

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Минцаев Маршал Шаварши

Должность: Ректор

Дата подписания: 04.10.2023 17:17:04

Уникальный программный ключ:

236bcc35c296f119dbaa9dc228568218b52dbc07974a886865a3025f9a4504cc

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ АКАДЕМИКА М.Д. МИЛЛИОНЩИКОВА

Кафедра «Информационные технологии»

А.А. Албакова

**Методические указания к выполнению лабораторных работ
по дисциплине «Операционные системы»**

для студентов, обучающихся по направлению подготовки
09.03.02 Информационные системы и технологии,
уровень подготовки – бакалавр

Грозный, 2022

Составители:

Ассистент кафедры
«Информационные технологии»

Албакова Аминат Ахмедовна

Рецензент:

Э.Д. Алисултанова, доктор педагогических наук, кандидат физико-математических наук, профессор, директор Института прикладных информационных технологий, заведующая кафедрой «Информатика и вычислительная техника»

В данных методических указаниях изложены краткие сведения из теории по Операционным системам, цель лабораторной работы, теоретический материал, задание и контрольные вопросы, а также содержится пример оформления отчета о выполненных заданиях.

Методические указания предназначены для бакалавров по направлению подготовки 09.03.02 Информационные системы и технологии Института прикладных информационных технологий.

Методические рекомендации рассмотрены и утверждены на заседании кафедры «Информационные технологии»:

Протокол № ____ от _____ 2022г.

Рекомендовано к изданию редакционно-издательским советом ГГНТУ.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
Краткие сведения из теории	5
Лабораторная работа 1. Основы командной строки	8
Лабораторная работа 2. Файловая структура	16
Лабораторная работа 3. Пакетные файлы	23
Лабораторная работа 4. Элементы программирования в cmd	26
Лабораторная работа 5. Планирование задач	30
Лабораторная работа 6. Разграничение прав пользователей	37
Лабораторная работа 7. Реализация политики безопасности.....	41
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	44
Приложение 1. Образец оформления отчета	45

ВВЕДЕНИЕ

Методические указания содержит лабораторные работы по дисциплине «Операционные системы», отвечающие требованиям программы дисциплины по направлению подготовки 09.03.02 Информационные системы и технологии.

Цель лабораторных работ - формирование у студентов понимания состава, функций и принципов работы операционных систем и выработка умений их использования в профессиональной деятельности.

Тематика лабораторных работ охватывает изучаемые разделы дисциплины и способствует закреплению и лучшему усвоению теоретического материала, приобретению практических навыков работы в определенной ОС.

В работах содержатся пояснения теоретического и практического содержания, что способствует выполнению работы студентами с разным уровнем подготовки.

КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ

Операционная система (ОС) – комплекс управляющих и обрабатывающих программ, которые выступают как интерфейс между устройствами вычислительной системы и прикладными программами и предназначены для управления устройствами и вычислительными процессами, эффективного распределения вычислительных ресурсов между вычислительными процессами и организации надежных вычислений.

Одной из основных задач операционной системы является предоставление пользователю удобного способа работы с данными, хранящимися на дисках. Для этого ОС подменяет физическую структуру хранящихся данных некоторой удобной для пользователя логической моделью. Логическая модель файловой системы в ОС семейства Windows материализуется в виде дерева папок (каталогов), выводимого на экран такими утилитами, как Windows Explorer, Windows Commander или Far Manager, и символьных составных имен файлов. Базовым элементом этой модели является файл, который, как и файловая система в целом, может характеризоваться и логической, и физической структурой.

Таким образом, **файловая система** представляет собой комплекс системных программных средств, реализующих различные операции с файлами, такие как создание, уничтожение, чтение, запись, именование и поиск файлов. Под файловой системой также понимают набор всех файлов и служебных структур данных, хранящихся на внешнем носителе.

Объекты файловой системы ОС Windows:

1. Диск – носитель информации, который может представлять собой физическое устройство или часть пространства носителя, называемого логическим диском. Он именуется одной буквой латинского алфавита с добавлением после нее знака «:» – двоеточия, именование происходит последовательным присвоением латинских букв по алфавиту (например C:).

2. Файл – это именованная область внешней памяти, в которую можно записывать и из которой можно считывать данные (информацию). Данные, с которыми работает пользователь, различны, поэтому ОС использует разные средства для их обработки. Чтобы ОС правильно «понимала», что следует выполнить с разного вида данными, их разделяют на несколько типов.

В файловых системах NTFS и FAT32, входящих в состав операционной системы Windows, имя файла может содержать до 255 символов (буквы и цифры латинского и русского алфавита), кроме символов \ / : * ? " | < > и имен CON, NUL, PRN, AUX, COM1÷9, LPT1÷9.

Вторая часть имени – это **тип (расширение, формат)**, который начинается с символа точки (.) и может включать от 1 до 4 символов. Существуют зарегистрированные типы, по которым ОС связывает файл с приложением (программой), которой он будет открыт. (табл.1)

Таблица 1. – Типы файлов и простые имена

Расширение	Тип файла	Пример имени
.exe .com	Исполнимые программы	calc.exe command.com
.hlp	Справка	windows.hlp
.txt	Простые текстовые документы	text.txt

.doc .docx	Текстовые документы Microsoft Word	Резюме.doc Реферат.docx
.xls .xlsx	Электронные таблицы Microsoft Excel	Вычисления.xls Отчет 2019.xlsx
.ppt .pptx	Презентации Microsoft PowerPoint	Исследование.ppt New version.pptx
.htm .html	Страницы Интернета	book.htm Новости.html
.zip	Архив WinZIP	Курсовой проект.zip
.rar	Архив WinRAR	ВРК.rar
.bmp .jpg .tif .gif	Графические файлы	Веранда.bmp Фотостудия.jpg My Card.tif Рисунок1.gif
.mp3 .wma	Аудиофайлы	Хорошая песня.mp3 Вальс.wma
.mp4 .avi	Видеофайлы	Кино.mp4 Ролик2.avi

В иерархических файловых системах разным файлам разрешено иметь одинаковые простые символьные имена при условии, что они принадлежат разным папкам. Для однозначной идентификации файла в таких системах используется так называемое **полное имя** - цепочку, состоящую из имени диска, простых символьных имен всех папок (каталогов), через которую проходит путь от корня до данного файла и простого имени файла. Таким образом, полное имя является составным, в котором имена папок отделены друг от друга обратным слешем «\».

