по первый пробел /ПСТР(А1;1;С1) – выделить подстроку из А1 с позиции 1, количество символов С1/;
- 2) один символ после пробела;
- 3) точка и пробел;
- 4) один символ после второго пробела;
- 5) точка.

Далее копируем за один прием ячейки **В1**, **С1** и **Д1** в строки 2 – 5.

Получаем результат, показанный на рисунке 6.3.

	А	В	С	Д
1	Корельская Галина Юрьевна	Корельская Г. Ю.	11	18
2	Медведский Денис Сергеевич	Медведский Д. С.	11	17
3	Мурина Мария Вячеславовна	Мурина М. В.	7	13
4	Неманова Юлия Александровна	Неманова Ю. А.	9	14
5	Нистор Олеся Николаевна	Нистор О. Н.	7	13

Рисунок 6.4 – Использование функций поиска и выделения подстроки

2 Работа с информацией типа дата/время

Ранее было показано, что некоторые операции по работе с датами можно выполнять с использованием автозаполнения (см. рисунок 5.5).

Как указывалось в лабораторной работе 5, дата/время – единый

внутренний формат системы. Представление информации на листе таблицы зависит от заданного формата.

Количество функций для работы с информацией типа дата/время в системе Excel более двух десятков.

Например, чтобы показать на листе Excel текущую дату, можно использовать функцию СЕГОДНЯ(), чтобы получить не только текущую дату, но и время, следует использовать функцию ТДАТА(). По умолчанию для первой функции задан формат Дата в формате операционной системы, однако, можно задать и формат с показом даты и времени (время будет равно 00:00). Аналогично и для второй функции.

С использованием формата ячеек для типа дата/время можно показать даже название дня недели и месяца, как показано на рисунке 6.5.

	A	B	C	D	E
1	08.11.2011	← =СЕГОДНЯ()		08.11.2011 00:00	← =A1, формат ячейки ДД.ММ.ГГГГ чч:мм
2	08.11.2011 18:45	← =ТДАТА()		08.11.11	← =A1, формат ячейки ДД.ММ.ГГ
3	2011	← =ГОД(A1)		2011	← =A1, формат ячейки ГГГГ
4	11	← =МЕСЯЦ(A1)		11	← =A1, формат ячейки ММ
5	8	← =ДЕНЬ(A1)		8	← =A1, формат ячейки Д
6	3	← =ДЕНЬНЕД(A1)		ноя	← =A1, формат ячейки ММММ
7	2	← =ДЕНЬНЕД(A1;2)		Ноябрь	← =A1, формат ячейки МММММ
8	вторник	← =ТЕКСТ(A1;"ДДДД")		Н	← =A1, формат ячейки ММММММ
9	Вт	← =ТЕКСТ(A1;"ДДД")		вторник	← =A1, формат ячейки ДДДД
10	Ноябрь	← =ТЕКСТ(A1;"ММММ")		Вт	← =A1, формат ячейки ДДД

Рисунок 6.5 – Использование различных функций и форматов типа дата/время

Ранее было также показано, как объединить в одну строку данные текстовые и даты/времени (см. рисунок 6.3), в этом случае можно использовать все форматы даты/времени, показанные на рисунке 6.5.

Представление результата вычисления разности между двумя датами также зависит от формата ячейки. При использовании формата *Общий* будет показана разница в днях между датами (рисунок 6.6).

	A	B	C	D	E
1	Дата 1	Дата 2	= B2 - A2	Единицы	Формат
2	28.02.2011	01.03.2012	367	дней	Общий
3			01	лет	ГГ
4			01	мес.	ММ
5			01	дн.	ДД
6	Функции в колонке D →			Значение	
7	(разница в целых годах, месяцах и днях)			функции	
8	= РАЗНДАТ(A2;B2;"Y") & " лет "			1 лет	
9	= РАЗНДАТ(A2;B2;"YM") & " мес. "			0 мес.	
10	= РАЗНДАТ(A2;B2;"MD") & " дн. "			2 дн.	

Рисунок 6.6 – Использование различных форматов и функции РАЗНДАТ при вычислении разности двух дат

На практике встречается задача представить стаж работника, как количество целых лет + месяцев + дней, эту задачу можно решить в системе Excel с использованием недокументированной функции РАЗНДАТ (сохранена для совместимости со старыми версиями), как это показано на рисунке 6.6.

Т. к. в справке Excel нет данных по этой функции, ниже приведено описание ее синтаксиса:

РАЗНДАТ(нач_дата; кон_дата; единица)

Единица	Возвращаемое значение
"Y"	Разница в полных годах
"M"	Разница в полных месяцах
"D"	Разница в полных днях
"YD"	Разница в днях с начала года без учета лет
"MD"	Разница в днях без учета месяцев и лет
"YM"	Разница в полных месяцах без учета лет

В системе Excel присутствует 4 функции для определения количества рабочих дней в интервале и, наоборот, даты отстоящей вперед или назад на заданное количество рабочих дней: ЧИСТРАБДНИ, ЧИСТРАБДНИ.МЕЖД и РАБДЕНЬ, РАБДЕНЬ.МЕЖД. Для примера приведем синтаксис 2-й функции (в описании синтаксиса в квадратных скобках – необязательные аргументы):

ЧИСТРАБДНИ.МЕЖД(нач_дата, кон_дата, [выходной], [праздники])

В этой функции необязательный аргумент **выходной** может быть задан числом из первого столбца таблицы 6.1 или строкой второго столбца этой таблицы.

Необязательный аргумент **праздники** – набор из одной или нескольких дат, которые необходимо исключить из календаря рабочих дней. Значение аргумента должно быть диапазоном ячеек, содержащих даты, или константой массива, включающей порядковые значения, которые представляют даты. Порядок дат или значений может быть любым. Пример использования функции показан на рисунке 6.7.

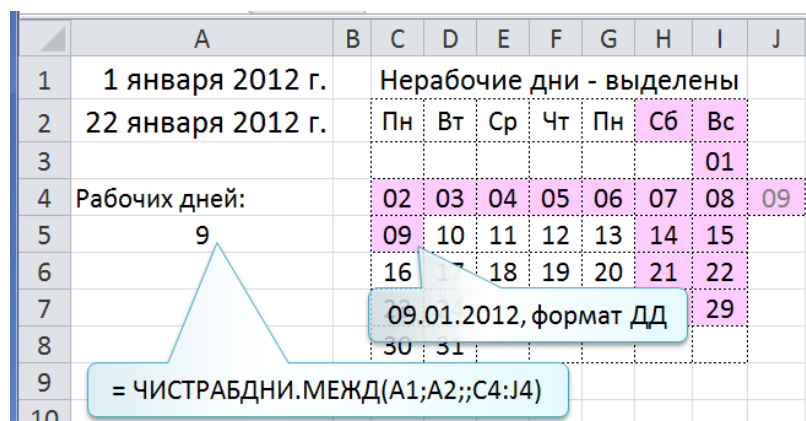


Рисунок 6.7 – Использование функции ЧИСТРАБДНИ.МЕЖД

Таблица 6.1 – Значения аргумента **выходной** для функций **ЧИСТРАБДНИ.МЕЖД** и **РАБДЕНЬ.МЕЖД**

Номер выходного дня	Выходные дни
1 или опущен	Суббота, воскресенье
2	Воскресенье, понедельник
3	Понедельник, вторник
4	Вторник, среда
5	Среда, четверг
6	Четверг, пятница
7	Пятница, суббота
11	Только воскресенье
12	Только понедельник
13	Только вторник
14	Только среда
15	Только четверг
16	Только пятница
17	Только суббота

Учебное задание

а) Выполнить вариант задания в таблице, соответствующий номеру Вашего компьютера, с использованием функций для работы с текстовой информацией:

№	Задание
1	Для данного предложения, написанного в ячейке A1, показать в строке 2 таблицы коды первых двадцати символов. Использовать одну формулу (для вычисления номера символа использовать функцию СТОЛБЕЦ)
2	Показать в строке 1 символы с кодами от 64 до 84. Использовать одну формулу (для задания номера символа использовать функцию СТОЛБЕЦ)
3	Определить позиции всех пробелов в этом предложении. Использовать одну формулу для поиска пробелов, начиная со второго.
4	Разделить данное предложение на отдельные слова, исходное предложение написать в ячейке A1. Со второй строки в колонке A определить номер каждого пробела (использовать одну формулу для поиска первого пробела и вторую – для всех остальных), в колонке B показать слова.
5	Определить количество символов в каждом слове данного предложения (подсказка: сначала в одной строке найти позиции пробелов, в другой строке – количество букв)
6	Для данного предложения выполнить следующие операции: а) преобразовать все его символы в строчные; б) преобразовать все символы в прописные; в) вариант «б» преобразовать в вид исходного предложения
7	Заменить в данном предложении первую фразу «в данном предложении» на фразу «в данном тексте»
8	Заменить в данном предложении 20 символов, начиная с 10-го на фразу "с использованием текстовых функций"
9	Найти в данном предложении слово «проверка» и заменить его на «функции»
10	Используя функции работы с текстом ЛЕВСИМВ, ПРАВСИМВ, НАЙТИ и ЗАМЕНИТЬ поменять местами первое слово и последнее в этом предложении

б) Выполнить вариант задания в таблице, соответствующий номеру Вашего компьютера, с использованием функций для работы с информацией типа дата/время:

№	Задание
1	Определить текущую дату на часах компьютера, с использованием функции ДАТА, прибавьте к ней 1 год, 5 месяцев и 35 дней и определите название дня недели для полученной даты
2	Определите текущее времени с точностью до секунд. Вычислите количество секунд, оставшихся до конца суток
3	Вычислите количество дней, часов, минут и секунд, прошедших с начала 21 века до текущего момента, который взять с часов компьютера.
4	Рассчитайте стаж работника – количество целых лет, кроме того целых месяцев и дней (например, 10 лет 1 месяц и 1 день) начиная с даты 1.09.2001 до текущей даты, которую определить по часам компьютера.
5	Рассчитайте количество рабочих дней при пятидневной рабочей неделе с 1.02.2012 по 30.04.2012 с учетом того, что 23.02.2012 – праздничный день
6	Рассчитайте количество выходных и праздничных дней при шестидневной рабочей неделе с 1.09.2012 по 31.12.2012 (4.11.2012 – праздник, выходной перенесен на 05.11.2012)
7	Рассчитайте количество отработанных часов за период с 1.07.2012 по 30.11.2012 (по пятидневке, рабочий день 8 часов) с учетом того, что в этом периоде 04.11.2012 – праздник, выходной перенесен на 05.11.2012
8	Рассчитайте дату, отстоящую от 1.07.2012 на 60 рабочих дней при шестидневной рабочей неделе, и определите для нее название дня недели
9	Определите название дня недели через 100 дней от текущей даты
10	Определите название дня недели через 10 месяцев от текущей даты (используйте функцию ДАТАМЕС)

Лабораторная работа № 7. Microsoft Excel 2010. Построение диаграмм

Цель работы

Освоение приемов построения и оформления диаграмм.

Учебные вопросы

1. Построение диаграмм
2. Настройка параметров диаграмм

Краткое изложение учебного материала по теме

1 Построение диаграмм

В системе Excel существует достаточно большое количество типов диаграмм, часть из них представлены на вкладке ленты *Вставка* в группе *Диаграммы* (рисунок 7.1).

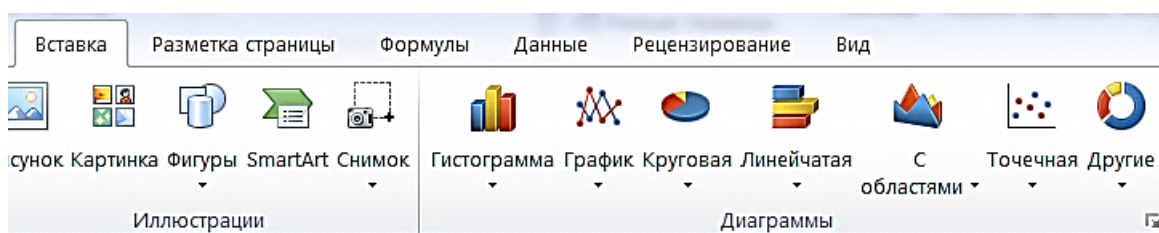


Рисунок 7.1 – Группа *Диаграммы* на вкладке ленты *Вставка*

Для кнопки *Другие* на рисунке 7.1 открывается панель с выбором еще пяти типов диаграмм: *Биржевая*, *Поверхность*, *Кольцевая*, *Пузырьковая*, *Лепестковая*. Можно также открыть диалоговую панель всех типов диаграмм, аналогичную той, которая ранее была показана при работе с диаграммами в системе Word (см. рисунок 4.10).

При этом среди всех типов диаграмм только *Точечная* и *Поверхность* показывают зависимости типа $Y=F(X)$ и $Z=F(X, Y)$ (пример расчета и построения показан ранее на рисунках 5.13 и 5.14). Другие типы показывают данные, распределенные с равномерным шагом по меткам времени или специальные виды (например, *Биржевые*).

Например, если мы хотим показать сравнительный график изменения температуры за ноябрь месяц 2010 и 2011 года, мы заносим эти данные в таблицу (см. рисунок 7.2). Далее выделяем ячейки с надписями осей и числовыми данными температур (начиная с ячейки с текстом *Дата*, вниз и вправо) и выбираем на вкладке ленты *Вставка* в группе *Диаграммы* тип *График* (рисунок 7.2), *Гистограмма* или *Линейчатая* (рисунок 7.3).

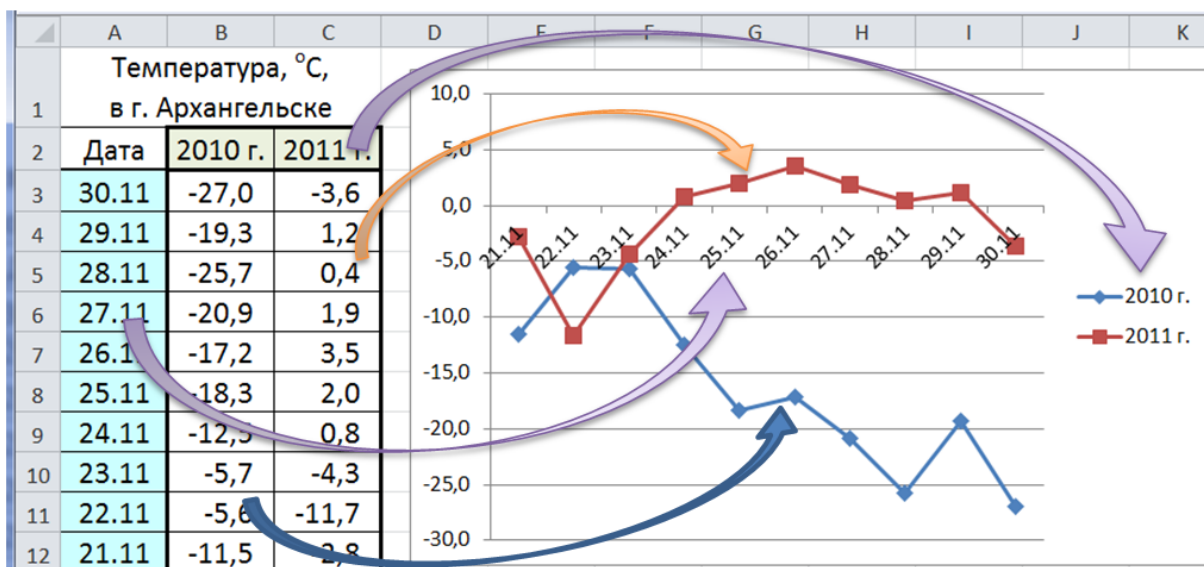


Рисунок 7.2 – Исходные данные и начальный вид диаграммы типа График

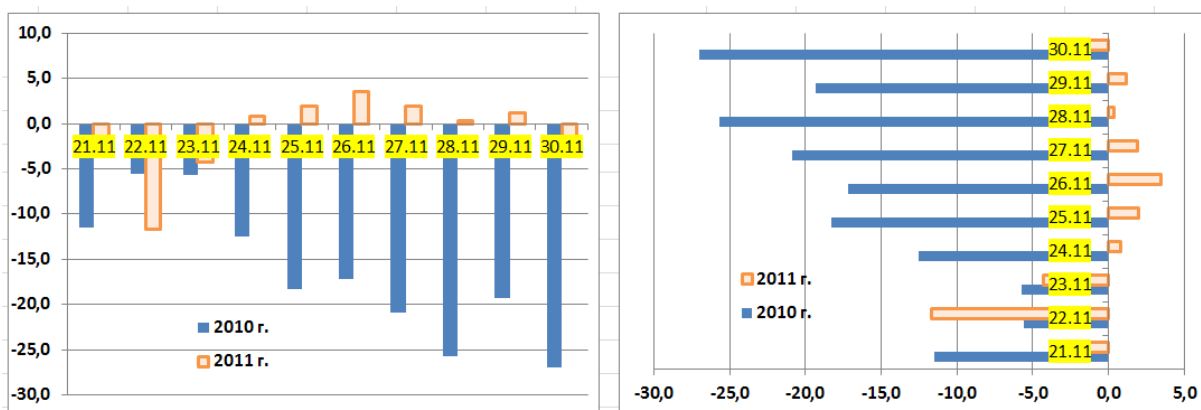


Рисунок 7.3 – Вид диаграмм типа Гистограмма и Линейчатая

2 Настройка параметров диаграмм

Покажем на примере диаграммы рисунка 7.2, как выполнить оформление диаграммы в соответствии с требованиями нормативных документов, например, стандарта на правила оформления студенческих работ. Для изменения параметров диаграмм следует использовать кнопки вкладки ленты Макет, либо диалоговое окно Формат..., которое можно вызвать кнопкой Формат выделенного той же ленты, либо двойным щелчком мыши на элементе диаграммы (это окно можно не закрывать, если будут изменяться параметры других областей диаграммы).

Внесем следующие дополнения и изменения:

- 1) нанесем названия осей на диаграмму (контекстная вкладка ленты Макет кнопка Названия осей);
- 2) зададим числовой формат с числом десятичных знаков = 0 для вертикальной оси;
- 3) уберем штрихи у вертикальной и горизонтальной осей, задав параметр Основные – нет в том же диалоговом окне; для диаграммы

- типа График зададим сетку – вертикальную и горизонтальную;
- 4) зададим цвет границы для *области построения* – Сплошная линия, для *области диаграммы* – Нет линий (в том же диалоговом окне);
 - 5) оформим диаграмму в черно-белых цветах для печати на монохромном принтере;
 - 6) для диаграммы типа График зададим Тип линии – Сглаженная линия;
 - 7) расшифровку обозначений (легенду) разместим под диаграммой в одну строку.

После внесения всех изменений диаграмма будет выглядеть, как показано на рисунках 7.4 и 7.5.

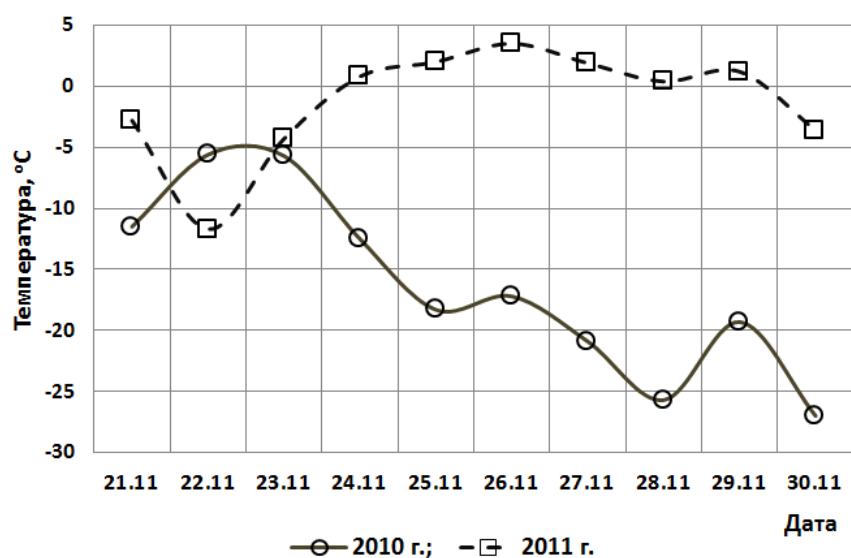


Рисунок 7.4 – Диаграмма типа График после настройки параметров

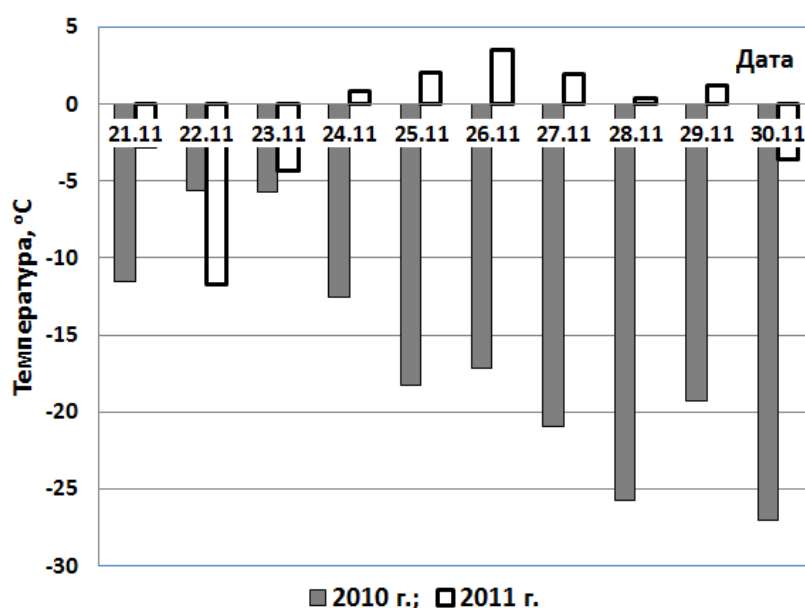


Рисунок 7.5 – Диаграмма типа Гистограмма после настройки параметров

Учебное задание

а) построить зависимость $Y=F(x)$ для данных, рассчитанных ранее в лабораторной работе № 5;

б) для функции $Z = F(x, y)$ выполнить расчет значений x , y и Z , как показано на рисунке 5.14 (количество точек по x и y – не менее 15, вариант задания в таблице выбрать в соответствии с номером Вашего компьютера) и показать графическое представление данных:

Вариант	Уравнение	Диапазон изменения x		Диапазон изменения y	
		$x_{\text{нач.}}$	$x_{\text{кон.}}$	$y_{\text{нач.}}$	$y_{\text{кон.}}$
1	$Z = x^3 - y^3$	-5	5	-5	5
2	$Z = x^3 - y^2$	-2	5	-11	11
3	$Z = x^2 + y^2$	-2	3	-2	3
4	$Z = \sqrt{x} + \sqrt{y}$	0,01	5	0,01	5
5	$Z = \sqrt{x^2 + y^2}$	-5	5	-5	5
6	$Z = x^2 - y^3$	-3	10	-3	3
7	$Z = \sqrt[3]{x} - \sqrt[3]{y}$	-1	1	-1	1
8	$Z = x^2 / \sqrt[3]{y}$	-3	0	1	5
9	$Z = x^3 + y^2$	-3	3	-5	5
10	$Z = x^2 / y$	-1	1	1	5

в) для № строки в таблице, соответствующей номеру Вашего компьютера, показать диаграммы 2-х типов: График и Гистограмма:

Население, тыс. чел., за период 1995 – 2010 гг.								
№	Наименование	1995	2000	2005	2007	2008	2009	2010
1	Архангельская обл.	1498	1390	1305	1280	1272	1262	1254
2	Белгородская обл.	1455	1502	1512	1514	1519	1525	1530
3	Брянская обл.	1469	1423	1346	1317	1309	1300	1292
4	Владимирская обл.	1630	1576	1487	1459	1449	1440	1430
5	Вологодская обл.	1340	1300	1245	1228	1223	1218	1214
6	Воронежская обл.	2495	2441	2334	2295	2280	2270	2262
7	Ивановская обл.	1260	1195	1115	1088	1080	1073	1067
8	Магаданская обл.	267	202	175	169	166	163	161
9	Московская обл.	6672	6628	6630	6646	6673	6713	6753
10	г. Москва	9085	9933	10407	10443	10470	10509	10563

Лабораторная работа № 8. Microsoft Excel 2010. Технологический расчет

Цель работы

Освоение приемов расчета и оформления одного из узлов технологического процесса.