Например: **d:\папка1\папка2\пример1.txt**.

3. Папка (каталог) – это особый тип файлов, который содержит системную справочную информацию о наборе файлов, сгруппированных пользователями по какому-либо признаку. Папки устанавливают соответствие между именами файлов и их характеристиками, используемыми файловой системой для управления файлами, такими как информация о типе файла и расположении его на диске, правах доступа к файлу, его размере и дате его создания или модификации.

4. Ярлык – это файл, содержащий ссылку на объект, программу или команду, а также дополнительную информацию. Ярлык служит для быстрого доступа к нужному объекту. Его можно размещать в любом месте, выбранном пользователем.

Современные файловые системы имеют **иерархическую (древовидную) структуру**, упрощающую именование файлов и их поиск. В иерархической структуре уровни создаются за счет того, что папка более низкого уровня может входить в папку более высокого уровня.

Граф, описывающий иерархию папок, может быть деревом или сетью. Папки образуют дерево, если файлу разрешено входить только в одну папку (рис. 1), и сеть, если файл может входить сразу в несколько папок (рис. 2). Например, в Windows папки образуют древовидную структуру, а в Unix – сетевую. В древовидной структуре каждый

файл является «листом». Папка самого верхнего уровня называется **корневой папкой**, или **корнем** (root).

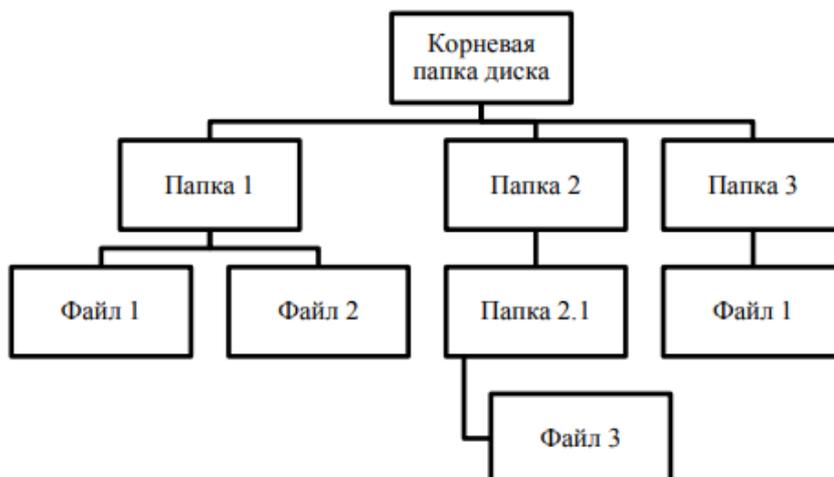


Рис. 1. Иерархическая многоуровневая файловая система (Windows)



Рис. 2. Сетевая файловая система (Unix)

ЛАБОРАТОРНАЯ РАБОТА 1. ОСНОВЫ КОМАНДНОЙ СТРОКИ

Цель работы: практическое изучение возможностей командной строки операционной системы Microsoft Windows на примере часто используемых команд.

1. ТЕОРЕТИЧЕСКИЙ МАТЕРИАЛ

1.1. Знакомство с интерфейсом

Командная строка представляет собой программное средство ввода команд пользователем и получения результатов их выполнения на экране. В современных операционных системах семейства Windows, командную строку обеспечивает стандартное приложение **cmd.exe**, так же называемое командным процессором, интерпретатором команд и консолью. Приложение "Командная строка" предоставляет пользователю текстовый интерфейс для ввода команд и получения результатов их выполнения. Фактически, командная строка является программным эмулятором классической консоли первых компьютерных систем, представляющей собой терминал с клавиатурой, используемой оператором в качестве средства общения с компьютером. Как и во времена первых компьютеров, командная строка поддерживает стандартное устройство ввода – клавиатуру, и стандартное устройство вывода – дисплей. Пользователь вводит команды с клавиатуры и получает результаты их выполнения на экране дисплея.

Для запуска командной строки необходимо выполнить одно из действий:

- **Пуск - Выполнить** (или клавиши Win+R) введите **cmd** и нажмите клавишу Enter;
- **Пуск - Все программы - Стандартные - Командная строка**;
- **Пуск - Поиск - Командная строка**.

В независимости от использованного способа запуска у вас откроется окно командной строки, которое выглядит довольно непривычно. (рис.3)

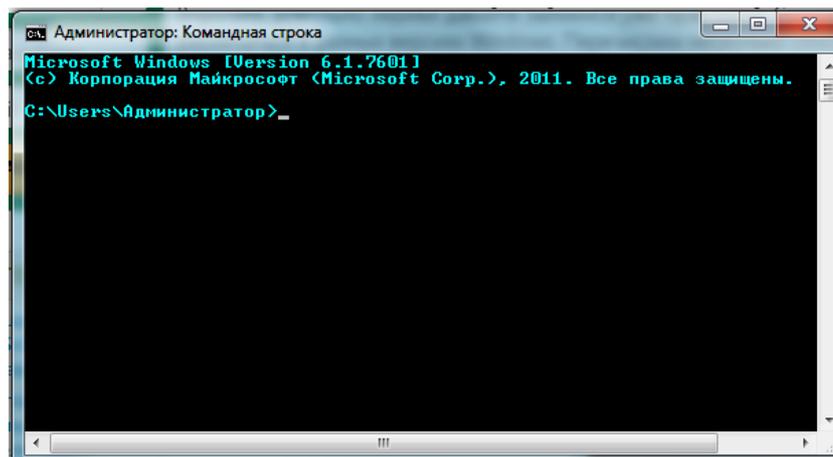


Рис.3. Окно командной строки

Стандартное окно командной строки – это окно с символами белого цвета на черном фоне. Параметры шрифтов, цвет и фон символов, использование буфера обмена и прочие свойства командной строки можно изменить в соответствии со своими предпочтениями.

Часть настроек можно выполнить, изменив свойства самого приложения, свойства ярлыка, с помощью которого запускается приложение командной строки, а часть - редактированием реестра Windows.

Настройка командной строки через изменение свойств приложения (ярлыка для запуска приложения):

1. Откройте окно **Командная строка**
2. Щелкните указателем мышки левый верхний угол окна и в открывшемся меню выберите пункт **Свойства** (рис.4)

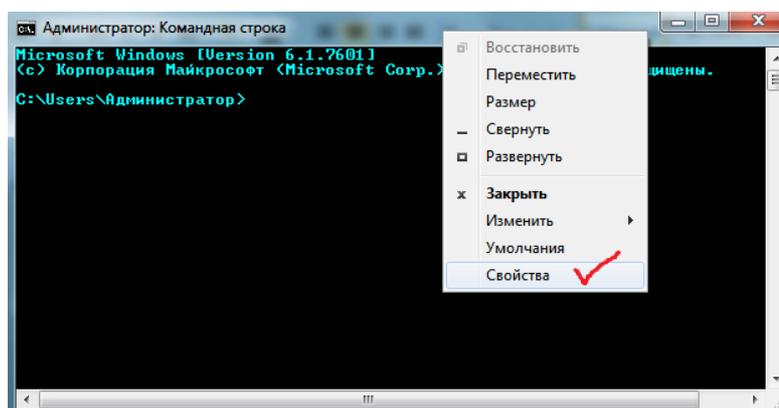


Рис.4. Свойства командной строки

3. Вкладка **Общие** (рис.5)

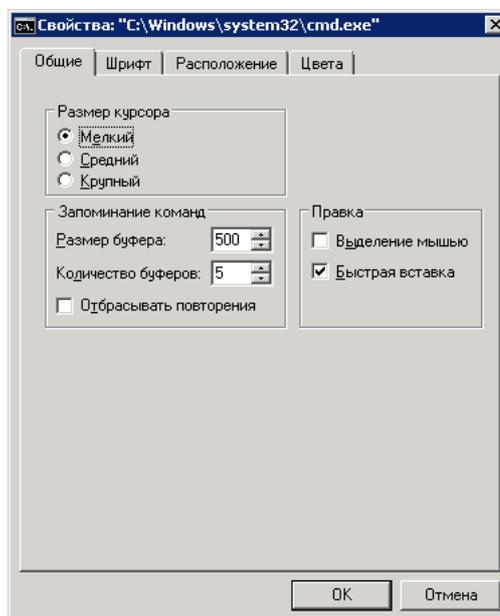


Рис.5. Вкладка «Общие»

В области **Запоминание команд** можно изменить размер буфера под историю команд. Если есть необходимость, выберите или введите новое значение (до **999** в поле **Размер буфера**, а затем выберите или введите число (до **5**) в поле **Количество буферов**. Увеличение размера буфера под запоминание команд может пригодиться в тех случаях, когда в течении сессии приходится вводить большое количество текста. Клавиши **Стрелка вверх** и **Стрелка вниз** позволяют вызывать ранее набранные и сохраненные в

буфере команды. Для того, чтобы в буфере не запоминались одинаковые команды, можно включить флажок **Отбрасывать повторения**. Значения размера буфера и количество буферов могут обличаться для разных версий Windows.

В области **Правка** установите флажки **Выделение мышью** и **Быстрая вставка**. Флажок **Выделение мышью** позволяет копировать и вставлять данные в окне командной строки с использованием мышки. Чтобы скопировать, выделите левой кнопкой текст в окне командной строки и нажмите правую кнопку мыши. Чтобы выполнить вставку в командную строку или текстовый файл, щелкните правой кнопкой мыши. Можно вместо выделения мышью, использовать стандартный режим, вызываемый через контекстное меню правой кнопкой мыши. Выбирается пункт **Пометить**, выделяется копируемый текст и нажимается правая кнопка мыши, которая используется и для копирования и для вставки. Стандартные комбинации клавиш для копирования и вставки в командной строке не работают, за исключением Windows 10 / Sever 2016.

4. На вкладке **Шрифт** можно изменить вид окна, размер шрифта

5. На вкладке **Расположение** можно изменить размер буфера экрана, а также размер и положение окна

6. Вкладка **Цвет** позволяет изменить цвет текста на экране, цвет фона экрана, цвет текста всплывающего окна, а также цвет фона всплывающего окна.

1.2. Технология работы с командной строкой

После запуска командной строки на экран дисплея выводится так называемое **приглашение** операционной системы для **ввода команд** (рис.6), который имеет обычно следующий вид:

имя текущего диска: \ имя текущего каталога > _

где > – системный символ приглашения;

_ – курсор.

Например,

C:\Users > _

где *C:* – имя текущего диска, *Users* – имя текущего каталога.

Если текущий каталог является корневым, то имя текущего каталога опускается, так как оно совпадает с именем текущего диска. Приглашение в этом случае выглядит следующим образом:

C:\> _



Рис.6. Пример приглашения к вводу команд

Справа от приглашения командной строки пользователь вводит с клавиатуры команду. Она набирается латинскими буквами и выполняется нажатием клавиши Enter. Синтаксис или общий формат команды имеет следующий вид:

имя команды [аргументы] [/параметры]

где ***аргументы*** – это объекты, на которые действует команда;
параметры – идентификаторы режимов исполнения команды (они разделяются наклонной чертой "/").

Детальное описание каждой команды не входит в объем настоящих методических указаний, однако список команд можно получить в окне командной оболочки, выполнив команду ***help***. Также по любой команде windows может быть выведена на экран справка. Для этого нужно в командной строке набрать ***имя команды*** и ***/?***, например (рис.7):

ver /?