Учебные вопросы

1. Оформление таблицы исходных данных
2. Условное изображение технологического узла
3. Написание расчетных уравнений и вывод расчетной формулы
4. Расчет и оформление результатов

Краткое изложение учебного материала по теме

Темой курсовых работ и проектов по специальным дисциплинам часто бывает расчет технологического участка, состоящего из нескольких связанных между собой производственных узлов. Исходные данные для расчета могут быть заданы по количеству готовой продукции или перерабатываемого сырья. Требования по оформлению расчетов могут несколько различаться, но использование системы Excel позволяет выполнить и оформить все расчеты в необходимом виде. В цепочке технологических узлов часть исходных данных может задаваться числовыми величинами (например, концентрации, определяемые допустимыми параметрами работы оборудования или справочные данные по теплоемкости потоков), другие же могут образовываться как расчетные значения, поступающие с предыдущих стадий расчета. При правильной организации расчетов изменение одной из задаваемых исходных величин вызывает перерасчет по всем формулам, связанным между собой в технологическую цепочку, даже для расположения расчетов на разных листах Excel.

Расчет материального баланса многих технологических узлов сводится к составлению и решению системы линейных уравнений. Для двух уравнений и двух неизвестных величин расчет можно выполнить методом подстановки, но для большего количества уравнений следует использовать решение матричным методом, как показано ранее на рисунке 5.19.

Оформление таблицы исходных данных

Как сказано выше, в таблице исходных данных могут присутствовать ссылки на ячейки листов Excel расчета предыдущих (или, наоборот, последующих) узлов и числовые значения.

Например, при расчете узла сгущения целлюлозной суспензии исходными данными могут быть:

– поток уходящей с узла целлюлозной массы, т/ч;

- концентрации целлюлозы: в поступающем потоке, в потоке сгущенной целлюлозы и в очищенной воде (с присутствием небольшого количества целлюлозы), % (в расчете на абсолютно сухое вещество);
- потери целлюлозы, в данном расчете невелики и их можно не учитывать.

Таблица исходных данных будет выглядеть следующим образом.

Таблица 8.1 – Исходные данные узла сгущения

Наименование показателя	Обозначение	Единица измерения	Величина	Обоснование
Поток уходящей с узла целлюлозной массы	G_2	т/ч	50,05	известно из расчета другого узла
Концентрация целлюлозы:				
– в поступающем потоке	c_1	%	2,50	принимаем
– после сгущения	c_2	%	12,00	задано при расчете другого узла
– в отобранной воде	c_3	%	0,05	принимаем

Условное изображение технологического узла

Изобразим в простейшем виде схему узла с указанием поступающих и уходящих с него потоков.

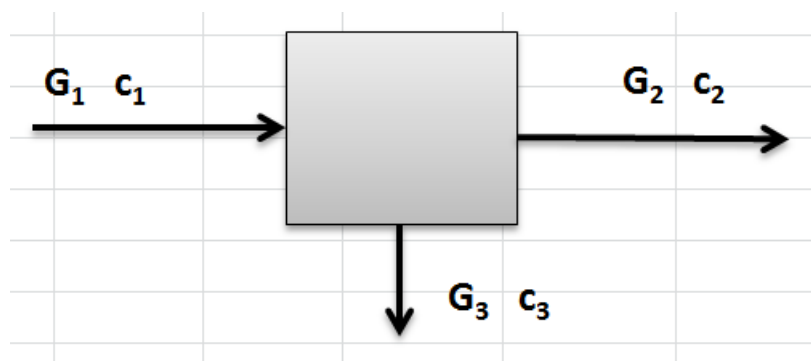


Рисунок 8.1 – Схема узла сгущения

3 Написание расчетных уравнений и вывод расчетной формулы

Материальный баланс узла состоит из двух уравнений: общего и частного баланса.

Общий баланс: сумма поступающих потоков равна сумме уходящих потоков, для данного узла:

$$G_1 = G_2 + G_3$$

Частный баланс: количество поступающей целлюлозы равно сумме целлюлозы в уходящих потоках:

$$M_1 = M_2 + M_3$$

где

$$M_i = G_i \cdot c_i / 100$$

Таким образом, получаем следующую систему из двух линейных уравнений, в которой неизвестные величины G_1 и G_3 :

$$\begin{cases} G_1 = G_2 + G_3 \\ G_1 \cdot c_1 / 100 = G_2 \cdot c_2 / 100 + G_3 \cdot c_3 / 100 \end{cases}$$

Если умножить каждый член второго уравнения на 100, получим систему:

$$\begin{cases} G_1 = G_2 + G_3 \\ G_1 \cdot c_1 = G_2 \cdot c_2 + G_3 \cdot c_3 \end{cases}$$

Данную систему можно решить методом подстановки или матричным методом, перенеся известные значения в правую часть:

$$\begin{cases} G_1 - G_3 = G_2 \\ G_1 \cdot c_1 - G_3 \cdot c_3 = G_2 \cdot c_2 \end{cases}$$

Матрица A коэффициентов при неизвестных членах:

$$\begin{vmatrix} 1 & -1 \\ c_1 & -c_3 \end{vmatrix}$$

Вектор B известных членов уравнений:

$$\begin{vmatrix} G_2 \\ G_2 \cdot c_2 \end{vmatrix}$$

Решение системы (значения G_1 и G_3) рассчитываются по матричному уравнению:

$$G = A^{-1} \cdot B$$

4 Расчет и оформление результатов

После расчета по формулам необходимо оформить таблицу материального баланса, в левой части которой будут показаны значения поступающих на узел потоков и а. с. целлюлозы в них, в правой части – то же для уходящих потоков.

На листе системы Excel весь расчет будет выглядеть, как показано на рисунке 8.2.

	A	B	C	D	E	F	G	H	I
1	Таблица 8.1 – Исходные данные узла сгущения								
2	Наименование показателя		Обозначение	Единица измерения	Величина	Обоснование			
3	Поток уходящей с узла целлюлозной массы		G_2	т/ч	50,04	известно из расчета другого узла			
4	Концентрация целлюлозы:								
5	- в поступающем потоке		c_1	%	2,50	принимаем			
6	- после сгущения		c_2	%	12,00	задано при расчете другого узла			
7	- в отобранной воде		c_3	%	0,05	принимаем			
8									
9									
10									
11									
12									
13									
14									
15									
16	Система уравнений материального баланса								
17	$\begin{cases} G_1 = G_2 + G_3 \\ G_1 \cdot c_1 = G_2 \cdot c_2 + G_3 \cdot c_3 \end{cases}$								
18									
19	Система для решения матричным методом:								
20	$\begin{cases} G_1 - G_3 = G_2 \\ G_1 \cdot c_1 - G_3 \cdot c_3 = G_2 \cdot c_2 \end{cases}$								
21									
22									
23	Матрица А		Матрица В		Решение системы				
24	1	-1	50,04		$G_1 = 244,07$				
25	2,5	-0,05	600,48		$G_3 = 194,03$				
26	Матрица A^{-1}								
27	-0,020	0,408							
28	-1,020	0,408							
29	Таблица 8.2 – Материальный баланс узла сгущения								
30	Приход			Расход					
31	Наименование потока	Всего	Целлюлоза	Наименование потока	Всего	Целлюлоза			
32	Поступает	244,07	6,102	- после сгущения	50,04	6,005			
33				- в отобранной воде	194,03	0,097			
34	Итого	244,07	6,102	Итого	244,07	6,102			
35									
36	$=C32 * F5 / 100$								

Рисунок 8.2 – Расчет материального баланса узла сгущения целлюлозы

Учебное задание

Выполнить расчет материального баланса узла сгущения для следующих исходных данных:

Наименование показателя	Обозначение	Единица измерения	Величина	Обоснование
Поступающий поток целлюлозной массы	G_1	т/ч	244,07	известно из расчета предыдущего узла
Концентрация целлюлозы:				
– в поступающем потоке	C_1	%	2,50	известно из расчета предыдущего узла
– после сгущения	C_2	%	12,00	принимаем
– в отобранной воде	C_3	%	0,05	принимаем

Лабораторная работа № 9. Microsoft Access 2010. Проектирование и создание базы данных

Цель работы

Освоение приемов проектирования базы данных, описания структуры таблиц и связей между ними.

Учебные вопросы

1. Проектирование и создание базы данных
2. Описание структуры таблиц и связей

Краткое изложение учебного материала по теме

Вопросы проектирования базы данных подробно описаны в учебнике Информатика [1, с. 236 – 247], однако приведем здесь некоторые основные положения.

База данных (БД, database) поименованная совокупность структурированных данных, относящихся к определенной предметной области.

Система Access – реляционного типа, т. е. ее база данных состоит из совокупности связанных между собой таблиц. Каждая таблица имеет строгую структуру.

Таблица базы данных (table) – регулярная структура, состоящая из однотипных строк, которые называются **записями (records)**, разбитых на **поля (fields)**. Каждое **поле записи** обязательно имеет **имя, тип и формат (или ширину)**.

Для связей между таблицами используются ключи (физическая реализация ключей – индексы).

Первичный ключ (primary key) – главный ключевой элемент, однозначно идентифицирующий запись в таблице.

В системе Access под термином Ключевое поле подразумевается первичный ключ, для других ключей (уникальных или внешних) используется атрибут Индексированное поле (Совпадения не допускаются) или Индексированное поле (Совпадения допускаются).

Главный принцип проектирования – совокупность связанных таблиц создается таким образом, чтобы суммарный объем хранимой информации был минимален, и любую информацию можно было быстро найти.

Обычно в состав базы данных входят таблицы для хранения главной информации, которые могут постоянно пополняться данными, и справоч-

ные таблицы, редко изменяющиеся.

Связи между таблицами в системе Access задаются с использованием режима *Схема данных*. Для связей следует задать условия соблюдения ссылочной целостности.

Ссылочная целостность данных (referential integrity) – набор правил, обеспечивающих соответствие ключевых значений в связанных таблицах.

В состав информационной системы кроме информации базы данных входят также компоненты пользовательского интерфейса, важнейшие из которых – формы и печатные отчеты. Особенность системы Access – эти компоненты могут храниться в одном файле с таблицами или в разных файлах

Все имена в БД (таблиц, полей, форм, отчетов, запросов и пр.) конечно же лучше писать с использованием латинских букв и английских слов (если система разрабатывается для международных корпораций), но в учебном примере для простоты будем пользоваться русскими названиями.

1 Проектирование и создание базы данных

Процесс создания базы данных рассмотрим на примере разработки информационной системы «Студенты», которая должна хранить информацию о студентах и их экзаменационных оценках.

В учебнике Информатика [1, с. 245 – 248] подробно описан процесс проектирования информационной системы «Контингент студентов университета», начиная с раздела постановки задачи и далее описание концептуальной модели с подробным разбором всех сущностей, их основных атрибутов и связей.

В результате проектирования был сделан вывод о необходимости создания в ней 5-ти таблиц:

- 1) **Студенты** – для хранения основных данных о студенте;
- 2) **Оценки** – для хранения информации об оценках студентов;
- 3) **Институты** – справочник институтов;
- 4) **Специальности** – справочник специальностей;
- 5) **Предметы** – справочник предметов.

Для создания файла базы данных в папке хранения Ваших файлов вызовем контекстное меню и в нем выберем команду Создать → Microsoft Access База данных. Зададим имя базы данных **Студенты** (автор <Ваша фамилия>). Откроем базу данных двойным щелчком на созданном файле.

2 Описание структуры таблиц и связей

Выберем на ленте вкладку Создание и в группе Таблицы нажмем на кнопку Конструктор таблиц. По умолчанию для окна базы данных уста-

новлен параметр Вкладки, поэтому внутри главного окна мы увидим вкладку (вложенное окно с ярлычком сверху) Конструктора таблиц, показанное на рисунке 9.1 (данные двух полей уже заполнены и была нажата кнопка Сохранить на верхней рамке окна).

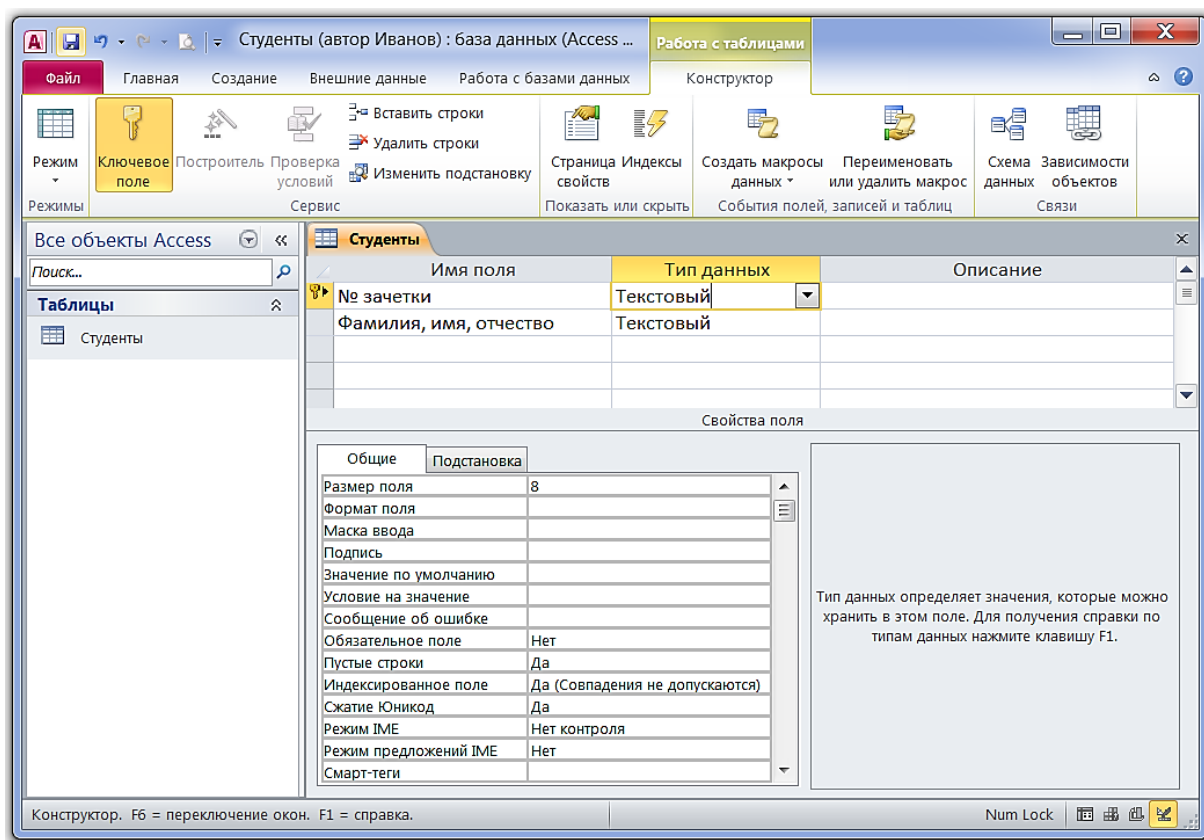


Рисунок 9.1 – Описание структуры таблицы **Студенты** в Конструкторе

Далее в Конструкторе добавим остальные поля в соответствии с данными таблицы 9.1, т.е. зададим имя, тип данных, размер или формат каждого поля таблицы, а также ключевое поле (если необходимо), индексированные поля и подписи. После чего закроем вкладку Конструктора таблицы **Студенты** (крестиком справа на темно-серой полоске или из контекстного меню ярлычка) с сохранением изменений структуры.

Затем снова выберем команду **Создание** → **Конструктор таблиц** и опишем структуру следующей таблицы – **Оценки** в соответствии с данными таблицы 9.2. Сохраним таблицу и закроем Конструктор данной таблицы.

Аналогично поступим при создании еще трех таблиц – **Институты** (структура приведена в таблице 9.3), **Специальности** (структура приведена в таблице 9.4) и **Предметы** (структура приведена в таблице 9.5).

В результате получим в базе данных 5 пустых таблиц с заданной структурой. При необходимости в любой момент можно обратиться к модификации структуры каждой из таблиц, открыв ее в Конструкторе.

Таблица 9.1 – Структура таблицы **Студенты**

Имя поля	Тип данных	Размер поля	Индексированное поле
№ зачетки	Текстовый	8	Ключевое поле
Фамилия, имя, отчество	Текстовый	45	Нет
Дата поступления	Дата/время	Краткий формат даты	Нет
№ института	Числовой	Байт	Да (Допускаются совпадения)
Код специальности	Текстовый	9	Да (Допускаются совпадения)
Курс	Числовой	Байт	Нет
Группа	Текстовый	4	Нет

Таблица 9.2 – Структура таблицы **Оценки**

Имя поля	Тип данных	Размер поля	Индексированное поле	Обязательное поле
№ зачетки	Текстовый	8	Да (Допускаются совпадения)	Да
Семестр	Числовой	Байт	Нет	Да
№ предмета	Числовой	Целое	Да (Допускаются совпадения)	Да
Оценка	Текстовый	1	Нет	Да
Дата получения	Дата/время	Краткий формат даты	Нет	Да
Преподаватель	Текстовый	45	Нет	Да

Таблица 9.3 – Структура таблицы **Институты**

Имя поля	Тип данных	Размер поля	Индексированное поле
№ института	Числовой	Байт	Ключевое поле
Название института	Текстовый	120	Нет

Таблица 9.1 – Структура таблицы **Специальности**

Имя поля	Тип данных	Размер поля	Индексированное поле
Код специальности	Текстовый	9	Ключевое поле
Название специальности	Текстовый	120	Нет

Таблица 9.5 – Структура таблицы **Предметы**

Имя поля	Тип данных	Размер поля	Индексированное поле
№ предмета	Числовой	Целое	Ключевое поле
Название предмета	Текстовый	120	Нет

Далее задаем связи (Один ко многим) между таблицами в базе. Для этого на вкладке ленты Работа с базами данных выбираем в группе Отношения команду Схема данных, добавляем в окно схемы все таблицы и, перетаскивая название поля первичного ключа к аналогичному полю другой таблицы создать связи. При этом задаем в окне Изменение связей для всех связей между таблицами 3 условия: обеспечения целостности данных, каскадное обновление связанных полей и каскадное удаление связанных записей. Схема базы данных показана на рисунок 3.57.

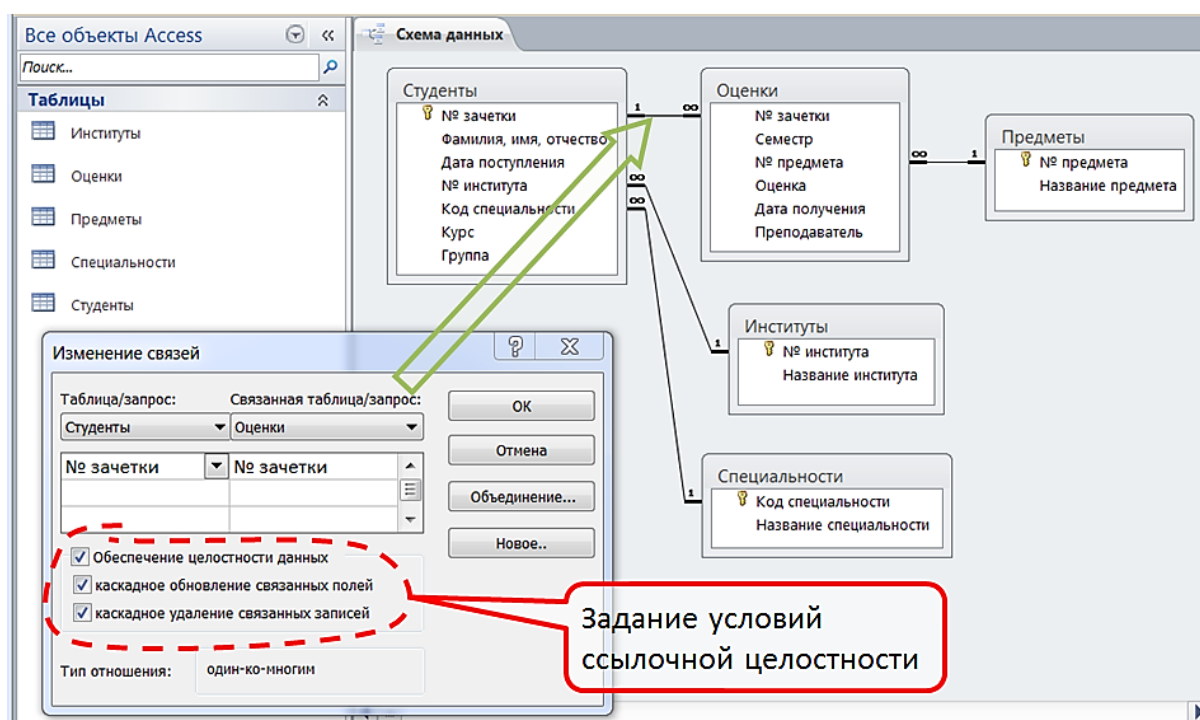


Рисунок 9.2 – Схема базы данных и задание условий ссылочной целостности для связи между таблицами Студенты – Оценки

Учебное задание

Создать все необходимые таблицы базы данных, описать связи между ними в соответствии с учебным примером.

Лабораторная работа № 10. Microsoft Access 2010. Занесение информации в базу данных. Разработка форм

Цель работы

Освоение приемов работы с информацией базы данных в табличном режиме, разработка и использование форм.

Учебные вопросы

1. Занесение информации в таблицы-справочники
2. Разработка Windows-формы для работы с информацией базы данных
3. Занесение информации с использованием Windows-формы

Краткое изложение учебного материала по теме

В информационных системах масштаба предприятия основным режимом работы с базой данных этой системы являются Windows-формы или Web-формы, которых может быть очень много и которые могут быть предназначены для тех или иных рабочих мест и различных решаемых задач. Простой пользователь в этом случае не знает, с какими таблицами базы данных он работает с использованием формы.

В системе Access также могут использоваться такие формы, но кроме того может использоваться режим непосредственной работы с таблицами, которых, как правило, немного и информация о которых наглядно представлена в этой системе.

В то же время Windows-формы и Web-формы предоставляют дополнительные возможности по одновременной работе с несколькими таблицами базы данных, по использованию разнообразных элементов управления на форме (поле со списком, кнопки, вложенные формы и пр.).

В системе **Access** имеется **Мастер** разработки экранных форм, который позволяет легко создавать формы нескольких видов (простые для работы с данными одной таблицы, сложные для работы с несколькими таблицами с использованием подчиненных или связанных форм). Полученные формы далее, как правило, приходится дополнять и модифицировать в **Конструкторе** форм для реализации всех необходимых условий работы с данными.

1 Занесение информации в таблицы-справочники

Для работы со справочниками можно использовать стандартный режим работы с таблицами. Перед тем, как заносить данные в таблицы, в настройках системы (вкладка Файл → Параметры → Текущая база данных) уберем галочку у параметра Включение изменений структуры таблиц

в табличном представлении (после этого изменение структуры будет возможно только в Конструкторе таблиц). После этого нужно закрыть и снова открыть файл базы данных.

Таблица **Предметы** в режиме работы с данными (двойной щелчок мышкой на ее названии в Области переходов или команда **Открыть** контекстного меню) показан на рисунок 10.1.