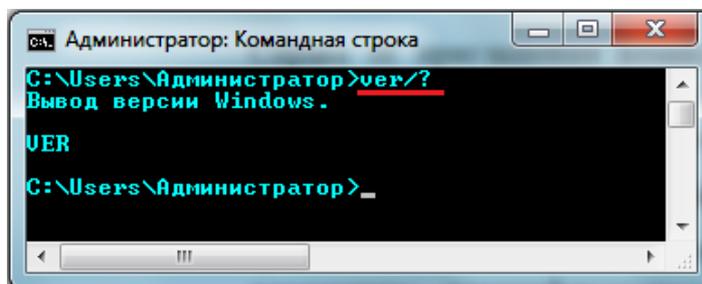


Рис.7. Вывод справки команды *ver*

Основные команды условно делятся на четыре группы:

1. Команды общего назначения
2. Команды работы с дисками
3. Команды работы с каталогами
4. Команды работы с файлами

Наиболее часто используемыми **командами группы общего назначения** являются:

- ***cls*** – очистка экрана;
- ***title*** – назначение заголовка окна для текущего сеанса интерпретатора командных строк CMD.EXE (рис.8);
- ***ver*** – вывод установленной в компьютере версии операционной системы;
- ***systeminfo*** – вывод информации о конфигурациях системы;
- ***date*** – вывод и установка текущей даты;
- ***time*** – вывод и установка текущего времени;
- ***help*** – вывод справочных данных по командам cmd;
- ***exit*** – выход из командной строки.

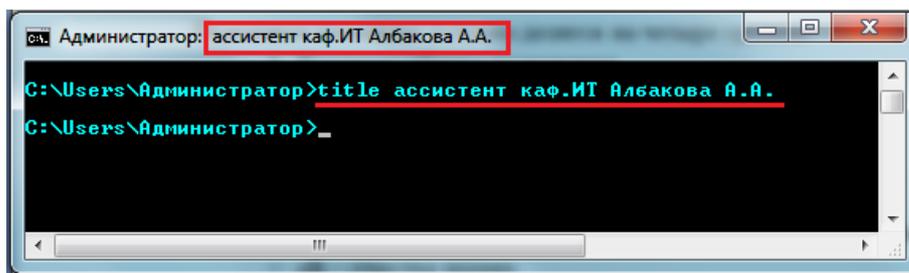


Рис. 8. Использование команды *title*

Для **смены текущего диска** необходимо набрать имя диска, который должен стать текущим, например, при переходе на диск *D:*: следует ввести команду (рис.9):

D:

После выполнения этой команды появится приглашение с текущим диском *D:*, например,

D:\>_

Т.е. формат данной команды выглядит следующим образом:

Имя_диска:

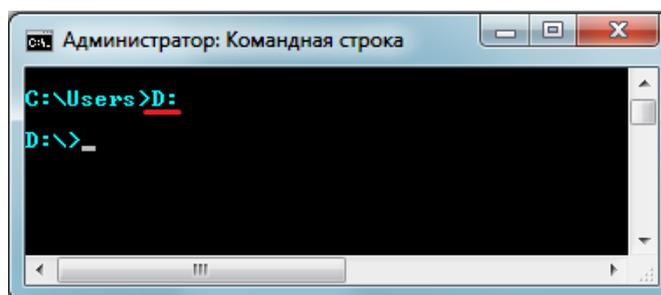


Рис.9. Смена текущего диска

Для вызова в командную строку **ранее введенных команд** и управления списком команд можно использовать следующие клавиши:

↑ – вызов предыдущей команды;

↓ – вызов следующей команды;

F7 – вывод на экран списка вводимых ранее команд;

F9 – копирование в командную строку одной из команд выведенного списка (**F7**) путем выбора ее номера;

Alt+F7 – очистка списка команд.

По умолчанию команды выводят результаты своего исполнения на экран монитора, и принимают входные данные с клавиатуры. Но это соглашение можно изменить и для этого служат **операторы перенаправления**. С использованием таких операторов можно направить вывод результатов работы команды в заданный файл или использовать эти результаты в качестве входных параметров для выполнения другой команды. Это позволяет организовывать эффективную коммуникацию между процессами при минимальных затратах времени и ресурсов. В таблице 2 описаны операторы перенаправления потоков ввода и вывода команд.

Таблица 2. – Операторы перенаправления ввода/вывода команд

Оператор перенаправления	Описание
>	Записывает данные на выходе команды вместо окна командной строки в файл или на устройство, например, на принтер.
<	Читает поток входных данных команды из файла, а не с клавиатуры.
>>	Добавляет выходные данные команды в конец файла, не удаляя при этом существующей информации из файла.
>&	Считывает данные на выходе одного дескриптора как входные данные для другого дескриптора.
<&	Считывает входные данные одного дескриптора как выходные данные другого дескриптора.
	Считывает выходные данные одной команды и записывает их на вход другой команды. Этот оператор также известен как оператор конвейеризации.

Пример команды с использованием операторов перенаправления (рис.10):

help>example.txt

где *help* – команда;

> – оператор перенаправления;

example.txt – текстовый документ куда перенаправляется вывод справочных данных по командам cmd.

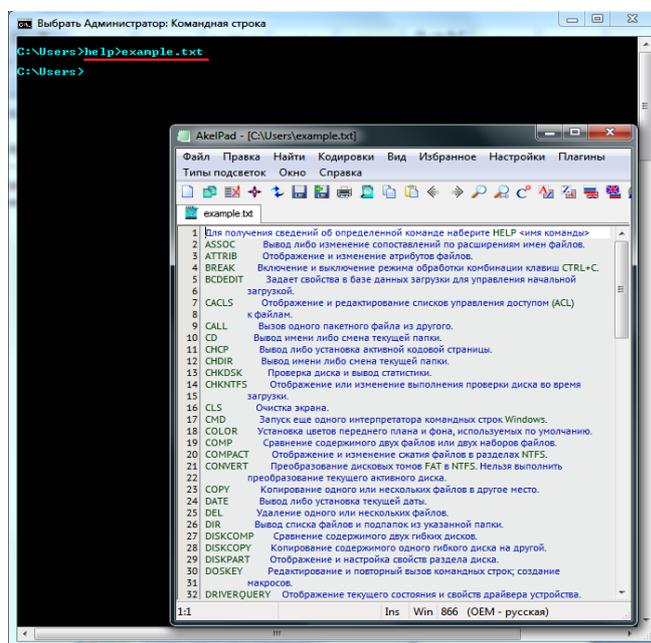


Рис.10. Пример применения оператора перенаправления ввода/вывода команд

Возможность сортировать, просматривать и отбирать часть выходной информации других команд дают **команды-фильтры**. Фильтры делят, переупорядочивают или выделяют отдельные части из проходящей через них информации. В приведенной ниже таблице 3 перечислены команды-фильтры, доступные в ОС Windows.

Таблица 3. – Команды-фильтры

Команда	Описание
more	Отображает содержимое файла или вывода команды в одном окне командной строки за раз.
find	Поиск указанных символов в файлах или выводе команды
findstr	Поиск образцов текста в файлах с использованием регулярных выражений
sort	Сортировка файлов или вывода команды по алфавиту

Зачастую решение некоторой практической задачи предполагает выполнение нескольких команд. Windows предоставляет возможность объединять несколько команд в одной строке, допуская их последовательное выполнение, а также разрешая обуславливать выполнение одних команд результатами выполнения других (табл. 4).

Таблица 4. – Операторы условного выполнения команд

Оператор	Синтаксис	Описание
&	<i>команда1 & команда2</i>	Используется для разделения нескольких команд в одной командной строке. Сначала выполняется первая команда, затем вторая.
&&	<i>команда1 && команда2</i>	Запускает команду, стоящую за символом &&, только если команда, стоящая перед этим символом была выполнена успешно.
 	<i>команда1 команда2</i>	Запускает команду, стоящую за символом , только если команда, стоящая перед символом была выполнена с ошибкой.
()	<i>(команда1 & команда2)</i>	Используется для группировки или вложения команд.

Пример команды с использованием операторов условного выполнения команд (рис.11):

ver & pause & help

где *ver* и *help* – команды;

& – оператор условного выполнения команд;

pause – команда пауза.

Другими словами эта команда означает вывести на экран установленную в компьютере версию операционной системы, дождаться нажатия клавиши (команда *pause*), вывести на экран справочник данных по командам *cmd*.

```

Администратор: Командная строка
C:\>ver & pause & help
Microsoft Windows [Version 6.1.7601]
Для продолжения нажмите любую клавишу . . .
Для получения сведений об определенной команде наберите HELP <имя команды>
ASSOC      Вывод либо изменение сопоставлений по расширениям имен файлов.
ATTRIB     Отображение и изменение атрибутов файлов.
BREAK      Включение и выключение режима обработки комбинации клавиш CTRL+C.
BCDEDIT    Задаёт свойства в базе данных загрузки для управления начальной
загрузкой.
CACLS      Отображение и редактирование списков управления доступом (ACL)
к файлам.
CALL       Вызов одного пакетного файла из другого.
CD         Вывод имени либо смена текущей папки.
CHCP       Вывод либо установка активной кодовой страницы.
CHDIR      Вывод имени либо смена текущей папки.
CHKDSK     Проверка диска и вывод статистики.
CHKNTFS    Отображение или изменение выполнения проверки диска во время
загрузки.
CLS        Очистка экрана.
CMD        Запуск еще одного интерпретатора командных строк Windows.
COLOR      Установка цветов переднего плана и фона, используемых по умолчанию.
COMP       Сравнение содержимого двух файлов или двух наборов файлов.
COMPACT    Отображение и изменение сжатия файлов в разделах NTFS.
CONVERT    Преобразование дисковых томов FAT в NTFS. Нельзя выполнить
преобразование текущего активного диска.
COPY       Копирование одного или нескольких файлов в другое место.
DATE       Вывод либо установка текущей даты.
DEL        Удаление одного или нескольких файлов.
DIR        Вывод списка файлов и подпапок из указанной папки.
DISKCOMP   Сравнение содержимого двух гибких дисков.

```

Рис.11. Пример применения оператора условного выполнения команд

2. ЗАДАНИЕ

1. Запустить окно командной оболочки cmd.exe и изменить общие настройки (вид окна, цвет фона и шрифта).
2. Сменить текущий диск на другой (предварительно проверив на компьютере название дополнительного диска);
3. Произвести очистку экрана;
4. Назначить заголовком окна для текущего сеанса интерпретатора командных строк CMD.EXE свою фамилию и группу;
5. Вывести на экран версию операционной системы, установленную в компьютере;
6. Вывести на экран информации о конфигурациях системы;
7. Используя оператор для разделения нескольких команд в одной командной строке, вывести на экран дату, а затем время, установленные в компьютере; при этом на запрос ввода новых параметров (даты и времени) нажмите Enter;
8. Проверить действие вышеперечисленных клавиш управления списком команд в следующем порядке: ↑, ↓, F7, F9, Alt + F7.
9. Подготовить отчет (приложение 1) в виде документа Word, содержащий сведения о выполнении работы с включением формулировки заданий, скриншоты выполненных 1-7 заданий и всплывающего окна клавиш F7 и F9, а также ответы на вопросы (если имеются).

Скриншот активного окна можно сделать с помощью комбинации клавиш Alt+PrtScr, затем вставить его в нужное место с помощью комбинации Ctrl+V.

3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое командная строка?
2. Как запустит командную строку?
3. Как изменить настройки внешнего вида?
4. На какие группы делятся основные команды?