№ предмета	Название предмета
+	1 Иностранный язык
+	2 История России
+	3 Правоведение
+	4 Математика
+	5 Информатика
+	6 Общая и неорганическая химия
+	7 Проблемы современного естествознания
*	

Рисунок 10.1 – Работа с таблицей Предметы базы данных

В данном режиме каждая запись таблицы базы данных представлена как строка, состоящая из столбцов – полей, над которыми показаны имена полей. В нижней части таблицы присутствует пустая запись с символом **звездочка *** в левой колонке – это несуществующая запись, которая добавляется в таблицу, как только в ней появляется какая-либо информация. На нижней рамке окна присутствуют кнопки для перемещения по таблице, номер текущей записи и информация об общем количестве записей в таблице, а также поле быстрого поиска и сведения об установленном фильтре отбора данных. Колонка слева с символом **+** присутствует, если у таблицы есть связь от одной записи данной таблицы ко многим записям связанной с ней таблицы. В данном случае при щелчке мышью на плюсе откроется подтаблица оценок для выбранного предмета.

После окончания добавления, редактирования или удаления данных можно закрыть окно таблицы, при этом все изменения будут сохранены автоматически.

Для занесения информации в справочную таблицу Специальности можно использовать импорт данных. Для этого откроем сайт университета и из раздела АБИТУРИЕНТУ – Специальности – Очно – Бакалавриат скопируем таблицу всех специальностей в новый файл Excel, сохраним его в своей папке с именем Специальности с сайта. В базе данных выполним модификацию структуры таблицы Специальности – добавим в нее поле Институт (текстовое, ширина 24). Удалим из таблицы Excel лишние столбцы, оставим только код специальности, ее название и институт, где есть эта

специальность. Добавим в файле новую первую строку, в которой подпишем названия столбцов в точном соответствии с названием полей в таблице базы данных Специальности. Зададим заполненных для столбцов в таблице Excel тип текстовый. Если в кодах специальностей вместо точки стоит запятая, выполним автозамену всех запятых на точки. После этого таблица Специальности с сайта должна выглядеть, как показано на рисунке 10.2.

	1	2	3
1	Название специальности	Код специальности	Институт
2	Автоматизация технологических	220700.62	ИЭТ
3	процессов и производств		
4	Биология	20400.62	
5	Биотехнические системы и технологии	201000.62	ИТПХ
6	Биотехнология	240700.62	ИТПХ

Рисунок 10.2 – Фрагмент таблицы Excel для экспорта данных

В базе данных закроем все таблицы, на вкладке Внешние данные нажмем кнопку **Excel** (импорт электронной таблицы Excel), выберем имя файла **Специальности с сайта**, выберем позицию **Добавить копию записей в конец таблицы: Специальности** и выполним импорт данных. Некоторые данные будут потеряны, т. к. есть одинаковые специальности в таблице в разных институтах. В результате получим заполненную таблицу Специальности, как показано на рисунке 10.3.

Код специальности	Название специальности	Институт
220700.62	Автоматизация технологических процессов и производств	ИЭТ
221000.62	Мехатроника и робототехника	ИЭТ
221400.62	Управление качеством	ИТПХ
221700.62	Стандартизация и метрология	ИТПХ
230400.62	Информационные системы и технологии	ИИКТ

Рисунок 10.3 – Фрагмент таблицы Специальности после импорта данных

Аналогично можно выполнить импорт данных в таблицу базы данных Института с сайта университета.

2 Разработка Windows-формы для работы с информацией базы данных

Для данной информационной системы разработаем форму, которая позволит одновременно редактировать информацию студентов и заносить

данные об их оценках. Главная таблица формы – Студенты, на форме будут представлены данные одной записи. Одновременно в подчиненной таблице на той же форме мы можем видеть все его оценки.

Разработку формы можно начинать в Конструкторе форм, начиная с пустой формы и размещая на ней необходимые поля таблиц и другие элементы управления.

Но более быстрый способ – использовать **Мастер форм** (рисунок 10.3).

Как показано на рисунке 10.4, на первом шаге Мастер (в окне Создание форм) просит выбрать **поля таблиц**, которые будут использоваться, как элемент управления **поля формы**. Выберем все поля таблицы Студенты и все поля таблицы Оценки (двойным символом >>), но затем удалим из списка выбранных поле Оценки.[№ зачетки] (для подчиненной таблицы поле [№ зачетки] будет заполняться автоматически; можно оставить его в списке, если хотите убедиться в автоматическом заполнении и в том, что на форме мы видим оценки только одного студента). Нажмем кнопку **Далее >** для перехода к следующему шагу Мастера форм.

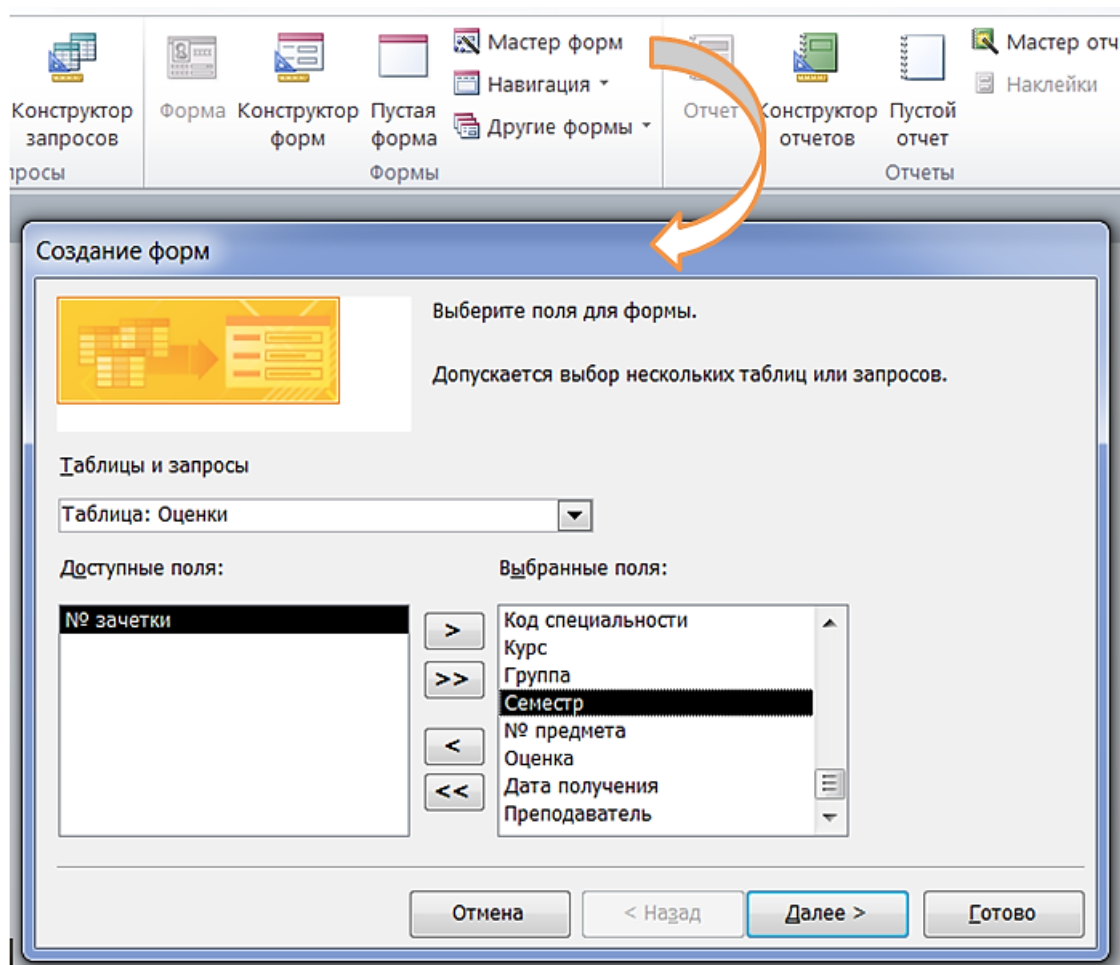


Рисунок 10.4 – Первый шаг Мастера форм

На втором шаге – *Выберите вид представления данных* – вы-

бираем **Подчиненные формы** – т. е. расположение данных главной таблицы Студенты и подчиненной формы с данными таблицы Оценки на одной форме (рисунок 10.5).

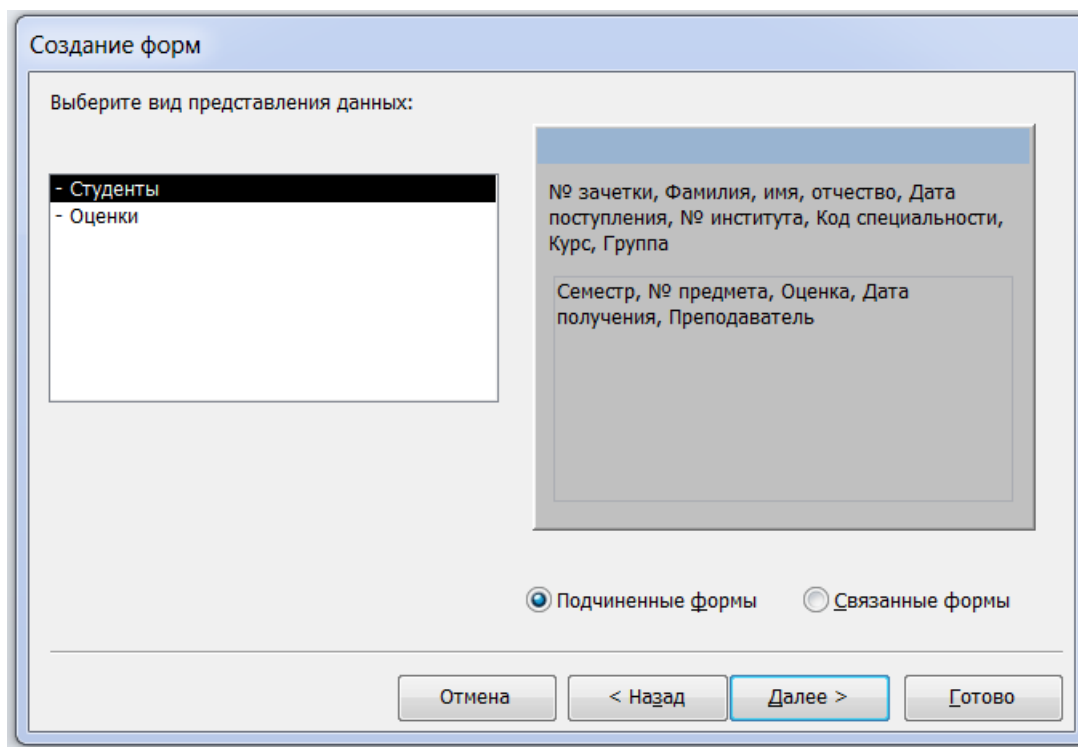


Рисунок 10.5 – Второй шаг Мастера форм

Третий шаг – *Выберите внешний вид подчиненной формы* – выбираем **ленточный**.

Последний шаг – *Задайте имена форм* – оставляем для главной формы название **Студенты**, для подчиненной формы задаем имя **Оценки студента** и на том же экране ниже выбираем пункт **Изменить макет формы**, после чего нажимаем на кнопку **Готово**.

Полученная форма будет открыта в Конструкторе форм, в Области навигации базы данных (левая панель окна Access, настройка области – тип объекта, все объекты) появится группа **Формы** с двумя именами новых форм – **Оценки студента** и **Студенты**. Для отображения в Конструкторе подчиненной формы следует закрыть окно формы и снова его открыть в Конструкторе, в этом случае она будет иметь вид, приведенный на рисунок 10.6.

На форме присутствуют элементы управления (объекты) следующих типов:

- **Надпись** – текст на форме, обычно не изменяющийся.
- **Поле** – объект для редактирования данных, связанный с полем таблицы базы данных или с переменной. Главное свойство этого объекта – **Данные** на странице **Данные** Окна свойств (открывается кнопкой **Страница**

свойств вкладки Конструктор), которое показывает, с какими данными связан этот объект формы.

- **Подчиненная форма** – вложенная форма для дочерней таблицы данных, связанной с главной таблицей, на которой могут присутствовать такие же элементы, как и на основной форме.

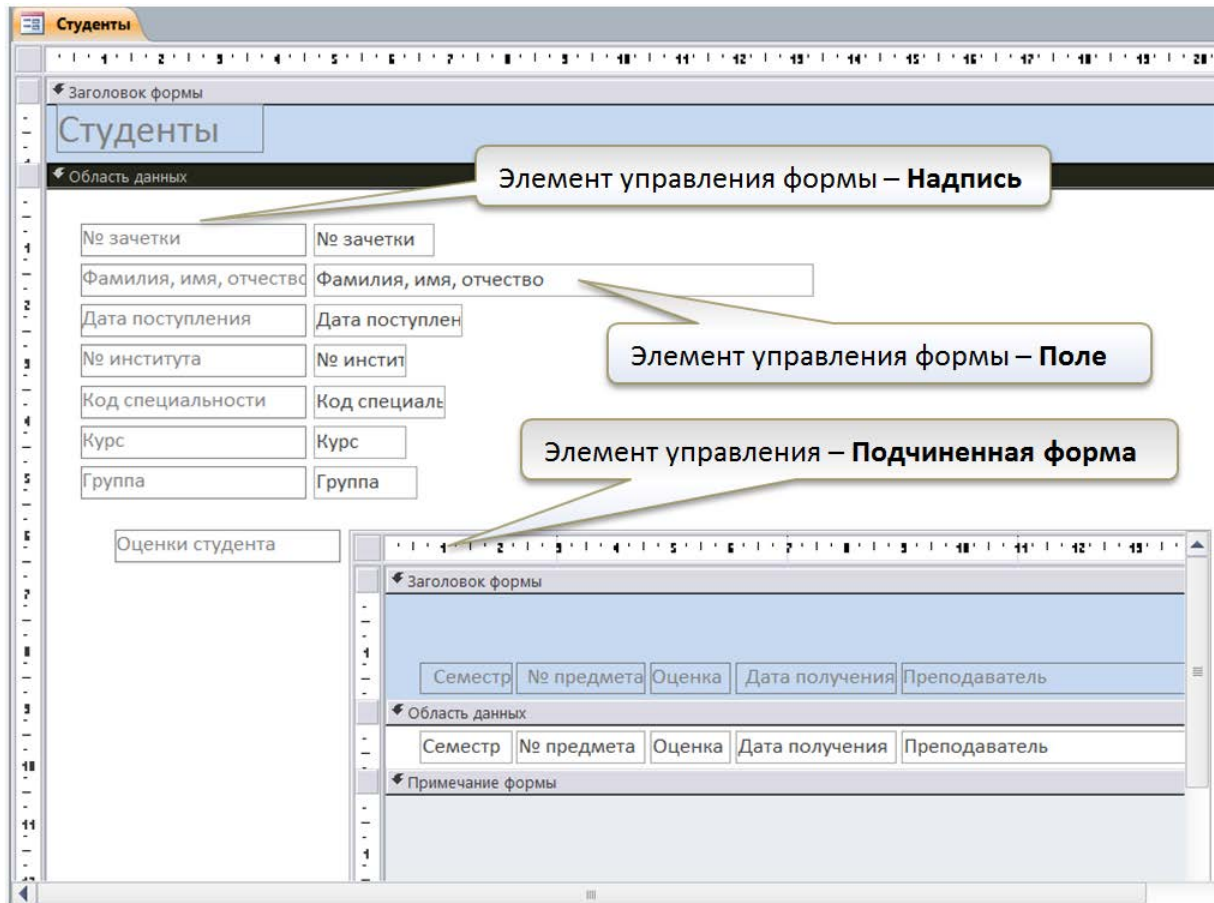


Рисунок 10.6 – Вид формы, созданной Мастером, в Конструкторе

Кроме того, на форме могут присутствовать и другие объекты, которые можно добавлять, используя группу кнопок Элементы управления контекстной вкладки Конструктор (рисунок 10.7).

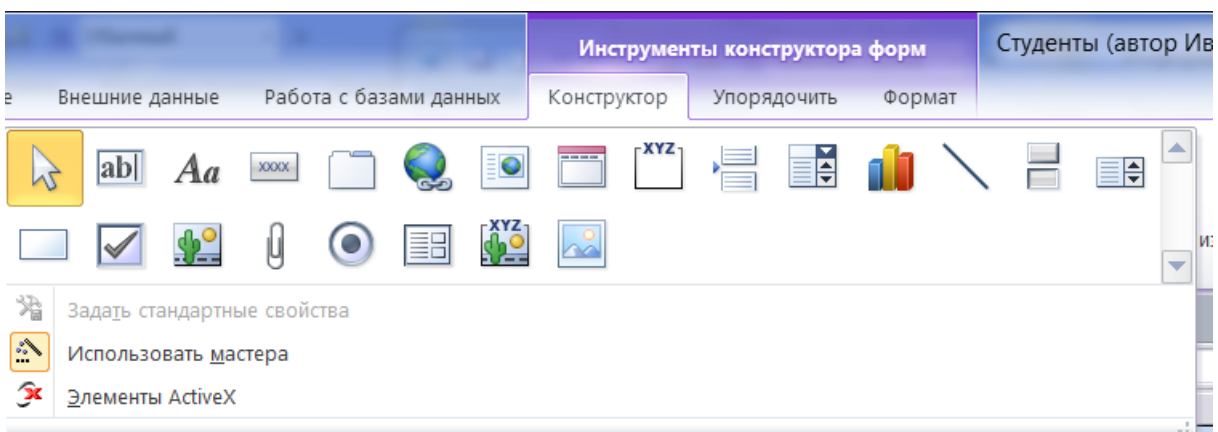


Рисунок 10.7 – Элементы управления для формы

Для использования формы можно запустить ее в работу с данными двойным щелчком мышкой на имени формы Студенты в Области навигации. Но пока не будем этого делать, т. к. форма нуждается в значительном усовершенствовании.

Прежде всего, можно изменить размер и расположение полей, чтобы все надписи были хорошо видны, и форма была более компактной. Для надписей можно использовать команду контекстного меню Размер → По размеру данных. Поля и их надписи в системе Access взаимосвязаны и перемещаются совместно. Для перемещения поля независимо от надписи нужно использовать маркер в левом верхнем углу рамки объекта. Можно изменить также цвет надписей и фона формы.

Далее добавляем на форму 3 элемента управления типа Поле со списком для выбора из справочных таблиц института, специальности (с занесением соответствующих номеров в главную таблицу) и предмета (с занесением его номера из справочника в таблицу оценок). Основные свойства этого объекта описываются с помощью Мастера, который запускается автоматически при добавлении этого типа объекта к форме, если в группе кнопок Элементы управления включена кнопка Использовать мастера (см. ранее на рисунке 10.5). Для добавления нового поля со списком необходимо выбрать его тип в группе Элементы управления вкладки Конструктор, затем показать его расположение и размер на форме, после чего запустится Мастер создания полей со списком.

На первом шаге Мастера выбираем пункт *Объект "поле со списком" будет использовать значения из таблицы или другого запроса* (рисунок 10.8).

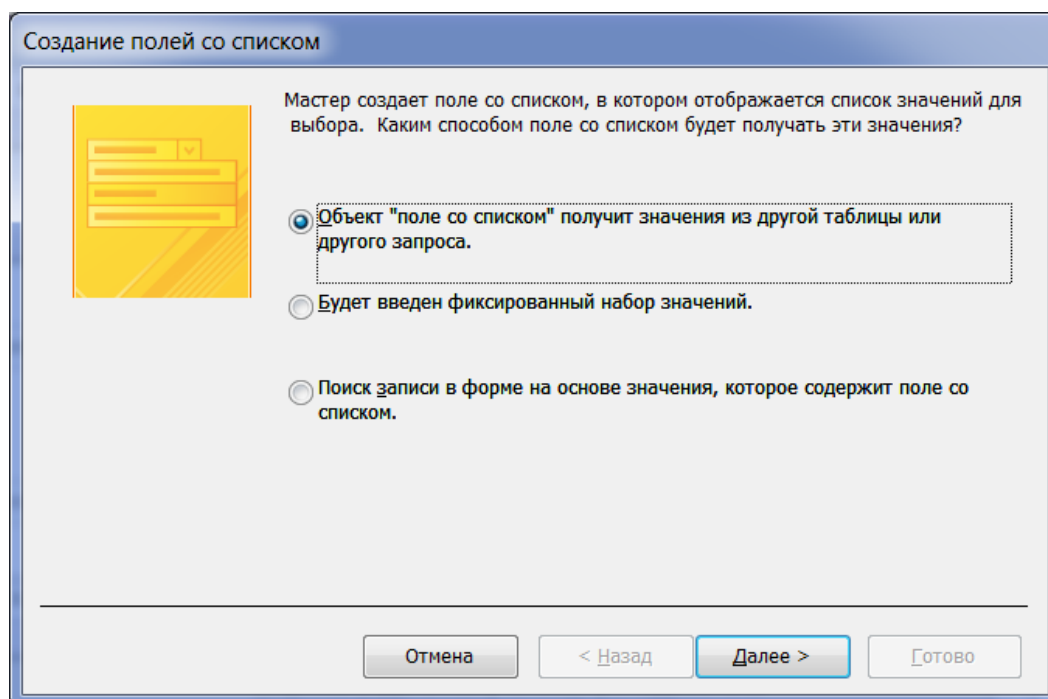


Рисунок 10.8 – Первый шаг Мастера создания полей со списком

На втором шаге выбираем из списка нужную нам справочную таблицу, на третьем – выбираем все поля (номер и название), на четвертом шаге – оставляем галочку у флажка *Скрыть ключевой столбец* и задаем ширину поля для названия (рисунок 10.9).

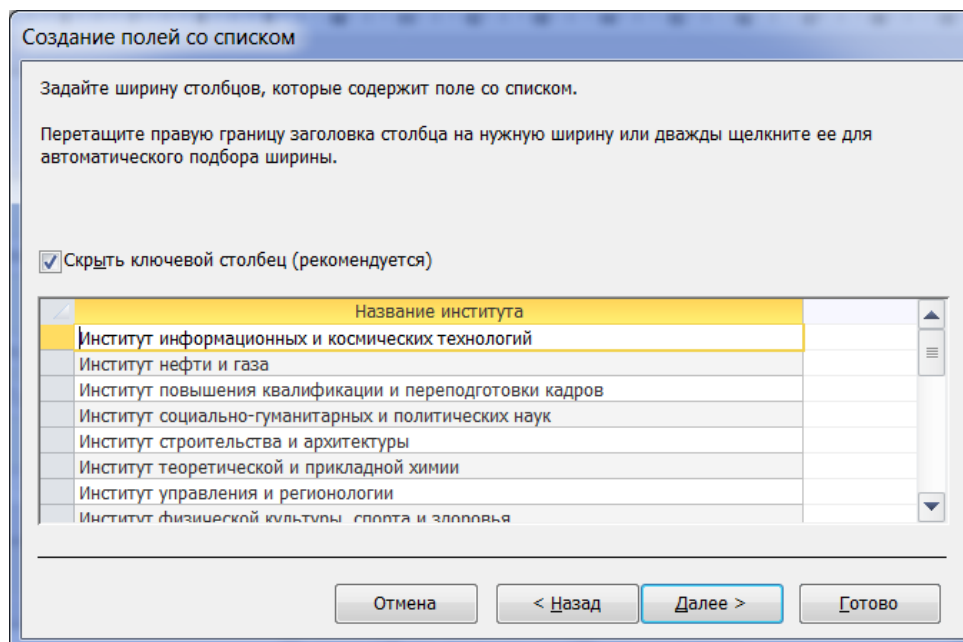


Рисунок 10.9 – Четвертый шаг Мастера создания полей со списком

Пятый шаг – задаем условие *Сохранить в поле* и выбираем из списка поле главной таблицы, в котором будет сохраняться значение ключевого поля справочной таблицы (рисунок 10.10).