ЛАБОРАТОРНАЯ РАБОТА 2. ФАЙЛОВАЯ СТРУКТУРА

Цель работы: приобрести теоретические знания и практические умения при решении задач по созданию файловой структуры в CMD.EXE

1. ТЕОРЕТИЧЕСКИЙ МАТЕРИАЛ

1.1. Команды работы с каталогами

При работе с каталогами в CMD, часто используются следующие команды:

– *md* – создание каталога

Синтаксис: *md имя_каталога*

где вместо *имя_каталога* указывается имя создаваемого каталога.

При создании одного каталога, который содержит пробел в имени, это имя необходимо указать в кавычках (рис.12), иначе выполнение команды приведет к созданию нескольких каталогов (рис.13).

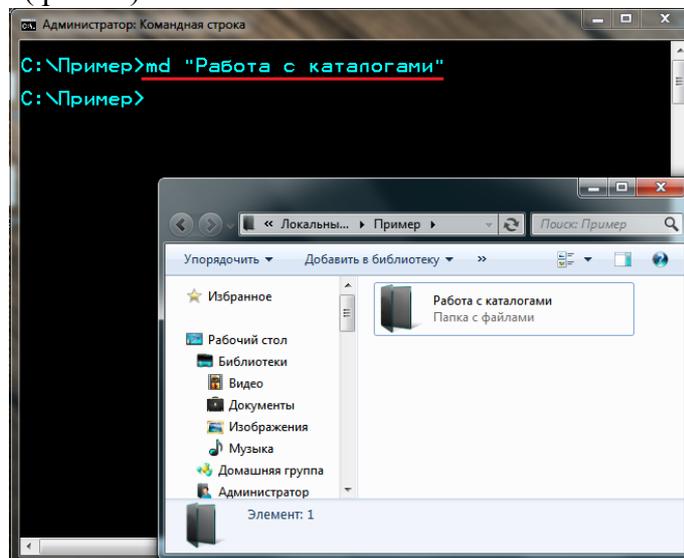


Рис. 12. Создание одного каталога с пробелами в имени

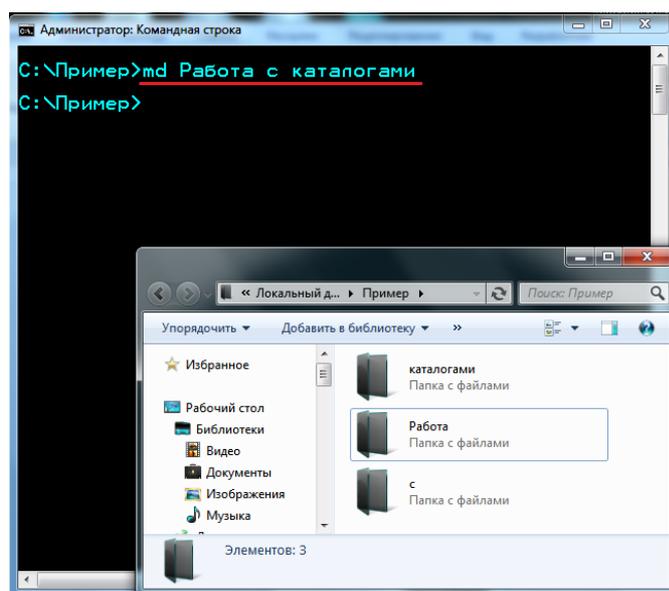


Рис. 13. Создание нескольких каталогов

- *rd* – удаление каталога

Синтаксис удаления пустого каталога: *rd имя_каталога*

где вместо *имя_каталога* указывается имя удаляемого каталога.

Если необходимо удалить не только указанный каталог, но и все содержащиеся в нем файлы и подкаталоги, то синтаксис команды в этом случае выглядит следующим образом:

rd имя_каталога /s (рис.14)

где */s* – параметр удаления дерева каталогов.

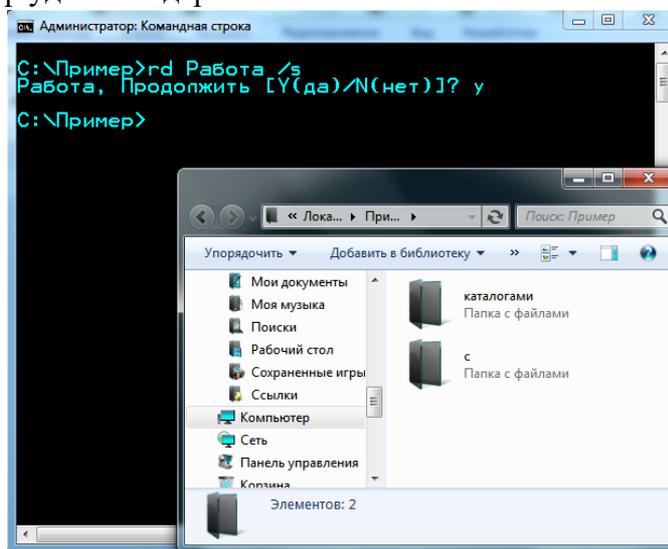


Рис. 14. Удаление каталога

- *cd* – переход в другой каталог

Синтаксис: *cd имя_каталога*

- *cd ..* – выход из каталога

- *cd * – выход в корневой каталог (т.е. переход к диску)

– *dir* – просмотр каталога (т.е. вывод списка файлов и подкаталогов в указанном каталоге)

Синтаксис: *dir диск:\путь\имя_каталога*

- *tree* – графическое представление структуры каталогов

Синтаксис: *tree диск:\путь\имя_каталога* (рис.15)