Если Вы не укажете поле таблицы, с которым связано поле со списком, все Ваши предыдущие шаги бессмысленны для данной задачи!!!

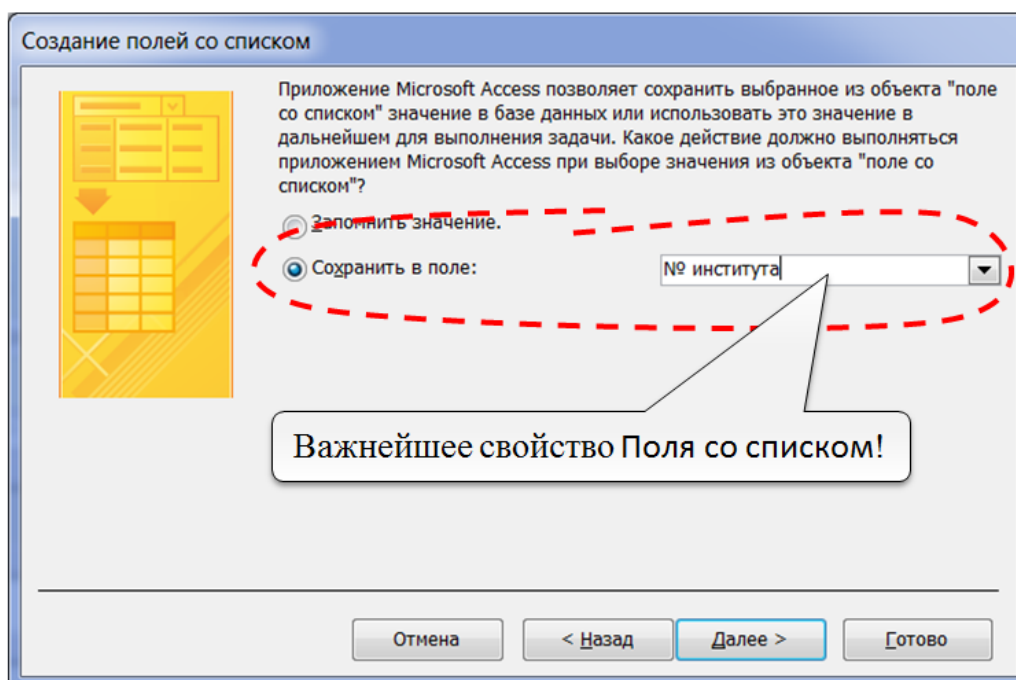


Рисунок 10.10 – Последний шаг Мастера создания полей со списком

Далее нажимаем кнопку Готово. Надпись для Поля со списком на форме можно удалить.

После модификации форма будет иметь в Конструкторе вид, приведенный на рисунок 10.11.

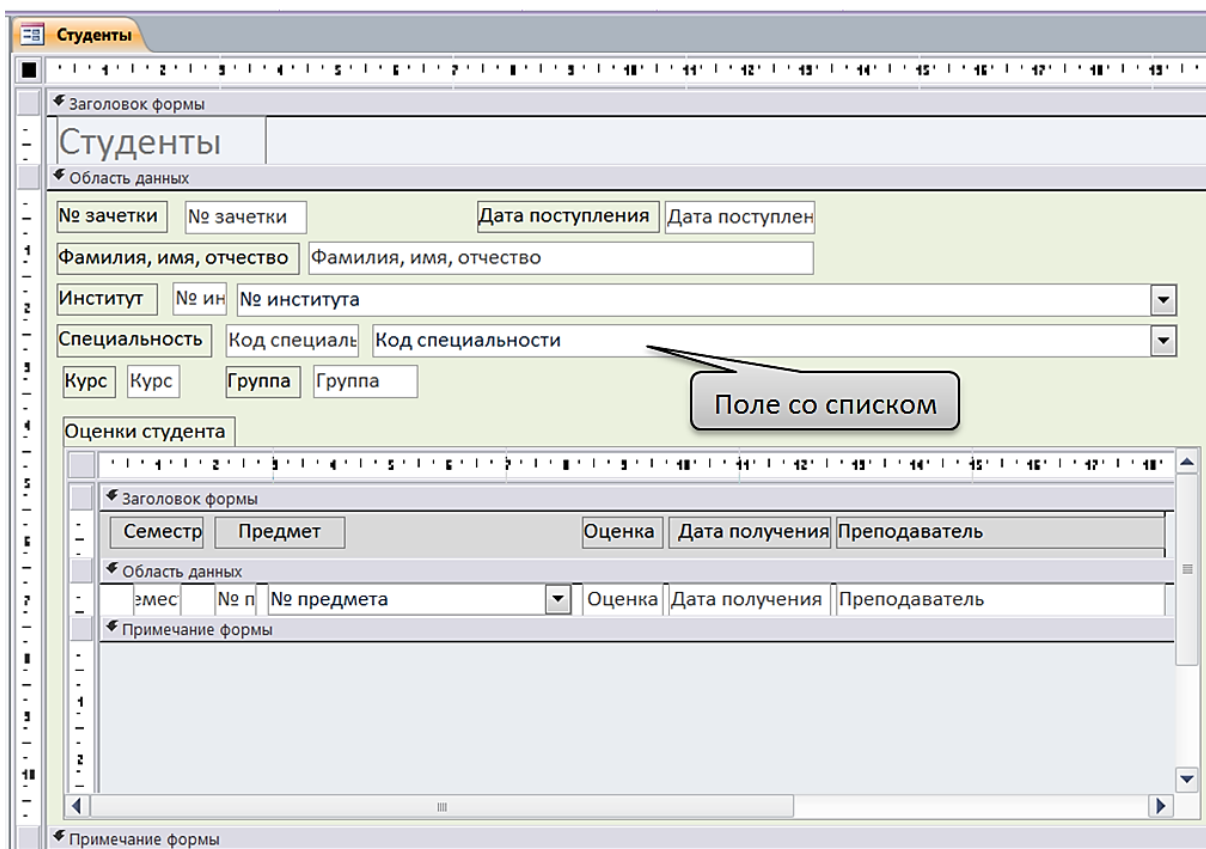


Рисунок 10.11 – Усовершенствованная форма в Конструкторе

3 Занесение информации с использованием Windows-формы

Запустим форму Студенты в работу с базой данных двойным щелчком на ее названии в Области навигации. В этом режиме можно редактировать существующие записи, добавлять новых студентов и новые оценки для каждого студента. Для удаления записей можно использовать контекстные меню для вертикальной полосы слева в главной форме и для меток у строк слева в подчиненной форме или клавишу Delete после выделения этих меток.

В усовершенствованной форме можно видеть названия факультетов, специальностей, предметов и выбирать их из справочных таблиц с помощью раскрывающихся списков (рисунок 10.12).

Перемещаться по полям ввода данных можно с помощью мышки или нажатием на клавиатуре клавиш Enter или Tab.

Для перехода от одного студента к другому или к новой записи используются навигационные кнопки (Первая запись, Предыдущая запись,

Следующая запись, Последняя запись, Новая запись) на нижней рамке формы.

Выбор института, специальности и предмета можно выполнять с использованием раскрывающихся списков или заданием номеров и кода, в обоих случаях мы будем видеть название в поле со списком.

Во время работы с формой можно задать сортировку данных по любому из полей формы и фильтр отбора данных с использованием кнопок группы Сортировка и фильтр вкладки Главная ленты.

Семестр	Предмет	Оценка	Дата получения	Преподаватель
1	Иностранный язык	5	11.07.2006	Орлова О.П.
1	Информатика	4	21.07.2006	Коновалова Э.И.
1	История России	5	15.07.2006	Веретнов М.Ю.
1	Математика	4	19.07.2006	Шапаров Е.А.
1	Правоведение	4	17.07.2006	Рай А.И.

Рисунок 10.12 – Использование формы с объектами Поле со списком для работы с базой данных

Учебное задание

1. Заполнить в табличном режиме справочную таблицу базы данных **Предметы**.
2. С использованием импорта данных с сайта университета заполнить информацией справочные таблицы базы данных **Институты** и **Специальности**.
3. Создать Windows-форму для занесения данных по студентам и их оценкам.
4. С использованием разработанной формы занести информацию об успеваемости студентов в базу данных. Минимальный объем информации: студенты двух институтов, в каждом 2 группы, в каждой группе 3 студента, у каждого студента 3 оценки.

Лабораторная работа № 11. Microsoft Access 2010. Разработка отчетов и запросов

Цель работы

Изучение возможностей и освоение приемов разработки отчетов и запросов.

Учебные вопросы

1. Разработка отчета
2. Разработка запросов

Краткое изложение учебного материала по теме

1 Разработка отчетов

Для разработки отчетов, отражающих информацию базы данных в печатных документах, в системе Access можно использовать Мастер отчетов и Конструктор отчетов. Отчеты могут быть созданы на основе всей информации, присутствующей в таблицах базы, но чаще для отчетов необходимо отобразить нужную информацию из базы с использованием SQL запроса и на основе его создать отчет. Важным свойством отчетов является возможность группировки данных и получения итоговых данных для групп и всего отчета.

Печатные документы отчетности предприятий часто должны иметь точное соответствие с унифицированными формами, утвержденными Госкомстатом РФ, как правило, Конструктор отчетов позволяет это сделать.

Поставим задачу разработать отчет, в котором показаны оценки всех студентов с группировкой данных по факультетам, курсам, группам. Для этого на вкладке Создание в группе Отчеты нажимаем кнопку Мастер отчетов.

На первом шаге Мастера выбираем поля главной таблицы базы Студенты, которые мы хотим показать в отчете и все поля дочерней таблицы Оценки, кроме поля [№ зачетки] (рисунок 11.1).

На втором шаге – *Выберите вид представления данных* – выбираем первый вариант, когда выделена таблица Студенты. При этом показано, что в верхней части отчета присутствуют поля таблицы Студенты, в нижней – поля таблицы Оценки.

На третьем шаге добавляем группировку данных по полям № института, Курс и Группа (рисунок 11.2). Более трех уровней группировки мастер задать не позволяет, однако это можно сделать при необходимости в **Конструкторе отчетов**.

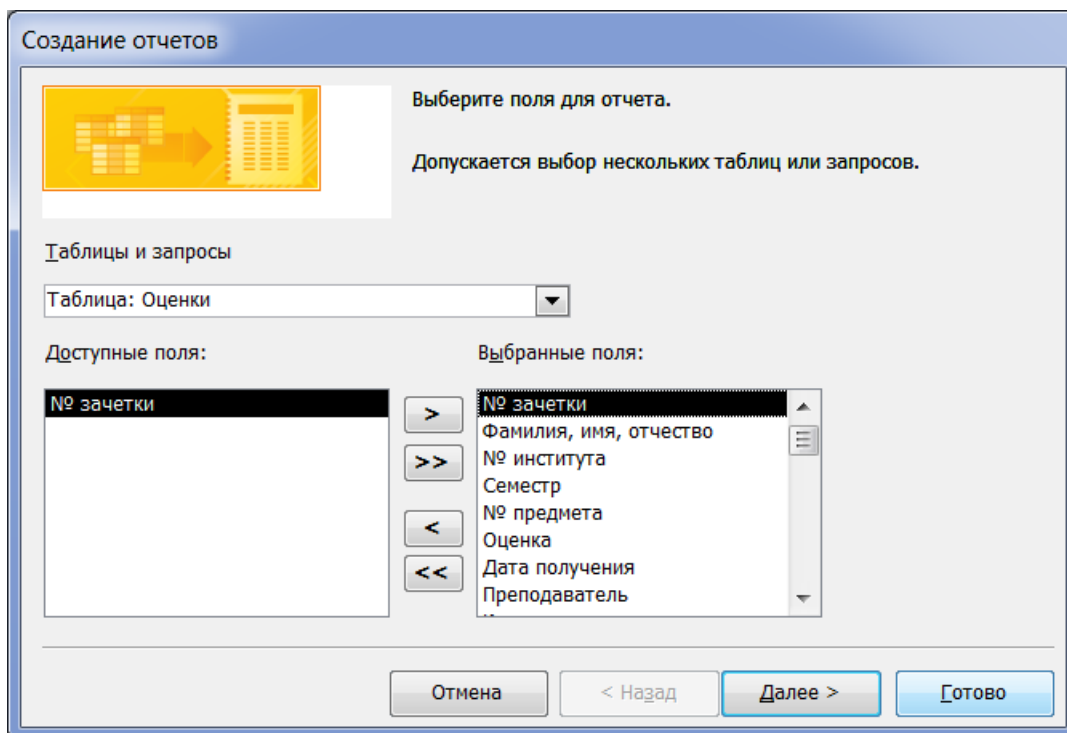


Рисунок 11.1 – Выбор полей для отчета на первом шаге Мастера отчетов

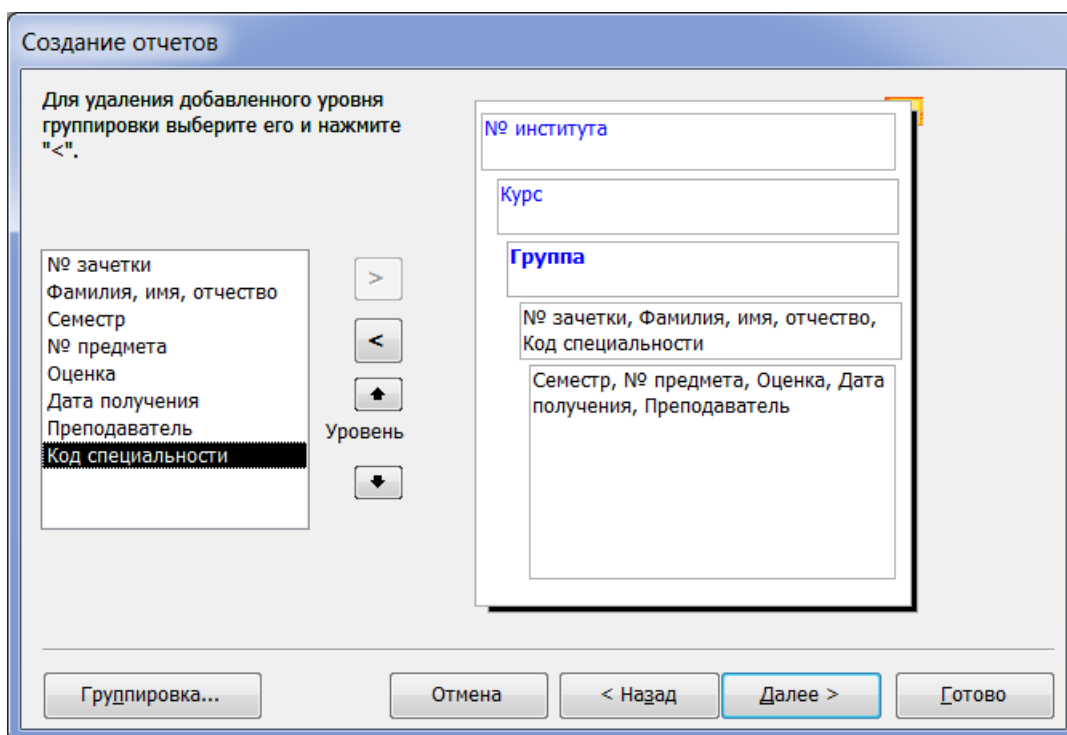


Рисунок 11.2 – Добавление уровней группировки для отчета

Сортировку на следующем шаге не задаем, т. к. она будет задана автоматически для выбранных группировок данных.

На 5-м шаге выбираем вид макета для отчета – *структура*, на следующем задаем название отчета **Студенты и их оценки** и нажимаем кнопку **Готово**. Полученный отчет в Конструкторе отчетов показан на рисунке 11.3.

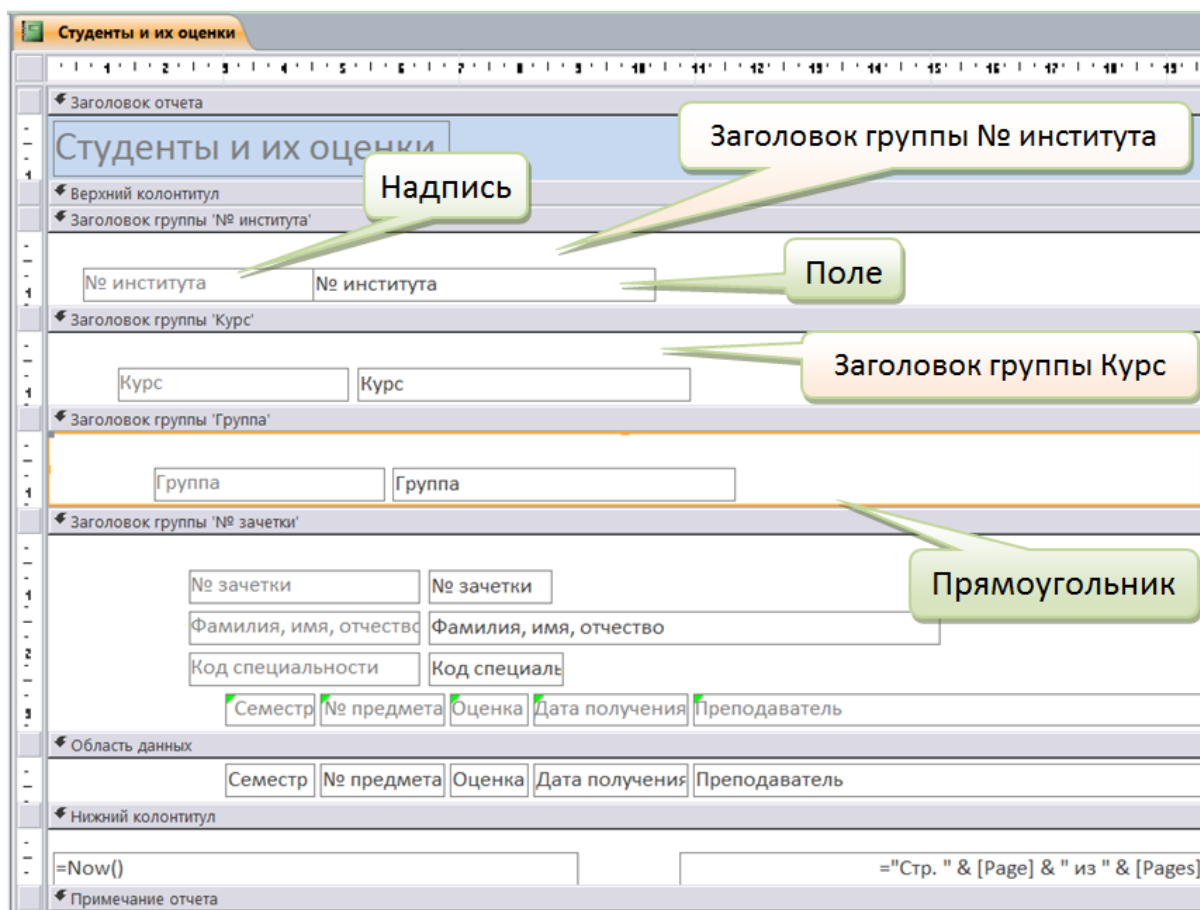


Рисунок 11.3 – Вид отчета в Конструкторе, созданного Мастером отчетов

В полученном отчете присутствуют объекты 3-х видов – **Поле**, которое в отчете может показывать данные поля таблицы базы, значение переменной или выражения, **Надпись** – любой текст в отчете и **Прямоугольник** – элемент оформления.

В **Конструкторе** отчет разбит на отдельные зоны, информация которых может присутствовать в отчете один раз (Заголовок отчета и Примечание отчета), в начале каждой страницы (Верхний колонтитул) или в конце каждой страницы (Нижний колонтитул), в начале каждой группы (Заголовок группы, групп может быть много) и в конце каждой группы (Примечание группы) и для каждой записи главной таблицы отчета (Область данных). Зоны Примечания... (их Мастер не создал) более правильно было бы назвать Итоги..., т. к. здесь можно разместить поля общих итогов (сумма, среднее и пр.) для группы или всего отчета.

Полученный отчет можно просмотреть на экране, отправить на принтер (например, с использованием соответствующих кнопок на стандартной панели инструментов) или в файл rtf -формата.

Созданный **Мастером** отчет весьма несовершенен, поля его страницы слишком малы, размещение полей в отчете нерационально. В отчет необходимо добавить названия факультетов, специальностей и предметов и изменить их надписи. А самое главное – необходимо задать наличие зон

Примечаний и в них разместить поле с расчетом средних оценок для всех 4-х групп.

Вначале настроим поля страницы отчета с использованием кнопки Параметры страницы вкладки Параметры страницы (верхнее 20 мм, нижнее 20 мм, левое 25 мм, правое 10 мм). После этого ширину отчета необходимо уменьшить до 17,4 см (при этом поля оценок нужно переместить влево и уменьшить ширину рамки в зоне Заголовков группы "Группа").

Для добавления новых и модификации существующих элементов управления следует использовать вкладку Конструктор контекстной группы вкладок Инструменты конструктора отчетов, для настройки расположения объектов – вкладку Упорядочить, для настройки параметров страницы (полей и размера бумаги) – вкладку Параметры страницы (рисунок 11.4).

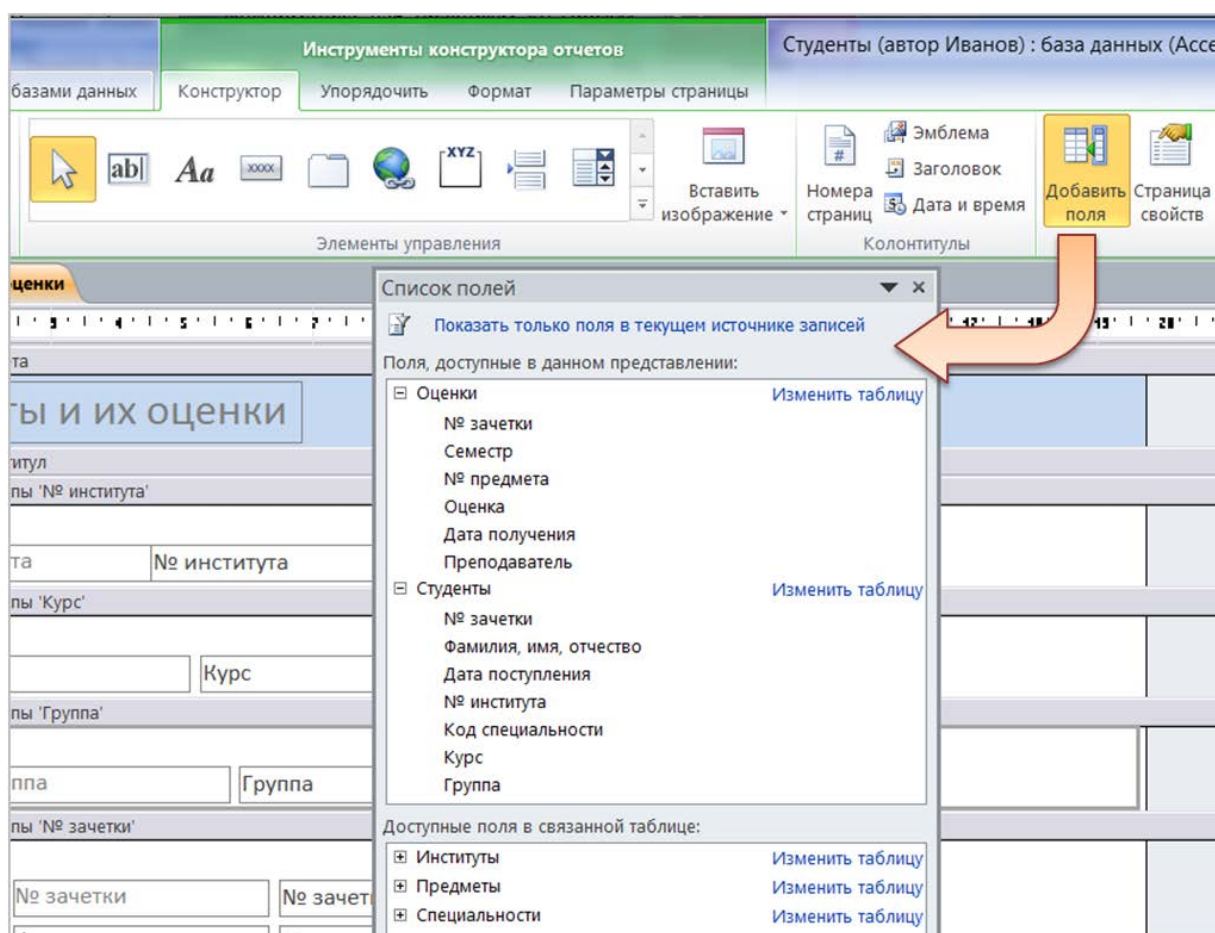


Рисунок 11.4 – Контекстная группа вкладок Инструменты конструктора отчетов и окно Список полей

Для использования в отчете данных справочных таблиц следует выбрать команду Добавить поля на вкладке Конструктор (см. рисунок 11.4), в верхней части открывшегося окна Список полей нажать на ссылку Показать все таблицы, после чего перетащить поля [Название института], [Название специальности] и [Название предмета] из справочных таблиц в соответствующие зоны отчета.

Кроме того, необходимо задать наличие **Примечаний** для всех групп, и в зону примечаний поместить вычисляемые поля, которые будут показывать средние оценки для студента, группы, факультета и для всего отчета. Для этого на вкладке **Конструктор** следует в группе команд *Группировка и итоги* нажать кнопку **Группировка**, после чего в нижней части окна базы данных откроется панель **Группировка, сортировка и итоги** (рисунок 11.5).

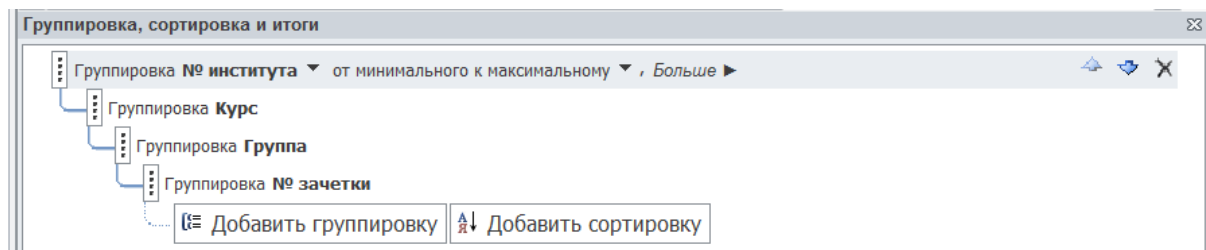


Рисунок 11.5 – Панель **Группировка, сортировка и итоги** для отчета

Для каждого уровня группировки нажмем кнопку *Больше* ►, в появившихся параметрах выберем *с разделом примечания*. Кроме того, зададим ненулевую высоту для зоны примечаний всего отчета.

Далее в **Конструкторе** отчета скопируем в зону примечаний для группы **№ зачетки** поле **[Оценка]**, в разделе *Данные* его **Окна свойств** изменим выражение на $=\text{Avg}([\text{Оценка}])$ для вычисления среднего значения поля, зададим на вкладке **Макет Окна свойств** **Формат поля** *Фиксированный* и **Число десятичных знаков** = 2, затем скопируем измененное поле в зоны примечаний других групп и всего отчета.

При создании вычисляемых полей на вкладке **Данные** **Окна свойств** можно нажать на кнопку с многоточием, после чего откроется **окно Построитель выражений**, где можно выбрать для построения выражений имеющиеся в системе функции и другие компоненты.

Разместим в отчете также надписи для итоговых значений, разлинуем таблицу (при этом для линий нужно задать свойство *Тип границы* – *Сплошная* и ее цвет).

Для группы **№ института** можно задать параметр **удерживать группу** на одной странице, тогда данные каждого института будут начинаться с новой страницы.

Вид отчета в **Конструкторе** после его модификации показан на рисунке 11.6.

Студенты и их оценки					
Заголовок отчета					
Студенты и их оценки					
Верхний колонтитул					
Заголовок группы '№ института'					
Институт	№ ил	Название института			
Заголовок группы 'Курс'					
Курс	Курс				
Заголовок группы 'Группа'					
Группа	Группа				
Специальность	Название специальности				
Код специаль					
Заголовок группы '№ зачетки'					
№ зачетки	№ зачетки	Фамилия, имя, отчество	Фамилия, имя, отчество		
Семестр	Предмет	Оценка	Дата	Преподаватель	
Область данных					
Сем	редл	Название предмета	Оценка	Дата получения	Преподаватель
Примечание группы '№ зачетки'					
Средняя оценка студента			g([Оцен		
Примечание группы 'Группа'					
Средняя оценка группы			g([Оцен		
Примечание группы 'Курс'					
Средняя оценка курса			g([Оцен		
Примечание группы '№ института'					
Средняя оценка по институту			g([Оцен		
Нижний колонтитул					
=Now()			="Стр. " & [Page] & " из " & [Pages]		
Примечание отчета					
Средняя оценка всех студентов			g([Оцен		

Рисунок 11.6 – Отчет после модификации его в Конструкторе

Вид фрагментов отчета при его печати на бумаге показан на рисунке 11.7.

Студенты и их оценки

Институт 1 **Институт информационных и космических технологий**

Курс 1

Группа **111**

Специальность **Химическая технология**

240100.62

№ зачетк **555321** Фамилия, имя, отчество **Вилегжанинова Елена Сергеевна**

Семестр	Предмет	Оценка	Дата	Преподаватель
1	3 Правоведение	4	17.07.2006	Рай А.И.
1	2 История России	5	15.07.2006	Веретнов М.Ю.
1	5 Информатика	4	21.07.2006	Коновалова Э.И.
1	1 Иностранный язык	5	11.07.2006	Орлова О.П.
1	4 Математика	4	19.07.2006	Шапаров Е.А.
Средняя оценка студента		4,40		

... И Т. Д. ...

№ зачетк **555354** Фамилия, имя, отчество **Мосеева Ольга Юрьевна**

Семестр	Предмет	Оценка	Дата	Преподаватель
1	3 Правоведение	4	17.01.2006	Орлова О.П.
1	4 Математика	5	20.01.2006	Шапаров Е.А.
1	1 Иностранный язык	4	12.01.2006	Орлова О.П.
1	2 История России	5	15.01.2006	Веретнов М.Ю.
1	5 Информатика	4	23.01.2006	Чернышова О.В.
Средняя оценка студента		4,40		
Средняя оценка группы		4,47		

Группа **112**

Специальность **Химическая технология**

240100.62

... И Т. Д. ...

№ зачетк **555269** Фамилия, имя, отчество **Перевернихата Анастасия Николаевна**

Семестр	Предмет	Оценка	Дата	Преподаватель
1	5 Информатика	5	20.01.2006	Чернышова О.В.
1	2 История России	5	13.01.2006	Хотенова О. А.
1	4 Математика	5	18.01.2006	Шапаров Е. А.
1	3 Правоведение	4	17.01.2006	Рай А.И.
1	1 Иностранный язык	4	10.01.2006	Орлова О.П.
Средняя оценка студента		4,60		
Средняя оценка группы		4,50		
Средняя оценка курса		4,18		
Средняя оценка по институту		4,18		
Средняя оценка всех студентов		4,28		

Рисунок 11.7 – Фрагменты отчета, напечатанного на принтере

2 Разработка запросов

Запросы при работе с базами данных в различных СУБД имеют близкий синтаксис, определяемый правилами языка SQL (Structured Query Language) и фактически являются текстовой командой на этом языке. Система **Access** использует команды языка SQL не только для отбора данных (команда **SELECT...**), но и для изменения множества данных одной командой **UPDATE...**, добавления множества записей одной командой **INSERT...** и удаления группы записей одной командой **DELETE ...**, а также для выполнения вычислений путем создания в запросах вычисляемых полей.

Основной тип – запрос для отбора данные по заданным сложным условиям из одной или нескольких таблиц баз данных, с показом результатов выполнения запроса в виде таблицы, либо с использованием его для форм и отчетов данных (команда SQL **SELECT**).

Принцип формирования запросов наиболее легко освоить при использовании Мастера запросов. Предположим, нам нужно отобрать тех студентов, которые по предмету высшая математика имеют только отличные оценки по результатам первого семестра.

Для создания запроса на вкладке Создание нажимаем кнопку Мастер запросов, после чего открывается окно с названием Новый запрос, в котором предлагается выбрать тип запроса, выберем первый тип – Простой запрос, после чего откроется окно Мастера запросов, **на первом шаге** которого требуется выбрать таблицы и поля для запроса (рисунок 11.8).

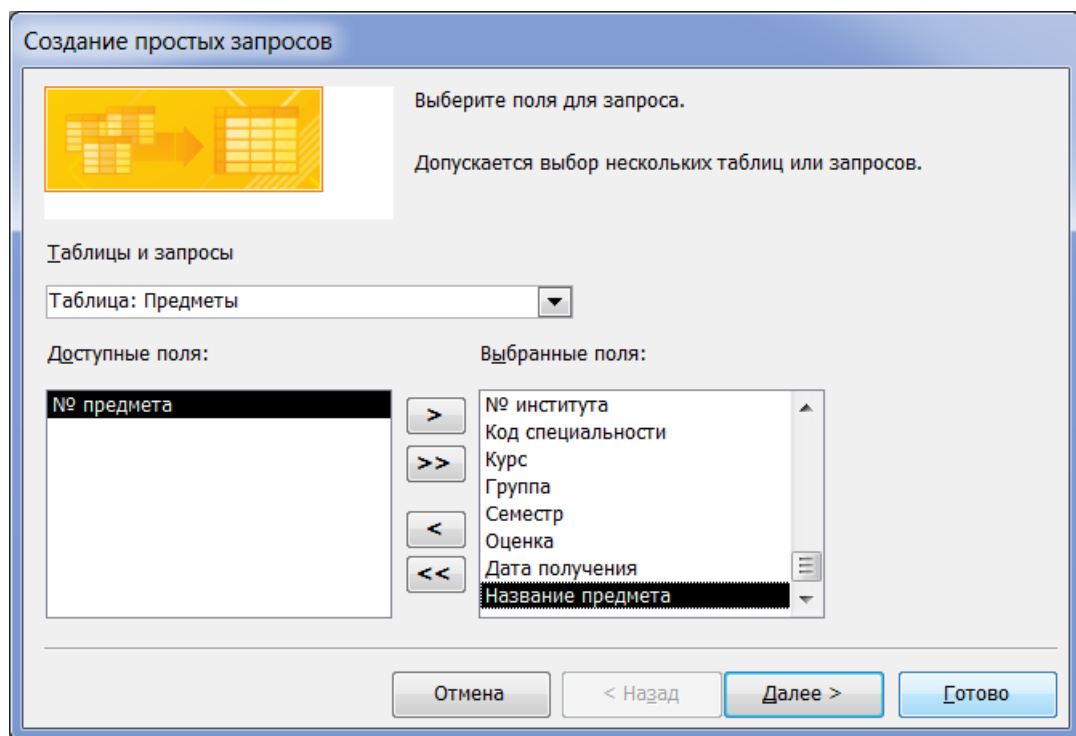


Рисунок 11.8 – Выбор полей на первом шаге Мастера запросов

Выбор полей может быть выполнен из нескольких таблиц базы. Для

нашего примера выбираем из таблицы **Студенты** все поля, кроме [Дата поступления], из таблицы **Оценки** 3 поля: [Семестр], [Оценка], [Дата получения] из таблицы **Предметы** поле [Название предмета].

На шаге 2 необходимо выбрать подробный или итоговый отчет, выбираем подробный.

На последнем **шаге 3** задаем название запроса Математика и выберем вариант Изменить макет запроса, после чего нажимаем кнопку Готово. Запрос открывается в **Конструкторе запросов**, его вид показан на рисунке 11.9.

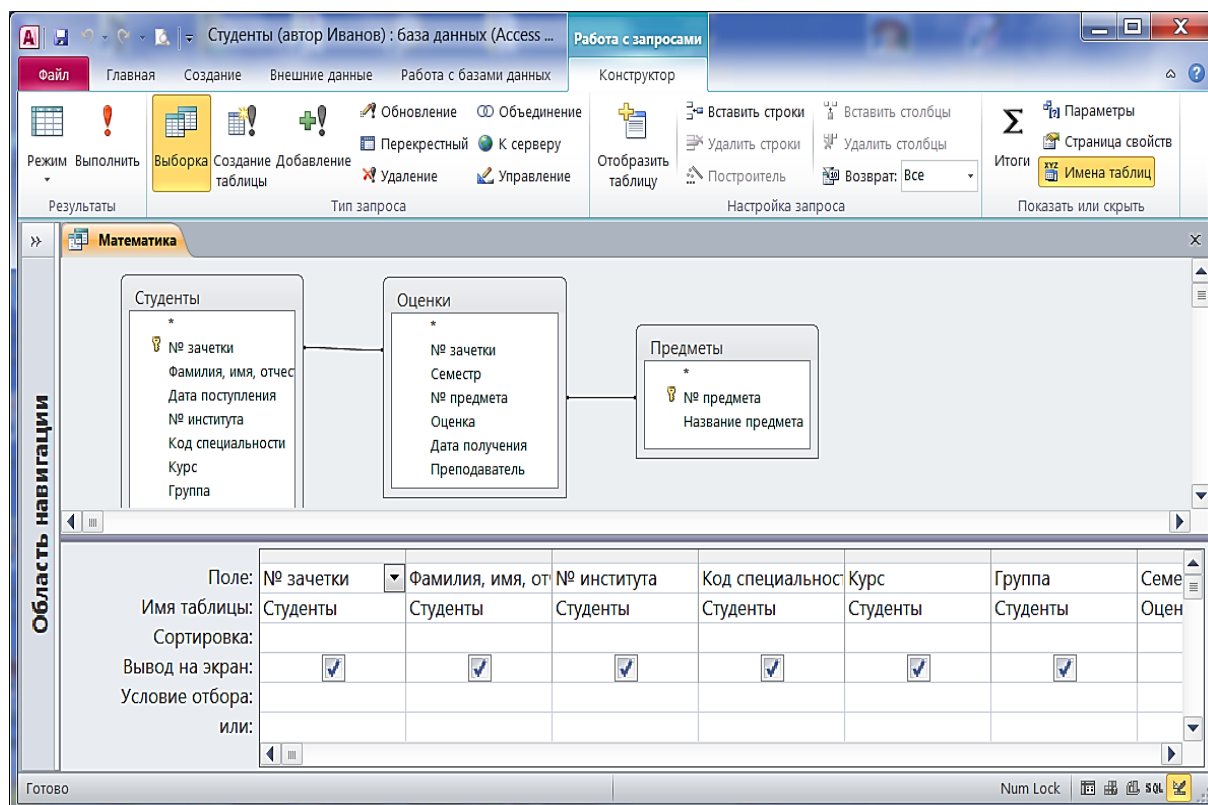


Рисунок 11.9 – Конструктор запросов

При работе с Конструктором запросов используется контекстная вкладка Работа с запросами – Конструктор (см. рисунок 11.8).

В верхней половине конструктора запросов показаны выбранные в Мастере таблицы базы данных и связи между ними, в нижней части – таблица для выбора полей, задания сортировки и условий отбора данных.

Модифицируем запрос для задания условия отбора данных и упорядочения студентов по их фамилии. Для этого в колонке поля [Фамилия, имя, отчество] зададим сортировку по возрастанию, для поля [Семестр] зададим условие отбора **1** (первый семестр), для поля [Оценка] зададим условие отбора **5** и для поля [Название предмета] зададим условие отбора **Математика**.

Сохраним запрос и посмотрим его текст в режиме **SQL** (Structured Query Language). Текст запроса будет выглядеть следующим образом:

```

SELECT Студенты.[№ зачетки], Студенты.[Фамилия, имя, отчество], Студенты.[№ института], Студенты.[Код специальности], Студенты.Курс, Студенты.Группа, Оценки.Семестр, Оценки.Оценка, Оценки.[Дата получения], Предметы.[Название предмета]
FROM Студенты INNER JOIN (Предметы INNER JOIN Оценки ON Предметы.[№ предмета] = Оценки.[№ предмета]) ON Студенты.[№ зачетки] = Оценки.[№ зачетки]
WHERE (((Оценки.Семестр)=1) AND ((Оценки.Оценка)="5") AND ((Предметы.[Название предмета])="Математика"))
ORDER BY Студенты.[Фамилия, имя, отчество];

```

Текст запроса состоит из нескольких частей:

- 1) после ключевого слова **SELECT** следует список полей, которые будут представлены в результате выполнения запроса (например, **Студенты.[№ зачетки]**– имя таблицы и её поле, разделенные точкой);
- 2) после слова **FROM** следует список таблиц, из которых будут отбираться данные, и условия объединения таблиц;
- 3) после слова **WHERE** следуют условия отбора данных;
- 4) после слов **ORDER BY** указаны условия упорядочения данных в запросе.

Закроем окно конструктора и выполним запрос двойным щелчком мышью. Результат отбора данных будет показан на экране в виде таблицы. Результаты выполнения запроса (а также и данные таблиц) можно представить в виде диаграмм и графиков. Создадим запрос, в котором покажем в графическом виде средний балл по студенческим группам по предмету **Математика**. Как и ранее, выбор полей для запроса выполним в Мастере, далее в Конструкторе запросов на вкладке ленты Работа с запросами – Конструктор нажмем на кнопку **Итоги**, после чего в нижней части окна Конструктора запросов появится строка Групповая операция для задания условий группировки данных и создания **итоговых вычисляемых полей** (рисунок 11.10). Зададим групповую операцию AVG (расчет среднего значения) для поля [Оценка] и условие отбора **Математика** для поля [Название предмета].

Получим следующий текст SQL-запроса:

```

SELECT Студенты.Группа, Avg(Оценки.Оценка) AS [Avg-Оценка]
FROM Студенты INNER JOIN (Предметы INNER JOIN Оценки ON Предметы.[№ предмета] = Оценки.[№ предмета]) ON Студенты.[№ зачетки] = Оценки.[№ зачетки]
GROUP BY Студенты.Группа, Предметы.[Название предмета]
HAVING (((Предметы.[Название предмета])="Математика"));

```

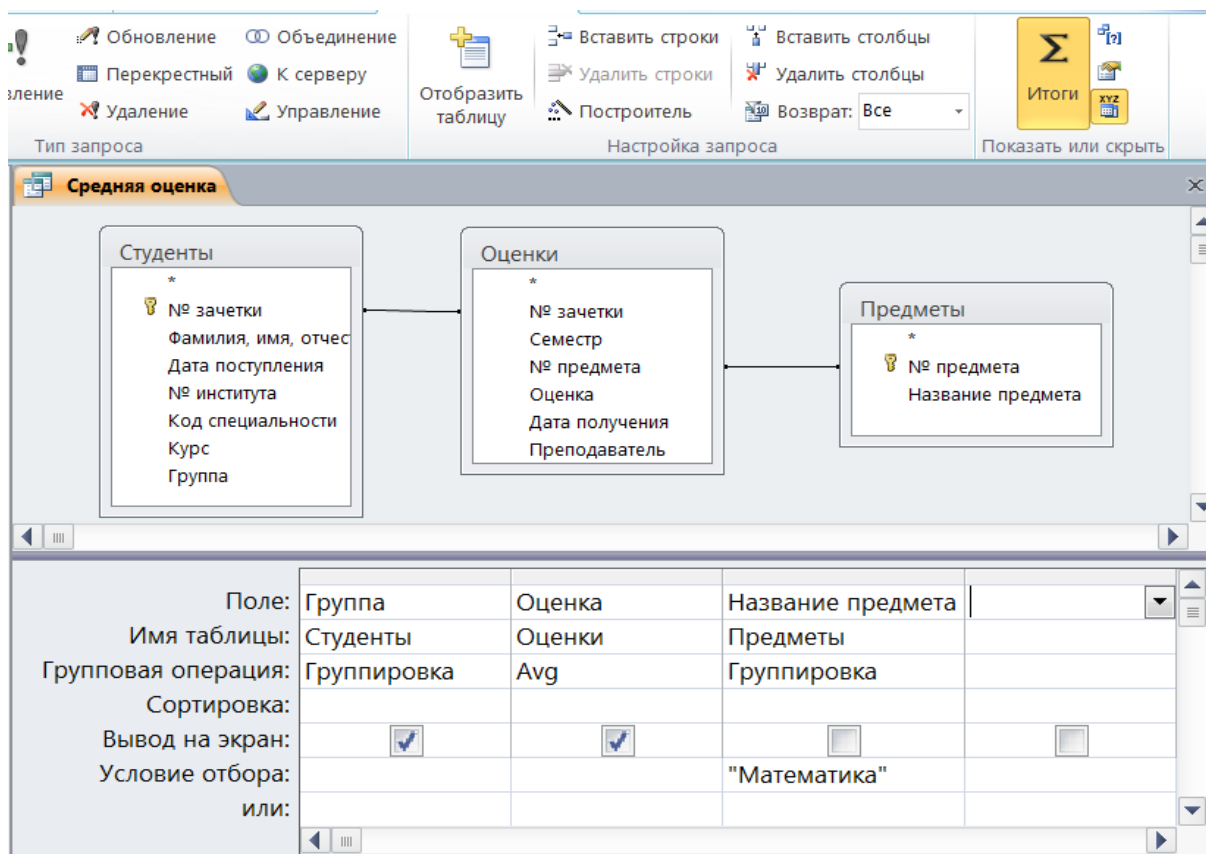


Рисунок 11.10 – Запрос с группировкой данных в Конструкторе

Для представления данных запроса в виде графика выбираем **Вид ► Сводная диаграмма**, после чего открывается окно **Построителя** диаграмм. Перетащим из окна Список полей диаграммы в поле для категорий (внизу диаграммы) поле запроса **Группа**, в поле для данных (вверху диаграммы) – поле **Avg-Оценка**. Получим Гистограмму, которую далее можно настроить в соответствии с правилами оформления. На рисунке 11.10 показана диаграмма для приведенного выше запроса, после ее модификации с использованием окна свойств.

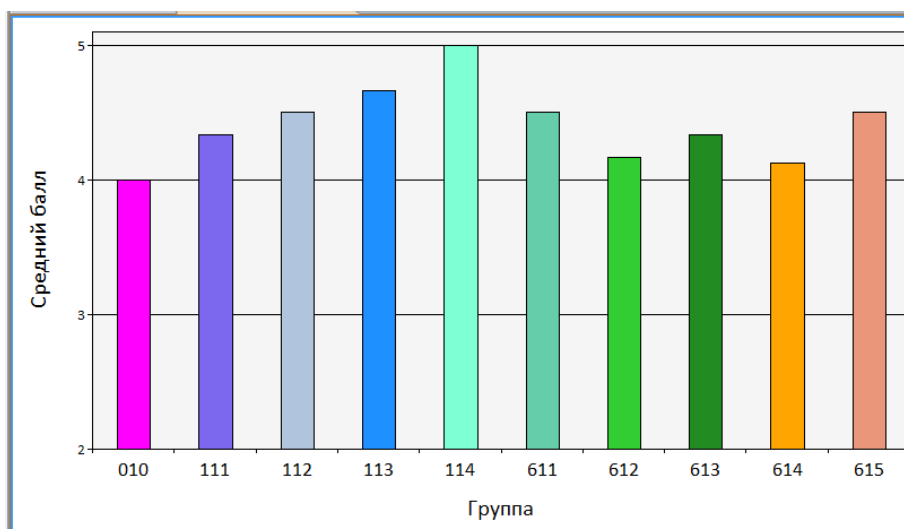


Рисунок 11.10 – Запрос с группировкой данных,

представленный в виде диаграммы

Учебное задание

1. Создать отчет с группировкой данных и расчетом средних оценок по группам по образцу в этой лабораторной работе.
2. Создать 2 запроса по примерам в этой лабораторной работе.