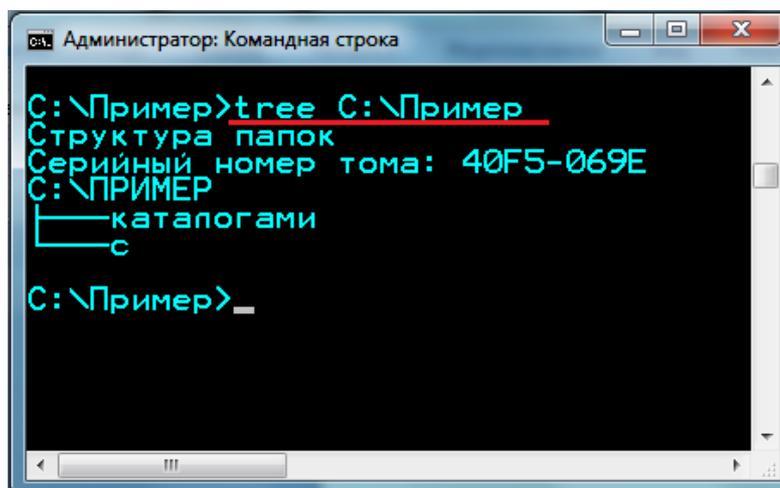


Рис.15. Графическое представление структуры каталога "Пример"

1.2. Команды работы с файлами

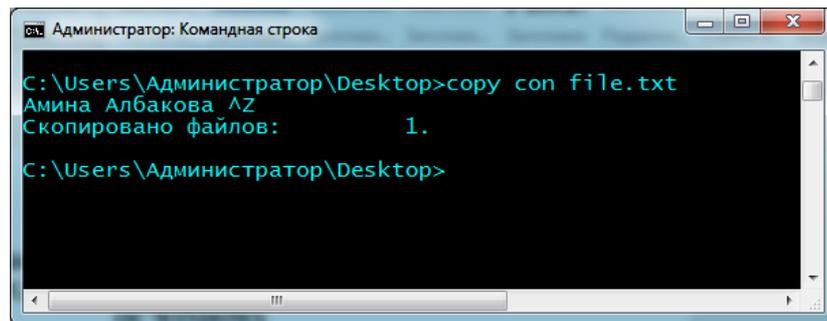
Для создания текстовых файлов в командной строке можно использовать два способа.

Способ 1. Копирование ввода с клавиатуры в файл:

copy con file.txt

где con (англ. Console – Консоль) – устройство стандартного ввода-вывода информации.

При выполнении этой команды данные с клавиатуры записываются в указанный файл. Для завершения ввода текста нажмите комбинацию [Ctrl + Z] или кнопку F6, а затем [Enter]. (рис. 16)



```
Администратор: Командная строка
C:\Users\Администратор\Desktop>copy con file.txt
Амина Албакова ^Z
Скопировано файлов: 1.
C:\Users\Администратор\Desktop>
```

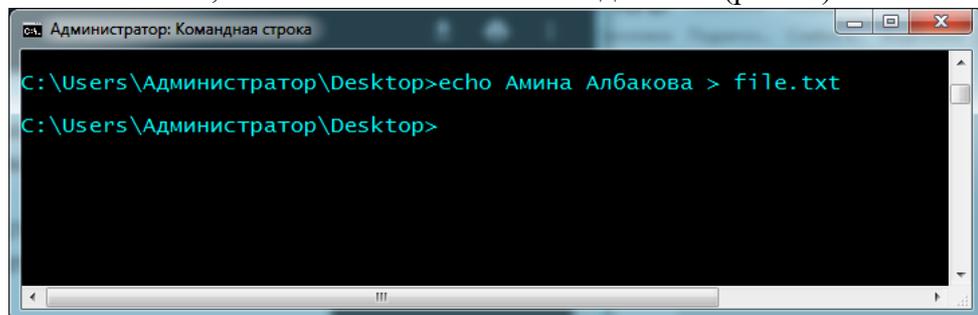
Рис.16. Копирование ввода с клавиатуры в файл

Способ 2. Перенаправление стандартного вывода:

echo текст > file.txt

где команда echo (англ. Echo – эхо) выводит текст, напечатанный после команды, на экран.

В результате, при выполнении этой команды будет создан файл file.txt, который будет содержать в себе текст, напечатанный после команды echo. (рис.17)



```
Администратор: Командная строка
C:\Users\Администратор\Desktop>echo Амина Албакова > file.txt
Амина Албакова
C:\Users\Администратор\Desktop>
```

Рис.17. Перенаправление стандартного вывода

Для вывода на экран содержимое файла, используются следующие команды:

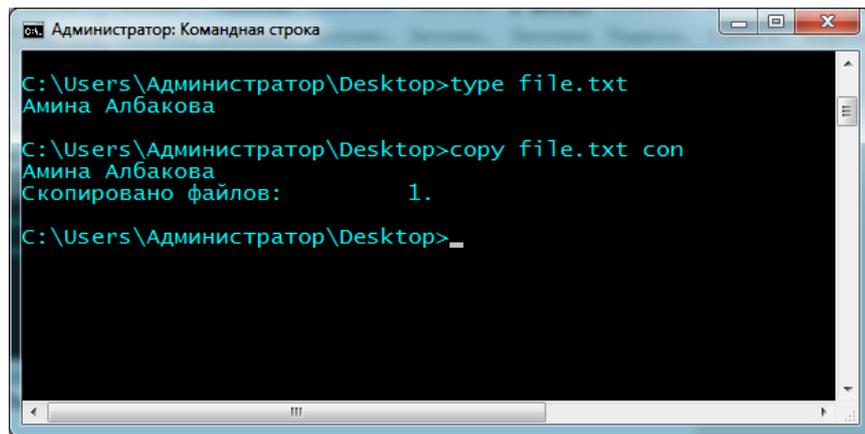
- **type** – команда выводит на экран содержимое текстового файла

Синтаксис: ***type имя_файла***

Пример: ***type file.txt*** (рис.18)

Можно также выполнить копирование файла на экран (устройство стандартного вывода, консоль), например:

copy file.txt con (рис.18)



```
Администратор: Командная строка
C:\Users\Администратор\Desktop>type file.txt
Амина Албакова
C:\Users\Администратор\Desktop>copy file.txt con
Амина Албакова
Скопировано файлов:      1.
C:\Users\Администратор\Desktop>_
```

Рис.18. Вывод на экран содержимое файла

Команда **copy** (англ. Copy – Копировать) или **xcopy** используется, чтобы скопировать один или несколько файлов.