Лабораторная работа № 12. **Программирование на языке VBS. Работа с окнами сообщений и ввода данных**

Цель работы

Освоение основ написания и выполнения программы на языке VBS.

Учебные вопросы

1. Создание простейшей программы.
2. Синтаксис и примеры использования функции MsgBox.
3. Синтаксис и примеры использования функции InputBox.

Краткое изложение учебного материала по теме

1 Создание простейшей программы

Программа на языке **Visual Basic Scripting Edition** (VBS) состоит из инструкций языка (statement) в виде текстовых строк.

Несколько инструкций языка можно объединить в одну строку в текстовом файле программы с использованием разделителя строк – символа двоеточия (:) и наоборот, одну строку программы можно написать на нескольких строках в тексте с использованием символа подчеркивания (_).

В русском языке инструкции языка программирования обычно называют операторами языка, хотя это не совсем точно: операторами в английских первоисточниках называют символы для обозначения математических, логических и строковых операций (=, +, -, /, and, or, eqv, & и пр.). Далее будет использоваться традиционная русская терминология с использованием слова операторы для обозначения инструкций языка.

Текст программы можно написать в любом простейшем редакторе, сохраняющем файлы в кодировке ASCII, например, в стандартных программах Windows Блокнот или WordPad.

Существуют также специализированные редакторы, предназначенные для написания в них программ.

Далее воспользуемся редактором Блокнот, как наиболее доступным.

Для создания простейшей программы делаем следующее:

- 1) запускаем Блокнот;
- 2) пишем в Блокноте строку: **MsgBox "Привет!"** (это имя функции с аргументом – текстовым значением (константой) между апострофами; имя **MsgBox** – сокращение от английского выражения Message Box, которое дословно можно перевести, как «коробка сообщений», в системе Windows – окно сообщений);

- 3) сохраняем текстовый файл с именем **Prg1.vbs**;
- 4) двойным щелчком мыши запускаем файл на исполнение.

Результат работы этой программы – диалоговое окно в системе Windows (Windows-форма), показанное на рисунке 12.1.

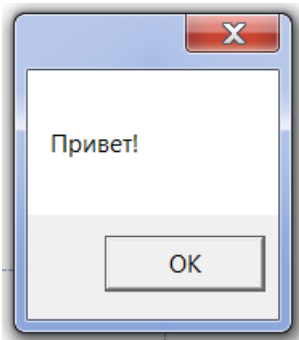


Рисунок 12.1 – Пример работы простейшей программы на языке *VBS*, исполняемой системой *Windows Based Script Host*

2 Синтаксис и примеры использования функции *MsgBox*

При выполнении этой программы используется стандартная функция языка *VBS* для вывода сообщений в окно Windows со следующим полным синтаксисом (*здесь и далее в описании синтаксиса в квадратных скобках [] приводятся необязательные элементы, элементы в скобках < > должны быть заменены конкретными значениями*):

```
[<р> = ] MsgBox ( <Сообщение>[, <Кнопки и значок>] _
[, <Заголовок окна>] [, <Справка, раздел>] )
```

где:

р – переменная, которой присваивается код нажатой кнопки;

Сообщение – аргумент функции, текст в диалоговом окне;

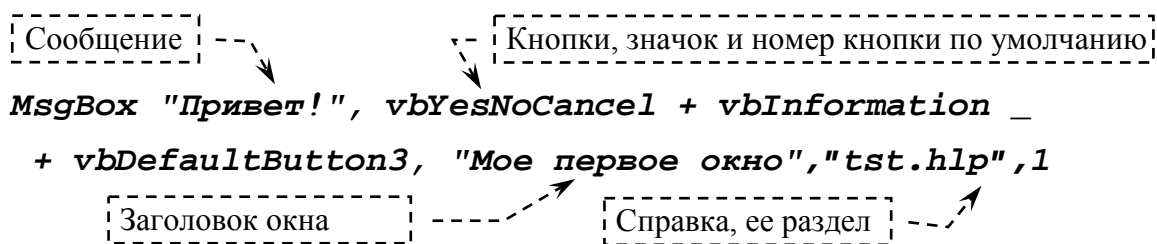
Кнопки и значок – стандартные переменные (приведены далее в таблице 1.1), определяющие кнопки, значок и номер кнопки по умолчанию в окне (например, *vbYesNoCancel + vbInformation + vbDefaultButton3* или *3+64+512*);

Заголовок окна – надпись на заголовке окна (например, «Мое первое окно»);

Справка, раздел – имя файла справки и идентификатор раздела, связанного с данным окном.

Аргументы функции следует писать в круглых скобках, если слева стоит переменная, которой присваивается значение, возвращаемое функцией, иначе аргументы следует писать за именем функции через пробел без скобок.

Дополним текст в файле **Prg1.vbs** указанными дополнительными параметрами (текст пишем в одну строку или используем знак подчеркивания **_** в конце первой строки для продолжения текста функции на следующей строке):



```
MsgBox "Привет!", vbYesNoCancel + vbInformation _
+ vbDefaultButton3, "Мое первое окно", "tst.hlp", 1
```

Окно запущенной программы будет иметь вид, показанный на рисунке 1.3.

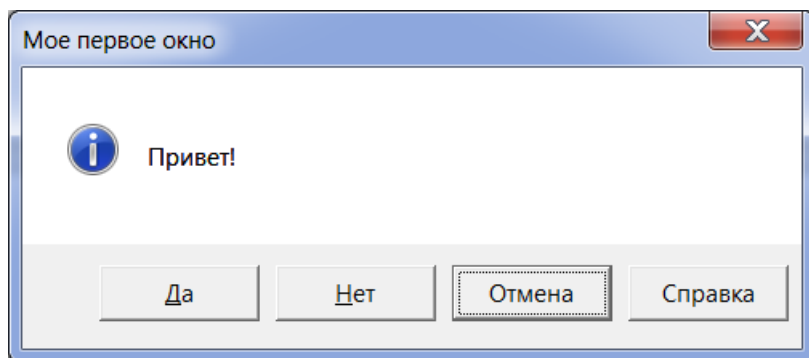


Рисунок 12.2 – Пример использования функции MsgBox с заданием набора кнопок, иконки и раздела справки

Функция **MsgBox** возвращает значение нажатой в окне кнопки (например **6**, если нажата кнопка **Yes (Да)**, либо другие значения для кнопок **vbNo**, **vbCancel** и пр., см. далее таблицу 12.1).

Таблица 12.1 – Константы диалоговых окон

Константа	Значение	Описание
vbOKOnly	0	Показана только кнопка OK
vbOKCancel	1	Показаны кнопки OK и Отмена (Cancel)
vbAbortRetryIgnore	2	Показаны кнопки Стоп (Abort), Повтор (Retry) и Пропустить (Ignore)
vbYesNoCancel	3	Показаны кнопки Да (Yes), Нет (No) и Отмена (Cancel)
vbYesNo	4	Показаны кнопки Да (Yes) и Нет (No)
vbRetryCancel	5	Показаны кнопки Повтор (Retry) и Отмена (Cancel)
vbCritical	16	Показан значок Stop Mark (знак стоп)
vbQuestion	32	Показан значок Question Mark (знак вопроса)
vbExclamation	48	Выводится значок Exclamation Mark (восклицательный знак)
vbInformation	64	Показан значок Information Mark (информационный знак)

vbDefaultButton1	0	По умолчанию в окне выбрана первая кнопка
vbDefaultButton2	256	По умолчанию в окне выбрана вторая кнопка
vbDefaultButton3	512	По умолчанию в окне выбрана третья кнопка
vbDefaultButton4	768	По умолчанию в окне выбрана четвёртая кнопка
vbSystemModal	4096	Диалоговое окно выводится в модальном режиме и располагается сверху всех других окон
vbOK	1	Нажата кнопка ОК .
vbCancel	2	Нажата кнопка Отмена (Cancel)
vbAbort	3	Нажата кнопка Стоп (Abort)
vbRetry	4	Нажата кнопка Повтор (Retry)
vbIgnore	5	Нажата кнопка Пропустить (Ignore)
vbYes	6	Нажата кнопка Да (Yes)
vbNo	7	Нажата кнопка Нет (No)

Для определения кода нажатой в окне **MsgBox** кнопки следует использовать следующий синтаксис функции: слева нужно написать переменную, которой будет присвоено возвращаемое функцией значение, далее следует написать символ присваивания (=) и справа от него функцию, у которой аргументы написаны в круглых скобках:

```
btn = MsgBox("Привет!", vbYesNoCancel + vbInformation _
+ vbDefaultButton3, "Мое первое окно")
```

Диалоговое окно будет иметь тот же вид, что и раньше (только без кнопки **Справка**, см. рисунок 12.2), но после нажатия кнопки в окне переменная **btn** будет иметь значение, соответствующее нажатой кнопке.

Если Вы желаете написать в окне **Сообщение** и **Заголовок окна**, пропустив второй аргумент (**Кнопки и значок**), после первого аргумента следует поставить 2 (**ДВЕ!**) запятые:

```
MsgBox "Сегодня я написал свою первую программу на VBS!" _
, , "Окно сообщений студента Вани Иванова"
```

↑
--- [Здесь 2 запятые, т. к. пропущен аргумент <Кнопки и значок>]

3 Синтаксис и примеры использования функции *InputBox*

Еще одна функция языка, позволяющая открывать окно для ввода пользователем с клавиатуры строки текста (максимальная длина строки 256 символов):

```
[<p> = ] InputBox( <Сообщение>[,<Заголовок окна>] _
    [,<Стр.умолч.>][,X][,Y][, <Справка, раздел>])
```

где новые параметры функции (по сравнению с **MsgBox**):

p – переменная, которой присваивается значение, введенное в поле ввода окна **InputBox**;

Стр.умолч. – строковое значение в поле ввода, которое будет показано по умолчанию при открытии окна;

X, Y – координаты левого верхнего угла окна в единицах **twips** (1440 twips = 1 дюйм, 567 twips = 1 см) по отношению к левому верхнему углу экрана; если координаты не указаны, окно выводится примерно в центре экрана.

Дополним программу вводом данных с клавиатуры в окне **InputBox** :

```
S = InputBox(vbLF & "Напишите строку текста:", _
    "Окно ввода. Студент Ваня Иванов", 4000, 2000)
```

```
Kod = MsgBox (S, vbYesNoCancel+vbInformation, _
    "Окно сообщений. Студент Ваня Иванов")
```

Здесь 2 запятые, т. к. пропущена
<Стр.умолч.>

Функция **InputBox** в данном примере имеет строку **<Сообщение>**, состоящую из двух частей: **vbLF** – константа языка VBS – код перехода на следующую строку, и текстового значения **"Напишите строку текста:"**; эти две части объединены в одну строку с использованием оператора конкатенации **&**.

Сохраним текст, как новый файл **Prg2.vbs**, откроем его двойным щелчком мышкой и увидим окно функции **InputBox** (рисунок 12.3). В поле ввода этого окна напишем новый текст (длиной более 256 символов) и нажмем кнопку **ОК**, после чего появится окно, показанное на рисунке 12.4, в сообщении которого присутствует 256 символов значения переменной **S** (если в первом окне нажать кнопку **Отмена**, сообщение в окне **MsgBox** будет отсутствовать).

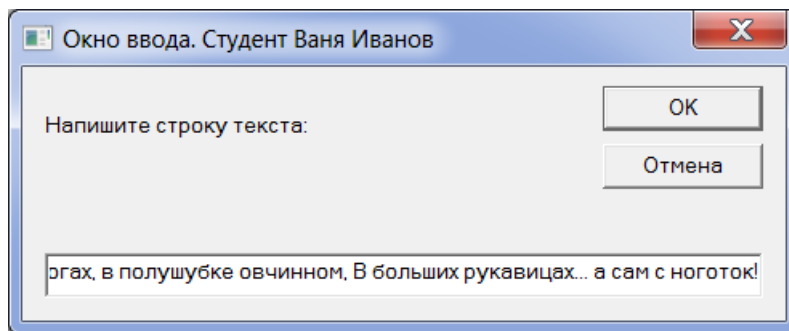


Рисунок 12.3 – Пример использования функции **InputBox**

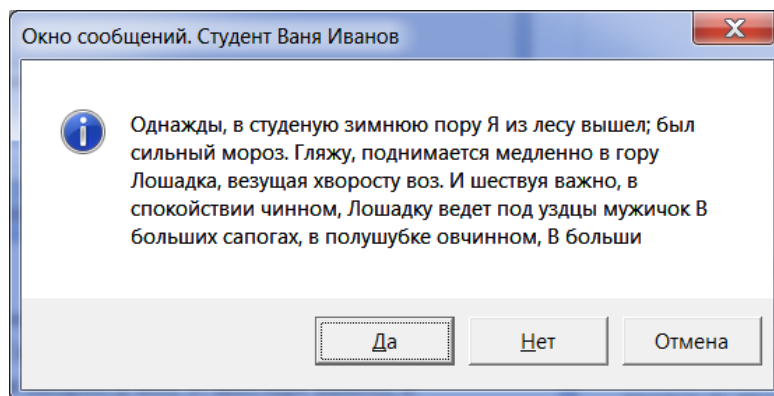


Рисунок 12.4 – Окно функции **MsgBox** с показом текста, написанного в окне **InputBox**

В изложенном выше материале использовались такие основополагающие понятия языка программирования, как строковые значения (символы между апострофами), стандартные константы окон и строковая константа **vbLF** – код перехода на новую строку, переменные (**btn**, **S**, **Kod**), операции присваивания (=) и конкатенации (& – объединение двух выражений любого типа в одну строку), функции с параметрами (**MsgBox**, **InputBox**).

Последующие лабораторные работы посвящены освоению использования всех этих компонентов языка VBS.

Учебное задание

Написать программу для варианта задания, соответствующего номеру Вашего компьютера. Выполнить программу, сохранить ее текст и скриншоты окон в отчете. Все окна должны иметь заголовки следующего вида: «Окно ввода (или сообщений) <Фамилия имя отчество студента>».

- 1) Вывести в левый верхний угол экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с одной кнопкой **ОК** и значком Information Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна.
- 2) Вывести на расстоянии 10 см по горизонтали и вертикали от левого верхнего угла экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **Да** и **Нет** и значком Exclamation Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.
- 3) Вывести в центре экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **Повтор** и **Отмена** и значком Stop Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.

- 4) Вывести примерно в правом нижнем углу экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **Стоп**, **Повтор** и **Пропустить** и значком Question Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.
- 5) Вывести в центре экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **ОК** и **Отмена** и значком Exclamation Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.
- 6) Вывести примерно в правом нижнем углу экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **Да**, **Нет** и **Отмена** и значком Question Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.
- 7) Вывести на расстоянии 20 см по горизонтали и 15 см по вертикали от левого верхнего угла экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **Стоп**, **Повтор** и **Пропустить** и значком Information Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.
- 8) Вывести примерно в правом верхнем углу экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **Да**, **Нет** и **Отмена** и значком Exclamation Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.
- 9) Вывести слева примерно в центре по вертикали экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **Стоп**, **Повтор** и **Пропустить** и значком Question Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.
- 10) Вывести справа примерно в центре по вертикали экрана окно для ввода текстовой строки, показать эту строку в окне сообщений с кнопками **Да** и **Нет** и значком Information Mark. Затем в окне сообщений показать код нажатой кнопки при выходе из предыдущего окна. Определить коды нажатия для всех кнопок.

Лабораторная работа № 13. Программирование на языке VBS. Типы данных. Константы. Переменные

Цель работы

Освоение понятий алгоритмического языка типы данных, константы и переменные.

Учебные вопросы

1. Типы и подтипы данных в языке VBS.
2. Константы в алгоритмическом языке.
3. Переменные в алгоритмическом языке.

Краткое изложение учебного материала

1 Типы и подтипы данных в языке VBS

Каждый язык программирования предназначен для обработки информации (данных) различных типов. Используемые типы данных и методы их обозначения и обработки могут несколько различаться в различных алгоритмических языках.

Типы данных определяют:

- формат представления данных в памяти компьютера;
- область или диапазон возможных значений;
- множество допустимых операций, применимых к данным.

В языке *Microsoft Visual Basic Scripting Edition* определен единственный тип данных – **Variant**. Это специальный тип, который может содержать в себе различные виды информации. Все функции языка также возвращают данные типа **Variant**.

Различные виды информации, которая может содержаться в типе **Variant**, называются подтипами.

В таблице 2.1 приведены подтипы данных, которые могут содержаться в типе **Variant**.

Таблица 13.1 – Подтипы данных в языке VBS

Подтип	Описание
Byte	Целые числа в диапазоне от 0 до 255
Boolean	Логические значения <i>True</i> или <i>False</i>
Integer	Целые числа в диапазоне от -32768 до 32767
Long	Целые числа в диапазоне от -2 147 483 648 до 2 147 483 647
Single	Числа одинарной точности с плавающей точкой в диапазоне от -3.402823E38 до -1.401298E-45 для отрицательных значений; от

	1.401298E-45 до 3.402823E38 для положительных значений
Double	Числа двойной точности с плавающей точкой в диапазоне от -1.79769313486232E308 до -4.94065645841247E-324 для отрицательных значений; 4.94065645841247E-324 до 1.79769313486232E308 для положительных значений
Currency	-922 337 203 685 477.5808 до 922 337 203 685 477.5807
Date / (Time)	Числа, которые представляют даты и время в диапазоне между 1-01-100 0:0:0 до 31-12-9999 23:59:59
Object	Содержит объект
String	Строка переменной длины, которая максимально может содержать 2 миллиона символов
Empty	Неинициализированное значение (0 для числовых переменных, строка нулевой длины ("") для строковых переменных)
Null	Содержит неверные для подтипа данные
Error	Содержит номер ошибки

Функция **VarType** возвращает информацию о том, как данные сохранены в типе **Variant**. Для преобразования одного подтипа в другой могут использоваться соответствующие функции (**Cbyte**, **Cdate**, **CSng**, **CDbl** и др.).

Другие диалекты языка Visual Basic также имеют тип **Variant**, но наряду с ним могут определять переменные различных **типов**, таких же, как **подтипы** языка VBS.

2 Константы в алгоритмическом языке

В тексте программы могут использоваться числа, строки текста, даты и время, которые являются константами.

Константа – некоторое неизменяемое значение в тексте программы. Константа может иметь имя (*идентификатор*).

Для тех констант, которые используются часто, можно задать имена. Задание имен константам делает программы легко читаемыми. Для этого в любом месте текста программы можно использовать следующее описание:

```

Const N = 1.15e-15           'числовая константа
Const FIO = "Иванов Иван Иванович" 'текстовая константа
Const Data_r = #05-13-1988 06:30:00# 'константа дата/время
Const Time_r = #06:30:00#    'константа дата/время

```

Как видно из примера, для числовых констант разделителем целой и дробной части является точка, можно использовать экспоненциальный вид чисел ($1.15e-15 = 1.15 \times 10^{-15}$). Значения строковых констант следует писать между двумя кавычками (" "), даты и времени – между двумя знаками решетки (# #).

В языке VBS существует достаточно большое количество predefined констант, которые сгруппированы по категориям:

- **Date Format Constants** – определяют форматы дат и времени (например, vbGeneralDate – дата и время выводятся в формате, определяемом региональными настройками системы);
- **MsgBox Constants** – используются в функции **MsgBox** и других диалоговых окнах (см. таблицу 13.1);
- **String Constants** – определяют скрытые символы, используемые для манипуляции со строками:
 - *vbCr* – возврат каретки (Chr(13), переход в начало следующей строки);
 - *vbLf* – новая строка (Chr(10));
 - *vbCrLf* – новая строка (Chr(13) + Chr(10));
 - *vbNewLine* – новая строка (Chr(10) или Chr(13) + Chr(10));
 - *vbNullChar* – символ с нулевым значением (Chr(0));
 - *vbNullString* – строка с нулевым значением (Chr(0));
 - *vbTab* – горизонтальная табуляция (Chr(9)) ;
 - *vbVerticalTab* – вертикальная табуляция (Chr(11)) ;
- **VarType Constants** – определяют форматы для различных подтипов (*vbEmpty, vbNull, vbInteger, vbLong, vbSingle, vbDouble, vbCurrency, vbDate, vbString, vbObject, vbError, vbBoolean, vbVariant, vbDataObject, vbDecimal, vbByte, vbArray*);
- и многие другие.

3 Переменные в алгоритмическом языке

Переменная – имя (*идентификатор*) в программе, связанное с областью оперативной памяти компьютера, предназначенной для хранения какой-либо информации, которая может изменяться во время работы программы.

Все переменные в языке VBS имеют один тип – **Variant** и во время использования могут хранить данные разных подтипов.

Правила написания идентификаторов переменных, констант, названий процедур, функций, объектов, их методов и свойств следующие:

- 1) идентификатор должен начинаться с латинской буквы;
- 2) может состоять из латинских строчных и прописных букв, цифр и символа подчеркивания;
- 3) длина его – не более 255 символов;
- 4) буквы в верхнем и нижнем регистре *не* различаются;
- 5) он должен быть уникален в области определения.

Для объявления переменных могут служить выражения:

```
Dim X, Y, Z  
Public A, B, C  
Private X1, X2, X3
```

Однако, переменные в языке Basic можно и не объявлять с помощью этих описаний, достаточно написать в программе новый идентификатор и присвоить ему значение, после чего транслятор будет знать, что это переменная, например:

```
Z = 1.2345 'переменной с именем Z присвоено числовое значение 1,2345  
S = "Строка текста"  
dt = #12-31-08#
```

Разница между описаниями **Dim**, **Public**, **Private** существенна при использовании внутри программы функций пользователя. В этом случае различается область действия переменных и массивов.

При отсутствии явного описания все переменные, используемые в головной программе, являются глобальными, отсутствующие в головной программе, но используемые в подпрограмме – локальные.

Если в начале программы написать строку:

```
Option Explicit
```

использование переменных без их явного описания в выражениях **Dim**, **Public** и **Private** будет запрещено. Попытка использовать необъявленную переменную вызовет сообщение об ошибке при выполнении программы.

Явное описание переменных способствует уменьшению количества ошибок при программировании, в частности, предотвращает конфликты между переменными, используемыми в основной программе и в процедурах при одинаковых именах; позволяет выявить неверно написанные имена предопределенных констант языка (иначе неверные имена просто игнорируются).

Учебное задание

В вариантах заданий в скобках < > задано значение переменных, которые нужно получить, в скобках () – номера констант Вашего задания. Значения переменных следует формировать из констант, пробелов и знаков препинания с использованием оператора конкатенации &.

1. Задайте в программе 3 константы и присвойте им значения 1) Вашего имени, 2) отчества, 3) фамилии. Создайте 2 переменные и присвойте им значения: первой – <(3), (1), (2)>, второй – <(1) (2) (3)>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).

2. Задайте в программе 3 константы и присвойте им значения Ваших 1) дня, 2) название месяца, 3) года рождения. Создайте 2 переменные и присвойте им значения: первой – <(1).(2).(3)>, второй – <Я родился (1) (2) (3) года .>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).
3. Задайте в программе 4 константы и присвойте им значения данных Вашего адреса проживания: 1) город; 2) улица 3) номер дома; 4) номер квартиры. Создайте 2 переменные и присвойте им значения: первой – <(1), (2), (3), (4)>, второй – <Я живу в городе (1) на улице (2) в доме (3), квартира (4).>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).
4. Задайте в программе 4 константы и присвойте им значения данных о Вашей учебе: 1) учебное заведение; 2) специальность 3) курс; 4) группа. Создайте 2 переменные и присвойте им значения: первой – <(1),(2), (3), (4)>, второй – <Я учусь в (1) на специальности (2) курс (3), группа (4).>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).
5. Задайте в программе 3 константы и присвойте им значения данных о Вашей учебе в школе: 1) населенный пункт; 2) № школы; 3) любимый предмет. Создайте 2 переменные и присвойте им значения: первой – <(1), (2), (3)>, второй – <Мой любимый предмет был (3), когда я учился в (2)-й школе города (или название другого типа населенного пункта) (1).>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).
6. Задайте в программе 4 константы и присвойте им значения паспортных данных (придумать близкие к возможным): 1) серия; 2) №; 3) кем выдан; 4) дата выдачи. Создайте 2 переменные и присвойте им значения: первой – <(1) – (2) , (3) (4)>, второй – <Паспортные данные: серия (1), номер (2), выдан (4) (3).>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).
7. Задайте в программе 3 константы и присвойте им значения Ваших антропометрических данных: 1) рост в см; 2) вес в кг; 3) окружность груди; 4) талии; 5) бедер, в см. Создайте 2 переменные и присвойте им значения: первой – <(1), (2), (3) – (4) – (5)>, второй – <Мой рост (1) см, вес (2), окружность груди, талии и бедер (3) – (4) – (5) см >. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).
8. Задайте в программе 3 константы и присвойте им названия предметов Вашего сегодняшнего расписания: 1) 1-я пара; 2) 2-я пара 3) 3-я пара. Создайте 2 переменные и присвойте им значения: первой – <(1) – (2) – (3)>, второй – <8.20-9.05 9.10-9.55 – (1); 10.10-10.55 11.00-11.45 – (2);

12.00-12.45 12.50-13.35 – (3)>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).

9. Задайте в программе 3 константы и присвойте им названия окон VBS: 1) **MsgBox**; 2) **InputBox** 3) **Popup**. Создайте 2 переменные и присвойте им значения: первой – <Окна VBS: (1), (2), (3)>, второй – <Их назначение и особенности: (1) – (здесь написать назначение), (2) – (здесь об этом окне), (3) – (здесь особенности этого окна)>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).
10. Задайте в программе 6 констант и присвойте им названия подтипов данных VBS для работы с числами. Создайте 6 переменных и присвойте им значения: <(подтип) – диапазон данных (здесь указать диапазон)>. Покажите константы и переменные в модальном окне MsgBox (каждую в отдельной строке).

Лабораторная работа № 14. **Программирование на языке VBS. Массивы**

Цель работы

Освоение понятия массив алгоритмического языка

Учебные вопросы

1. Описание массивов.
2. Динамический массив.
3. Работа с элементами массива.

Краткое изложение учебного материала

Переменная, которая хранит единственное значение, называется скалярной. Переменная может использоваться и для хранения серии чисел, такая переменная является массивом.

Массив – переменная, предназначенная для хранения пронумерованной серии значений (элементов массива).

1 Описание массивов

Скалярная переменная может использоваться без явного описания с использованием ключевых слов **Dim**, **Public** или **Private**, если в начале программы не присутствует директива **Option Explicit**.

Массив обязательно должен быть описан перед его использованием.

Для описания массивов используются те же операторы, что и для переменных, только после имени переменной в круглых скобках указывается количество индексов и их максимальное значение у элементов массива.

Полный синтаксис этих описаний следующий:

```
Dim имя_м([индексы])[, имя_м([индексы])]...
```

```
Public имя_м([индексы])[, имя_м([индексы])]...
```

```
Private имя_м([индексы])[, имя_м([индексы])]...
```

где:

Имя_м – имя массива;

Индексы – имеет формат: **индекс1[индекс2]...** – максимальные значения индексов, минимальное значение равно 0; массив может быть одномерный, двумерный и т. д. до 60.

Пример:

```
Dim X(99), Y(24,24), Z(99,99,99)
```

где:

X(99) – одномерный массив из 99 элементов;

Y(24,24) – двухмерный массив размерностью 24×24 элемента;

Z(99,99,99) – трехмерный массив размерностью 99×99×99.

Разница между описаниями **Dim**, **Public**, **Private** существенна при использовании внутри программы подпрограмм и функций пользователя. В этом случае различается область действия переменных и массивов.

Массив в языке VBS после его объявления имеет тип **Variant**, поэтому его элементам можно присваивать значения различных типов. Максимальный размер массива ограничен размером свободной виртуальной памяти операционной системы.

2 Динамический массив

Массив может быть динамический (изменяемой размерности), при его описании в круглых скобках размерность не указывают:

```
Dim имя_м( )[, имя_м( )]...
```

Для инициализации динамического массива следует использовать оператор:

```
ReDim [Preserve] имя_м(индексы)[, имя_м(индексы)]...
```

Параметр **Preserve** используется, если выполняется повторная инициализация для изменения размера массива и необходимо сохранить значения переменных, которые им уже были присвоены.

Пример:

```
Dim X(), Y()
```

```
ReDim X(10), Y(10)
```

```
...
```

```
ReDim Preserve X(10, 10), Y(10, 10)
```

Другой способ создания переменной типа **Variant**, содержащей одномерный массив, – с помощью функции **Array**:

```
A = Array(10,20,30,40)
```

Эта функция описывает массив и задает значения его элементам. В примере значения элементов массива будут следующими:

$A(0)=10$, $A(1)=20$, $A(2)=30$ и $A(3)=40$.

3 Работа с элементами массива

Пример программы с использованием массивов:

```
Dim Y(99), Z(99,99,99)
Y(0) = "Это 1-ый элемент массива Y - строка"
Y(1) = 123.456 ' Тип второго элемента - число
Y(99) = #12-30-2007# ' Тип 100-го элемента - дата
Z(99,99,99) = "Это элемент трехмерного массива Z " _
    & "с индексом 99,99,99"
MsgBox Y(0) & vbLf & "2-ой элемент массива Y - число: " _
    & Y(1) & vbLf & "100-ый элемент массива Y - дата: " _
    & Y(99) & vbLf & Z(99,99,99), , "Студент Иванов И. И."
```

Окно, которое показывает эта программа, показаны на рисунке 14.1.

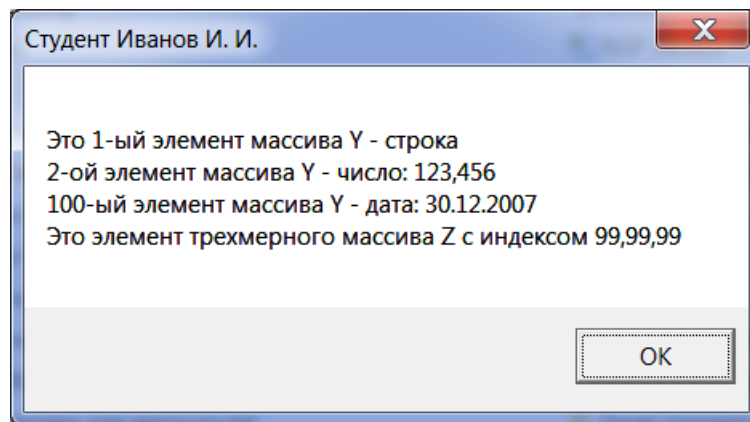


Рисунок 14.1 – Пример на использование массивов

Учебное задание

- 1) Опишите в программе два одномерных массива размерностью 3 и 5 элементов, присвойте значения элементам первого массива – нечетные числа, начиная с 21, второго – буквы русского алфавита, начиная с мягкого знака. Покажите все данные в окне сообщений.
- 2) Опишите в программе динамический массив. Выполните вначале его инициализацию для размерности 3 элемента и присвойте значения элементам массива – любые числа. Покажите все данные в 1-м окне сообщений. Затем выполните повторную инициализацию для размерности 7 с сохранением значений определенных ранее элементов. Присвойте элементам с 4 по 7-й любые даты. Покажите все данные во 2-м окне сообщений.
- 3) Опишите в программе двухмерный массива размерностью 2×3 элементов и присвойте значения каждому элементу массива – время в диапазоне от 7:00 до 19:00. Покажите данные в окне сообщений в виде мат-

рицы, в которой номер строки – первый индекс, в строке изменяется второй индекс.

- 4) Задайте с помощью функции **Array** значения 5-ти элементам массива, представляющим собой геометрическую прогрессию. Покажите все данные в окне сообщений.
- 5) Создайте с помощью функции **Array** одномерный массив, состоящий из 6-ти чисел. Покажите данные в окне сообщений. С помощью функции **ReDim** переопределите его размерность до двухмерного размерностью 3×2 . Задайте значения всем его элементам и покажите их в окне сообщений в виде матрицы, в которой номер строки – первый индекс, в строке изменяется второй индекс.
- 6) Опишите в программе трехмерный массив размерностью $2 \times 3 \times 4$ элементов и присвойте числовые значения элементам массива. Покажите данные в окне сообщений с указанием элемента массива и его значение (например, $A(0,0,0)=1$ и т. д.).
- 7) Задайте элементам двумерного массива текстовые значения – каждому одно слово какого-либо четверостишия. Покажите элементы массива в окне сообщений в виде стихотворения.
- 8) Опишите в программе два одномерных массива X и Y размерностью 5 элементов, присвойте числовые значения элементам массивов. Покажите данные в окне сообщений в виде таблицы, в первой строке которой показаны имена массивов, в последующих – значения их элементов.
- 9) Опишите в программе одномерный массив из 7 элементов. Присвойте значения элементам – целые числа. Покажите элементы массива в окне сообщений в следующем порядке: 1, 7, 2, 6, 3, 5, 4.
- 10) Опишите в программе два одномерных массива размерностью 5 элементов, присвойте числовые значения элементам массивов. Покажите данные в окне сообщений: в первой строке элементы первого массива от первого до 5-го, во второй строке – элементы второго массива от 5-го до первого.

Лабораторная работа № 15. Программирование на языке VBS. Условные операторы IF и CASE

Цель работы

Освоение приемов организации разветвляющихся алгоритмов в программе на алгоритмическом языке.

Учебные вопросы

1. Строчный синтаксис оператора условного перехода.
2. Блочный синтаксис оператора условного перехода.
3. Оператор выбора Case.

Краткое изложение учебного материала

Оператор условного перехода IF позволяет выполнить те или иные строки программы в зависимости от логических условий.

В языке VBS он может использоваться в двух различных видах (строчный и блочный синтаксисы).

1 Строчный синтаксис оператора условного перехода

If <условие> Then <операторы1> [Else <операторы2>]

где:

условие – логическое выражение, результатом вычисления которого может быть истина (**True**), ложь (**False**) или **Null** которое приравнивается к **False**;

операторы1 – один оператор или более (разделенных двоеточиями для строкового синтаксиса); выполняются, если условие истинно (**True**);

операторы2 – выполняются, если **условие** не является истиной (**False**).

В логических выражениях могут использоваться следующие основные операторы сравнения и логические операции:

=	Равно	And	Логическое «И»
<>	Не равно	Or	Логическое «ИЛИ»
<	Меньше	Xor	Логическое исключение (E1 Xor E2 возвращает True, если только E1 = True или только E2 = True, иначе – False)
>	Больше	Eqv	Логическое «эквивалентно»
<=	Меньше или равно	Imp	Логическая импликация
>=	Больше или равно		
Is	Сравнение объектов		

Not *Логическое отрицание*

(E1 Imp E2 возвращает False, если E1 = True и E2 = False, иначе –True)

Для простых условных операторов следует использовать строчный синтаксис.

Два примера строчного синтаксиса:

```
If A <= 0 Then A = A + 1 : B = B + A Else B = B + A
```

```
If X > 0 and X <= 1 Then Y = 1 else Y = 0
```

2 Блочный синтаксис оператора условного перехода

Блочный синтаксис является более структурированным, имеет большие возможности, легче читается и отлаживается. В одном операторе может быть выполнена проверка нескольких условий с заданием различных исполняемых фрагментов программы.

```
If <условие> Then  
    [операторы]  
    [ElseIf <условие-n> Then  
        [операторы-n]] ...  
    [Else  
        [else-операторы]]  
End If
```

где:

условие – логическое выражение, результатом вычисления которого может быть истина (**True**), ложь (**False**) или **Null** которое приравнивается к **False**;

операторы – один оператор или более (разделенных двоеточиями для строкового синтаксиса), которые выполняются, если условие истинно (**True**);

условие-n – то же, что и **условие**;

операторы-n – выполняются, если **условие-n** является истиной (**True**);

else-операторы – один оператор или более, выполняющиеся, если предшествующие условия не были истинны.

Когда выполняется блочный **If**, проверяется **условие**, и, если оно истинно (**True**), выполняются **операторы**, следующие за **Then**. Если **условие** не является истинным (**False**), каждое **условие-n**, идущее за **Elseif** (если они есть) проверяется. Когда истинное значение найдено, выполняются **операторы-n**, следующие за **Then** после истинного условия, после чего программа выходит за **End If** (т. е. последующие **Elseif**, если они есть, не проверяются). Если истинных условий для **Elseif** не найдено, выполняются **else-операторы**, следующие за **Else**.

Пример блочного синтаксиса:

```
FIO = "Лютикова Лилия Максимовна"  
a = InputBox("Задайте значение переменной a", "Пример  
IF. " & FIO)  
a = Eval(a) 'преобразование строки в число  
If a > 10 Then  
    b = "a > 10"  
ElseIf a > 0 Then  
    b = "a > 0" 'строка будет выполнена даже при a=1!  
ElseIf a = 1 Then  
    b = "a = 1" 'эта строка никогда не будет выполняться!  
Else  
    b = "Нет данных для заданного значения a"  
End If  
MsgBox "Результат выполнения IF для a = "& a & ": " &  
b,,FIO
```

Примеры исполнения программы показаны на рисунке 15.1.

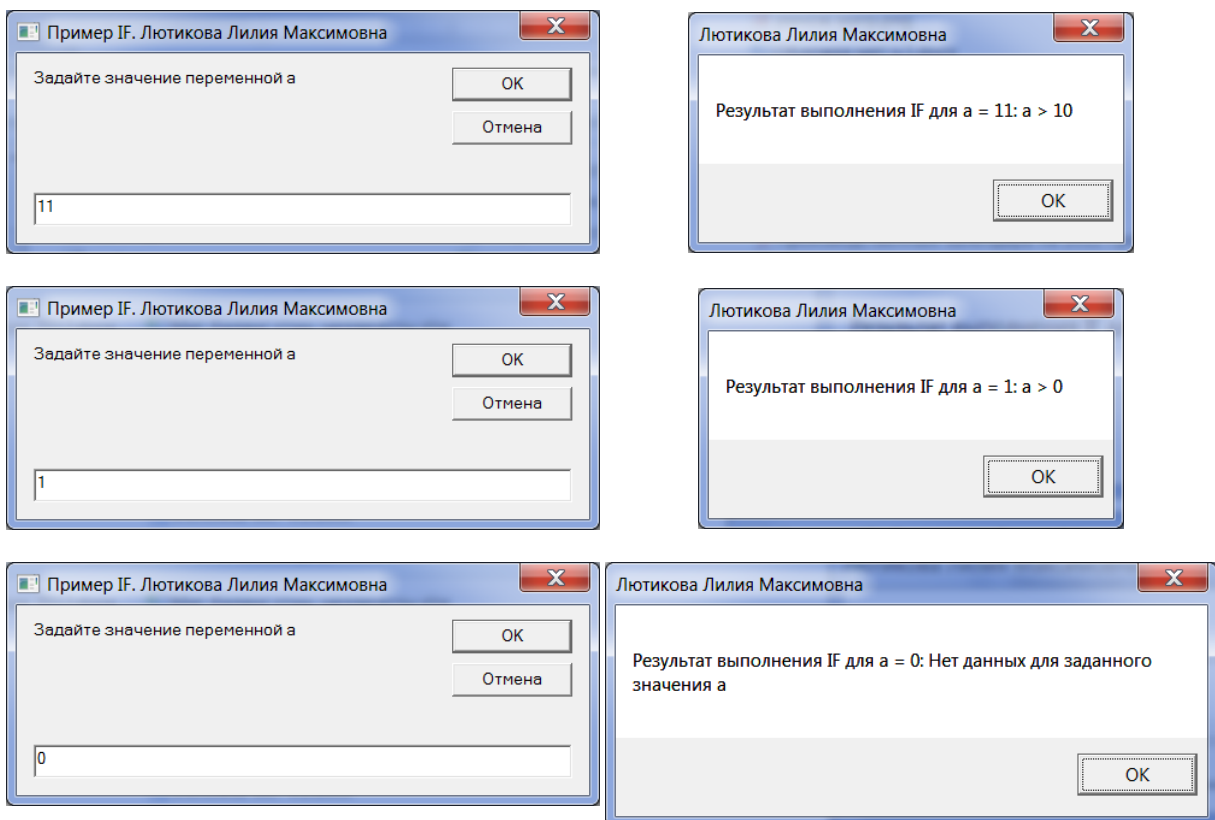


Рисунок 15.1 – Примеры работы программы с использованием блочного оператора IF

3 Оператор выбора Case

Оператор выбора Case позволяет выполнить те или иные операторы в зависимости от множества значений заданного выражения или переменной.

Синтаксис оператора выбора:

```
Select Case <тест-выражение>
  [Case <список_выр-п>
    [<операторы-п>]] . . .
  [Case Else
    [<else-операторы-п>]]
End Select
```

где:

тест-выражение – любое числовое или строковое выражение;

список_выр-п – список из одного или более выражений для соответствующего **Case**;

операторы-п – один оператор или несколько, выполняющихся, если **тест-выражение** имеет то же значение, что и значение одного из выражений **списка-п**;

else-операторы-п – один оператор или несколько, выполняющихся, если **тест-выражение** не совпадает ни с одним из значений **Case**-структур.

Пример использования структуры **Case**:

```
FIO = "Иванов Иван иванович"
X = InputBox("Задайте значение переменной a", _
  "Пример Case. " & FIO)
X = Eval(X) 'преобразование строки в число
Select Case X
Case -3, -2, -1 Y = "X из списка -1, -2, -3"
Case 0 Y = "X равно 0"
Case 1, 2, 3 Y = "X из списка 1, 2, 3"
Case Else Y = "Нет данных"
End Select
MsgBox "Результат выполнения Case для X = "& X & _
  ": " & Y, 4096, FIO
```

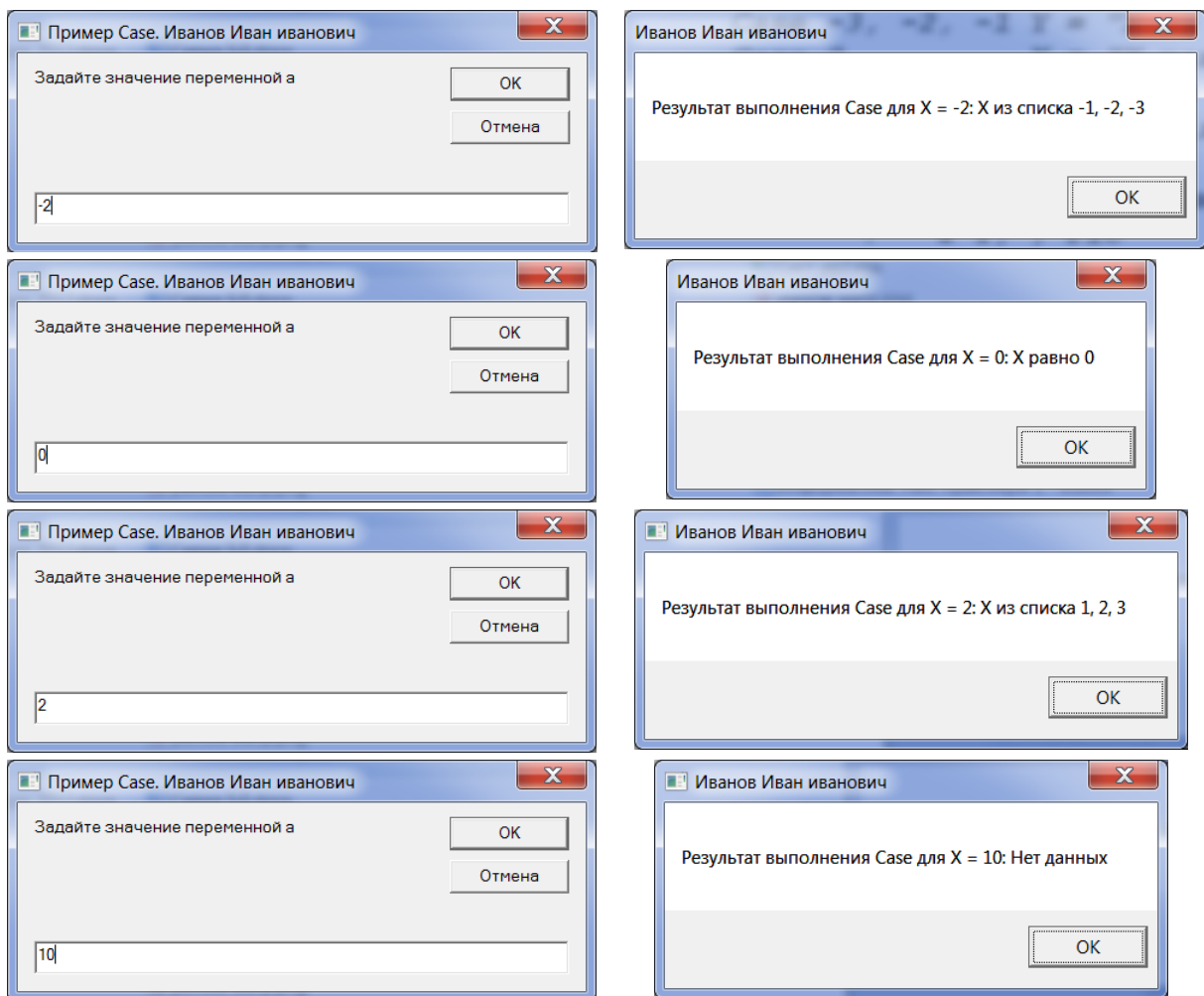


Рисунок 15.2 – Примеры работы программы с использованием структуры Case

Учебное задание

1) Для Вашего варианта таблицы 15.1 задайте в окне ввода значение переменной X с учетом заданного подтипа данных.

При вводе маленьких или больших чисел с использованием буквы **e** (например, $-1e15$) используйте преобразование подтипа строка в число с использованием функции **Eval**. При вводе даты и времени используйте функцию преобразования подтипа **Cdate**. При работе с датами учитывать, что их основные форматы **#мм/дд/гггг#** или **#мм-дд-гг#**.

Вычислите переменную Y по одному из выражений в зависимости от значения X . Значения переменных X и Y покажите в окне сообщений.

Для тех заданий, где варианты X охватывают не все возможные значения этой переменной (например, номер месяца – целое число только от 1 до 12, а не любое другое целое число), предусмотреть вывод сообщения «Нет решения для заданного значения X »

Выполнить данное задание с использованием:

- а) строчного синтаксиса оператора условного перехода,
- б) блочного синтаксиса оператора условного перехода.

Таблица 15.1 – Варианты заданий

№	Условие	Y	№	Условие	Y
1.1	$X \leq -10^{15}$	Y= «маленькое число»	1.6	X – месяц от 1 по 3	Y= «1-й квартал»
	$X > -10^{15}$ и $X < 0$	Y= «отрицательное число»		X – месяц от 4 по 6	Y= «2-й квартал»
	$X \geq 0$ и $X < 10^{15}$	Y= «положительное число»		X – месяц от 7 по 9	Y= «3-й квартал»
	$X \geq 10^{15}$	Y= «большое число»		X – месяц от 10 по 12	Y= «4-й квартал»
1.2	X – символ до «Г»	Y= 1	1.7	$X < -10^{308}$	Y= $-\infty$
	X – символ от «Г» до «Ж»	Y= 2		$X \geq -10^{308}$ и $X \leq 10^{30}$	Y= «диапазон действительных чисел»
	X – символ после «Ж»	Y= 3		$X > 10^{308}$	Y= $+\infty$
1.3	X – дата меньше 01.01.1900	Y= 19	1.8	X от 0 по 255	Y = «подтип Byte»
	X – дата от 01.01.1900 до 31.12.1999	Y= 20		X от -32768 по 32767	Y = «подтип Integer»
	X – дата начиная с 01.01.2000	Y= 21		X – целые числа другие	Y = «подтип Long»
1.4	X – время от 0 час. 00 мин. до 6 час. 00 мин.	Y= «ночь»	1.9	X – дата и время = 1.1.2010 0:0:0	Y = «С Новым годом!»
	X – время от 6 час. 01 мин. до 12 час.00 мин.	Y= «утро»		X – дата от 1 января 0000 года по 31 декабря 2099	Y = «21 век!»
	X – время от 12 час. 01 мин. до 18 час. 00 мин.	Y= «день»		X – дата от 1 января 1900 года по 31 декабря 0099 года	Y = «20 век!»

	X – время от 18 час. 01 мин. до 23 час. 59 мин.	Y= «вечер»		X – дата от 1 января 100 года по 31.12.9999	Y = «верный диапазон дат!»
1.5	X – месяц от 12 по 2	Y= «зима»	1.10	$X < 0$	Y= «X отрицательное»
	X – месяц от 3 по 5	Y= «весна»		$X \geq 0$ и $X < 10^{-15}$	Y= «X маленькое положительное число»
	X – месяц от 6 по 8	Y= «лето»		$X \geq 10^{-15}$ и $X < 1$	Y= «X меньше 1»
	X – месяц от 9 по 11	Y= «осень»		$X \geq 1$	Y= «X не меньше 1»

- 2) С использованием оператора выбора **Case** выполнить задания 1-го пункта:
 для компьютеров с № 1 по № 5 – вариант 1.5, с № 6 по № 10 – вариант 1.6.

Лабораторная работа № 16. Программирование на языке VBS. Операторы цикла Do и While

Цель работы

Освоение приемов организации циклов в программе на алгоритмическом языке.

Учебные вопросы

1. Цикл типа Do... Loop.
2. Цикл типа While ... Wend.

Краткое изложение учебного материала

Оператор цикла позволяет выполнить группу операторов несколько раз в соответствии с заданными условиями повтора.

Существует несколько видов оператора цикла:

- 1) **Do... Loop**
- 2) **While ... Wend**
- 3) **For ... Next**
- 4) **For Each ... Next.**

Данная лабораторная работа посвящена первым двум.

1 Цикл типа Do... Loop

Синтаксис оператора Do...Loop следующий (здесь и далее в фигурных скобках {} приведены два возможных варианта, разделенных вертикальной чертой |, один из которых необходимо использовать):

- 1) первый вариант – проверка условия в начале цикла

```
Do [{While | Until} <условие>]
  [<операторы>]
  [Exit Do]
  [<операторы>]
Loop
```

- 2) второй вариант – проверка условия в конце цикла

```
Do
  [<операторы>]
  [Exit Do]
  [<операторы>]
Loop [{While | Until} <условие>]
```

где:

условие – логическое выражение, которое имеет значение истина (**True**) или ложь (**False**); значение условия **Null** то же, что и **False**;
для **While** (англ. **пока**) выполнение цикла продолжается, пока условие истинно,

для **Until** (англ. **до**) – выход из цикла, когда условие истинно;

операторы – один или несколько операторов, выполнение которых повторяется, пока условие после **while** истинно (**True**) или условие после **Until** ложно (**False**);

Exit Do – может использоваться, как альтернативный выход из цикла (на следующую строку программы после **Loop**); любое количество **Exit Do** может быть помещено внутри цикла. Обычно эта команда используется с вычисляемым логическим выражением оператора **If...Then**.

Пример использования операторов циклов **Do While...Loop** и **Do Until...Loop** – вычисление числа π по формуле Беллара

$$\pi = \frac{1}{2^6} \sum_{n=0}^{\infty} \frac{(-1)^n}{1024^n} \left(\frac{256}{10n+1} + \frac{1}{10n+9} - \frac{64}{10n+3} - \frac{32}{4n+1} - \frac{4}{10n+5} - \frac{4}{10n+7} - \frac{1}{4n+3} \right)$$

```
eps = 1e-3      'точность расчета суммы на стадии отладки
n = 0
a = -1^n/1024^n*(256/(10*n+1)+1/(10*n+9)-64/(10*n+3)-_
    32/(4*n+1)-4/(10*n+5)-4/(10*n+7)-1/(4*n+3))
s = a
t1 = Time
Do While Abs(a) > eps      'цикл выполняется, пока |a| > eps
    'или Do Until Abs(a) <= eps, что аналогично предыдущему
    n = n + 1
    a = -1^n/1024^n*(256/(10*n+1)+1/(10*n+9)-64/(10*n+3)-_
        32/(4*n+1)-4/(10*n+5)-4/(10*n+7)-1/(4*n+3))
    s = s + a
Loop
t2 = Time
dt = FormatDateTime(t2-t1)
pi = 3.1415926535897932384626433832795 'копия из Калькулятора
r_pi = 1/2^6*s      'расчетное значение pi
MsgBox "Eps = " & eps & vbCrLf & "Ист. Pi = " & pi _
    & vbCrLf & "Расч. Pi = " & r_pi & vbCrLf & "n = " & n _
    & vbCrLf & "dt = " & dt, 4096, "Иванов И. И."
```

Результаты работы программы при $\text{eps} = 1e-3$ и при $\text{eps} = 1e-15$ показаны на рисунке 16.1.

Внимание! При неверном написании условий окончания цикла программа может зациклиться (будет работать бесконечно долго). Чтобы прекратить выполнение зациклившейся программы, необходимо открыть средство Windows *Диспетчер Задач* (Task Manager) с использованием сочетания клавиш Ctrl+Alt+Delete или, щелкнув правой кнопкой мыши на пустом месте панели задач, и выбрав в контекстном меню это средство, далее в разделе процессов найти и выделить **wscript.exe** и нажать кнопку **Завершить процесс** (End Process).

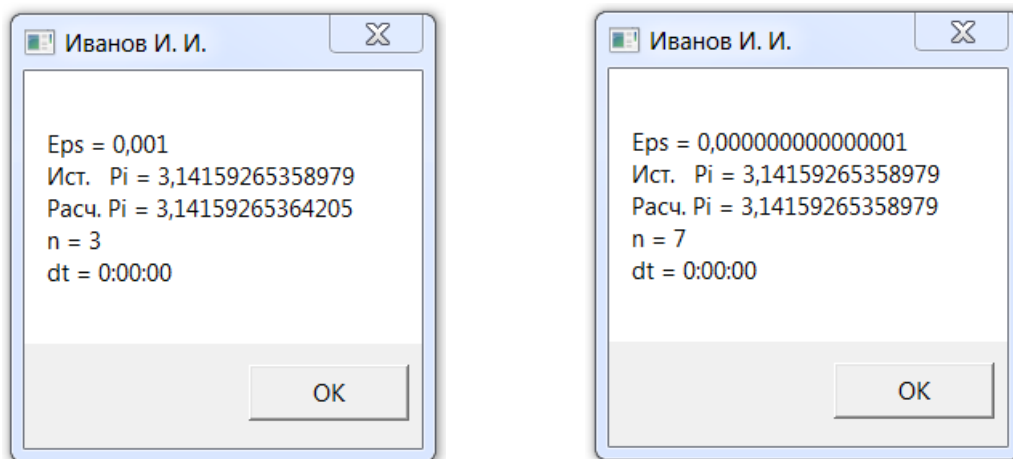


Рисунок 16.1 – Результаты расчета числа Pi по формуле Беллара

2 Цикл типа *While ... Wend*

Синтаксис оператора цикла **While...Wend** следующий:

```
While <условие>
    [<операторы>]
Wend
```

Выполнение операторов цикла повторяется, пока **<условие>** истинно (True).

Пример использования оператора **While...Wend** для расчета значения **y = arctg(x)** с использованием итерационного ряда:

$$y = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot x^{2n+1}}{(2n+1)}$$

```
eps = 1e-28
'x = 0.9      'вариант для предварительного тестирования
x = 0.999999
n = 0
a = -1^n*x^(2*n+1)/(2*n+1)
s = a
t1=Time
```

```

While Abs(a) > eps
n = n + 1
  a = -1^n*x^(2*n+1)/(2*n+1)
  s = s + a
Wend
t2=Time
dt = FormatDateTime(t2-t1)
MsgBox "Истинное значение " & Atn(x) & vbLf & _
      "Расчетное значение "& s & vbLf & _
      "Продолжительность расчета " & dt & vbLf & _
      " x = " & x & " n = " & n , 4096, _
      "Y = arcctg(x). Иванов И. И."

```

Результат выполнения этой программы показан на рисунке 5.2.

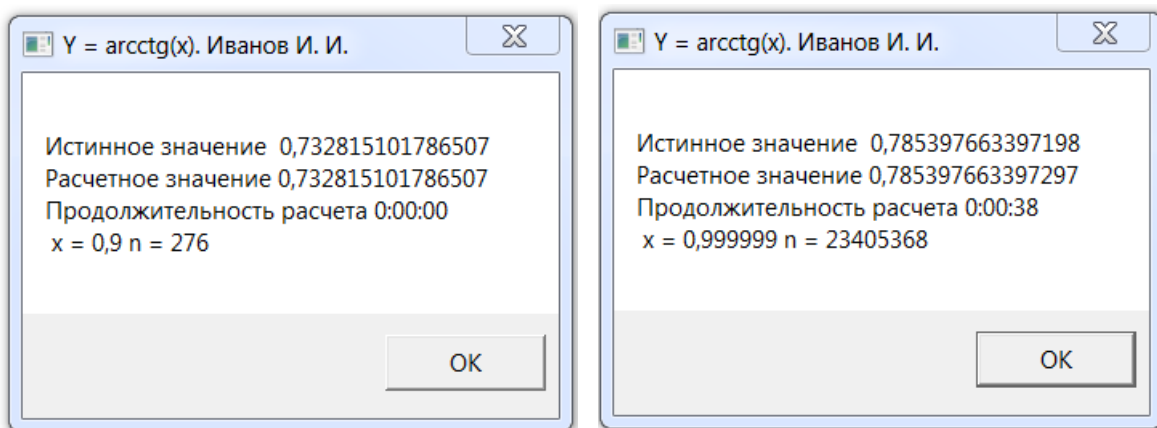


Рисунок 16.2. – Результаты расчета с циклом While... Wend

Учебное задание

Вычислите число π по итерационной формуле с номером, соответствующим номеру Вашего ПК, с абсолютной погрешностью вычисления от 10^{-5} до 10^{-16} . Найдите величину погрешности, при которой в числе π постоянными остаются 7 знаков после запятой. Определите программно время расчета для каждого варианта, покажите в окне сообщений расчетные значения π , n и продолжительность расчета для различной погрешности.

Напишите пять вариантов программы для цикла **DO** с проверкой условия в начале и в конце и для цикла **WHILE**. Расчет для максимальной точности выполнить для одного варианта, т. к. его продолжительность может составлять 15 – 30 мин.

$$1) \quad \frac{\pi}{2} = \prod_{n=1}^{\infty} \frac{4n^2}{4n^2 - 1} \qquad 2) \quad \frac{\pi}{3} = \sum_{n=0}^{\infty} (-1)^n \left(\frac{1}{6n+1} + \frac{1}{6n+5} \right)$$

$$3) \quad \frac{\pi}{4} = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} \qquad 4) \quad \frac{1}{6}\pi^2 = \sum_{n=1}^{\infty} \frac{1}{n^2}$$

$$5) \quad \frac{1}{8}\pi^2 = \sum_{n=1}^{\infty} \frac{1}{(2n-1)^2} \qquad 6) \quad \frac{\pi}{4} = \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{2n-1}$$

$$7) \quad \frac{\pi}{4} = \sum_{n=0}^{\infty} (-1)^n \left(\frac{1}{10n+1} - \frac{1}{10n+3} + \frac{1}{10n+5} - \frac{1}{10n+7} + \frac{1}{10n+9} \right)$$

$$8) \quad \frac{\pi}{2} = \prod_{n=1}^{\infty} \frac{4n^2}{(2n-1)(2n+1)}$$

$$9) \quad \pi = \sum_{n=0}^{\infty} \left(\frac{4}{8n+1} - \frac{2}{8n+4} - \frac{1}{8n+5} - \frac{1}{8n+6} \right) \left(\frac{1}{16} \right)^n$$

$$10) \quad \frac{\pi\sqrt{2}}{4} = \sum_{n=1}^{\infty} \left(\frac{(-1)^{n+1}}{4n-1} + \frac{(-1)^{n+1}}{4n-3} \right)$$

Лабораторная работа № 17. Программирование на языке VBS. Операторы цикла For и For Each

Цель работы

Освоение особенностей использования цикла **For...** в программе на алгоритмическом языке.

Учебные вопросы

1. Цикл типа **For... Next**.
2. Цикл типа **For Each... Next**.

Краткое изложение учебного материала

1 Цикл типа For... Next

Синтаксис оператора цикла **For...Next** следующий:

```
For <счетчик> = <нач.знач.> To <кон.знач.> [Step <шаг>]  
    [<операторы>]  
    [Exit For]  
    [<операторы>]  
Next
```

где:

счетчик – числовая переменная, используемая как счетчик (параметр) цикла; может быть положительной или отрицательной величиной

нач.знач. – начальное значение счетчика;

кон.знач. – конечное значение счетчика;

шаг – шаг изменения счетчика; на данную величину автоматически изменяется **счетчик** после каждого выполнения операторов цикла; если **шаг** не указан, значит, он равен 1;

операторы – выполняются повторно столько раз, сколько определено значениями, заданными для **счетчика**: один раз, много раз или ни одного;

Exit For – может использоваться, как альтернативный выход из цикла; обычно используется с проверкой условия выхода в операторе **If...Then**; выход выполняется на строку программы, следующую за **Next**.

Пример использования цикла **For...Next** (результат исполнения программы показан на рисунке 17.1):

```
n = 17  
Redim x(n)  
Randomize
```

```

For i = 1 to n                                'генерация n случайных чисел
    x(i) = 10 - 20*Rnd                          'в диапазоне от -10.000 до 9.999
Next
s = 0
For i = 1 to n
    s = s + x(i)                                'расчет суммы всех элементов массива X
Next
Ms = ""                                        'Ms - строка для показа всех элементов массива X
For i = 1 to n
    ms = ms & x(i) & vbLf
Next
ms = ms & "Сумма всех x = " & S
MsgBox ms, 4096, " For ... Next"

```

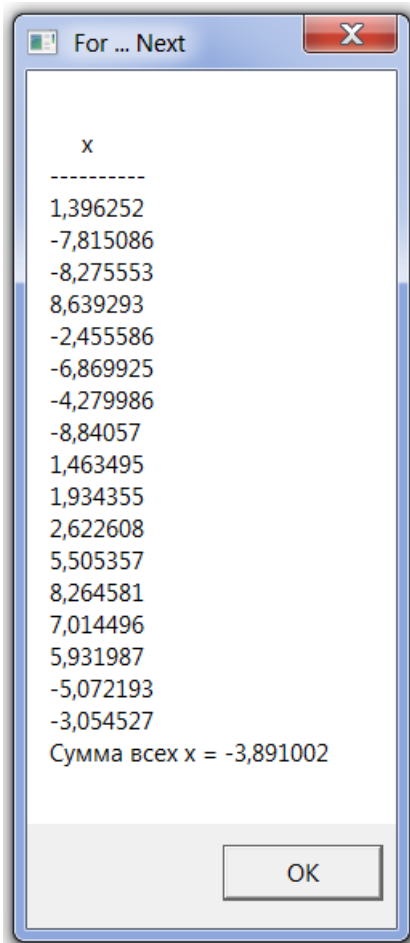


Рисунок 17.1 – **For...Next**

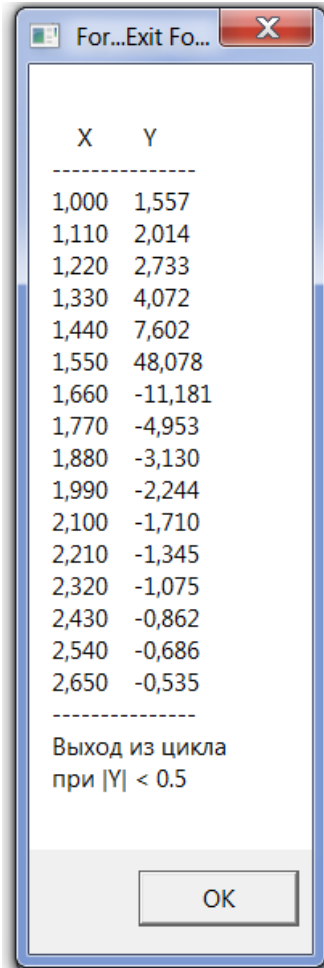


Рисунок 17.2 – **For...Exit For ...Next**

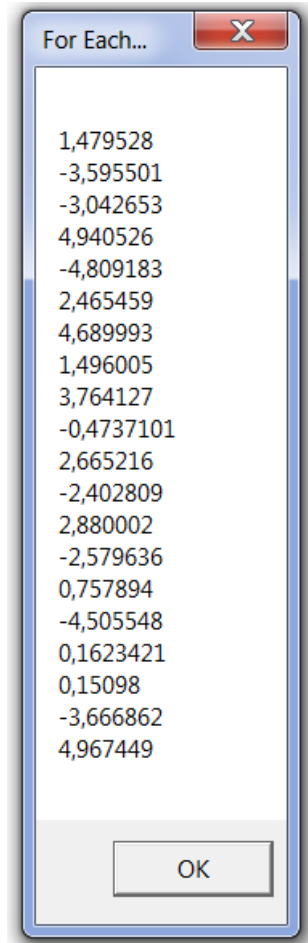


Рисунок 17.3 – Цикл **For Each**

Пример программы с альтернативным выходом (результат исполнения см. на рисунке 17.2):

```
S = "      X          Y" & vbLF
S = S & "-----" & vbLF
For X = 1 to 5 step 0.11
  Y = FormatNumber(Tan(X),3)
  If Abs(Y) < 0.5 Then
    s = s & "-----" & vbLf _
      & "Выход из цикла" & vbLf & "при |Y| < 0.5"
  Exit For
End If
  s = s & FormatNumber(X,3) & "      " & Y & vbLf
Next
MsgBox S,4096," For...Exit For...Next"
```

2 Цикл типа **For Each... Next**

Синтаксис оператора цикла **For Each...Next** следующий:

```
For Each <элемент> In <группа>
  [операторы]
  [Exit For]
  [операторы]
Next [<элемент>]
```

где:

элемент – переменная, которая используется для перебора всех элементов коллекции или массива;

группа – имя коллекции объектов или массива.

Пример программы с использованием оператора **For Each** для работы с массивом:

```
N=19
ReDim x(n)
Randomize
For i = 0 to n      'генерация n+1 случайных чисел
  x(i) = 5 - 10*Rnd 'в диапазоне от -5.000 до 4.999
Next
S = ""
For Each iks in X
  S = S & iks & vbLf
Next
MsgBox S, 4096," For Each... "
```

В данной программе цикл **For Each** использует все значения массива **X** для формирования строки **S** (результат исполнения см. на рисунке 17.3).

Учебное задание

С использованием оператора цикла **FOR** и функции **RND** сгенерировать массивы из n действительных чисел, необходимые для вычисления по заданной ниже формуле. Для расчета по заданной формуле использовать оператор **For Each**. Программу выполнить несколько раз для различных значений n . Исходные данные и результаты показать в окне сообщений.

1)	$s = \sum_{i=1}^n x_i \cdot x_j$	2)	$y_i = x_i \sum_{i=1}^n x_i$
3)	$c_i = a_i \cdot b_i$	4)	$z_i = \frac{(n-i) \cdot x_i}{y_i}$
5)	$z_i = \frac{x_i}{x_{\max} - x_{\min}}$ x_{\max}, x_{\min} - максимальное и минимальное значения x	6)	$y_i = \frac{x_i}{x_{cp}}$ $x_{cp} = \sum_{i=1}^n x_i / n$
7)	$s = \sum_{i=1}^n x_i / i$	8)	$y_i = \frac{x_i}{\sum_{i=1}^n x_i}$
9)	$y_i = ax_i^2 + bx_i + c$	10)	$s = \sum_{i=1}^{n-1} \frac{x_i}{x_{i+1}}$














Приложение 1. Горячие клавиши Windows 7

Операции с текстом и файлами

- **Ctrl + A** – Выделить всё.
- **Ctrl + C** – Копировать.
- **Ctrl + X** – Вырезать.
- **Ctrl + V** – Вставить.
- **Ctrl + Z** – Отменить.
- **Ctrl + Y** – Повторить действие.
- **Ctrl + B** – Сделать жирным (только текст).
- **Shift + Delete** – Удалить файл без перемещения в корзину.
- **CTRL+S** – Сохранение текущего файла или документа (работает в большинстве программ)
- **ALT+F4** – Закрытие текущего элемента или выход из активной программы
- **Ctrl** + вращение колесика мыши – меняется размер текста или иконок в Проводнике и на Рабочем столе.
- **F5** – в Проводнике – обновить окно

Прочие операции

- **Alt + Табуляция** – Переключение между окнами.
- **Shift + Ctrl + N** – Создать новую папку.
- **Ctrl + Shift + Esc** – Открытие диспетчера задач.
- **Alt + F4** – Закрыть активное окно.
- **F11** – Развернуть окно на весь монитор (без заголовка окна и панели задач, работает для некоторых окон, в том числе для Проводника и Интернет браузеров)
- **F1** – Справка
- **F5** – Обновить окно.
- **Esc** – Отмена текущего задания.

-  – Открытие меню «Пуск»
-  + **F1** – Вызов справки Windows
-  + **B** – Перенести курсор в область уведомлений.
-  + **D** – Показать рабочий стол (свернуть все окна).
-  + **E** – Открыть окно Мой компьютер.
-  + **F** – Открыть окно поиска.
-  + **G** – Показать гаджеты поверх окон.
-  + **L** – Заблокировать компьютер.
-  + **M** – Свернуть все окна.
-  + **P** – Диалог управления проектором.
-  + **R** – Открыть окно **Выполнить...**
-  + **T** – По очереди переводим фокус последовательно по иконкам **Панели задач**.
-  + **U** – Окно Центр специальных возможностей.

- **Windows** + X – Окно **Центр мобильности** (ноутбуки и нетбуки).
- **Windows** + Табуляция – Вызов **Flip 3D**.
- **Windows** + Пробел – Просмотр рабочего стола (**Aero Peek**, сделать все окна прозрачными).
- **Windows** + Стрелка - Управление расположением активного окна (Вверх - максимизировать, вниз - свернуть, влево - прикрепить к левому краю, вправо - прикрепить к правому краю).
- **Windows** + Pause - Свойства системы.
- **Windows** + Home - Свернуть все окна, кроме активного (тоже самое, что и **Aero Shake**(потрясти окно)).
- **Windows** + Shift + Вверх - Развернуть по максимальному окно по вертикали.
- Shift + **Windows** + Влево/вправо - Перенаправить окно на соседний монитор.
- **Windows** + Номер иконки на Панели задач - Выделить это окно, а остальные сделать прозрачными (**Aero Peek**).

Литература

- 1) Грошев А. С. Информатика: Учебник для вузов – Архангельск, Арханг. гос. техн. ун-т, 2010. – 470 с.