Команда **move** – перемещает файл.

Команда **ren** используется для переименования файлов.

Команда **del** – удаляет файл.

Для объединения нескольких файлов в один файл между именами файлов ставится знак +. Например:

copy f1.txt+f2.txt+ f3.txt f4.txt

объединяет файлы f1.txt, f2.txt и f3.txt в файл f4.txt

Второй параметр команды может содержать имя файла или путь к файлу. Примеры:

- ***copy 1.txt 2.txt*** – копирование файла 1.txt в файл 2.txt в текущем каталоге. Если файл с именем 2.txt уже существует в этом каталоге, то его содержимое заменяется содержимым файла 1.txt. Перед копированием ОС запрашивает согласие пользователя на замену существующего файла.

- ***copy 1.txt d:\t2.txt*** – копирование файла 1.txt в файл 2.txt в каталоге d:\t

Вспомним из предыдущей лабораторной, что для перенаправления стандартного ввода-вывода используют следующие символы-операторы:

> – записать в файл (вместо вывода на экран);

>> – добавить к файлу (вместо вывода на экран);

< – прочитать из файла (вместо ввода с клавиатуры);

Стандартный ввод: если в параметрах командной строки не указано имя файла, то команда принимает данные с клавиатуры. Например, команда **sort** сортирует текстовые строки по алфавиту. Команда **sort** без параметров сортирует текст, вводимый с клавиатуры до нажатия [Ctrl+Z] и [Enter]. Команда **sort f.txt** сортирует текст из файла **f.txt**. Команда **dir > f.txt** отправляет список файлов в файл f.txt вместо вывода на экран. Команда **dir >> f.txt** дописывает список файлов в конец файла f.txt, если этот файл уже существует.

В Windows используют два основных символа-шаблона:

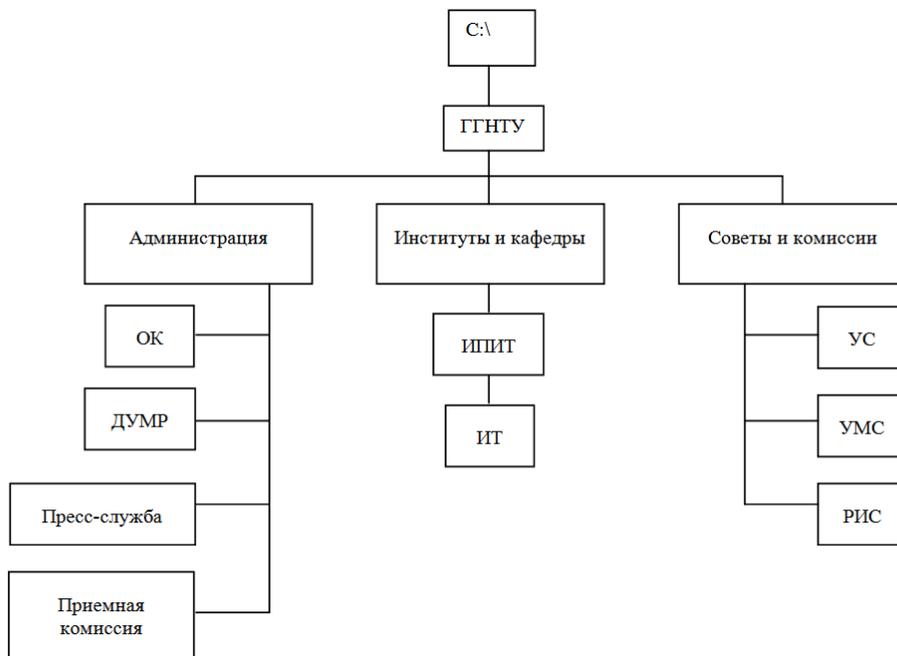
1. * - универсальный заменитель любого количества любых символов;

2. ? - универсальный заменитель любого одиночного символа.

Например: можно переименовывать группу файлов, используя символы-шаблоны, например команда **ren *.txt *.dat** переименует все файлы с расширением txt, присвоив им новое расширение dat

2. ЗАДАНИЕ

1. Назначить заголовком окна для текущего сеанса интерпретатора командных строк CMD.EXE свою фамилию и группу.
2. Сделать текущим диск C: (закройте все папки, если они открыты)
3. Создать на диске C: дерево каталогов с нижеприведенной структурой.



4. Вывести на экран графическое представление структуры каталога ГНТУ
5. Переместить подкаталог *Приемная комиссия* в каталог *Советы и комиссии*, при этом в случае успешного выполнения команды, используя операторы условного выполнения команд, вывести на экран графическое представление структуры каталога *ГНТУ* (прописать обе команды в одну строку)
6. Добавить в каталог ИПИТ два подкаталога ИВТ и СК
7. Вывести на экран содержимое каталога *Институты и кафедры*, при этом отсортировать выводимый на экран список файлов, расположив их в алфавитном порядке, для чего передать вывод команды *dir* на обработку команде *sort*, используя оператор конвейеризации (*dir | sort*)
8. Создать в каталоге ИВТ один файл со следующим содержанием:
file.txt – содержит следующий текст «Информатика и вычислительная техника»
9. Создать в каталоге ИТ три файла со следующим содержанием:
file1.txt – содержит следующий текст «Информационные системы и технологии»
file2.txt – содержит следующий текст «Информационные технологии в образовании»
file3.txt – содержит следующий текст «Информационные технологии в дизайне»
10. Вывести на экран графическое представление структуры каталога ГНТУ со всеми созданными файлами (для этого предварительно выведите справку по соответствующей команде)
11. Объединить (скопировать содержимое) три файла file1.txt, file2.txt и file3.txt в один файл file4.txt, при этом дописав туда первой строчкой следующий текст «Список направлений подготовки студентов по кафедре ИТ»
12. Вывести на экран командной строки содержимое файла file4.txt

13. Удалите следующие файлы: file1.txt, file2.txt и file3.txt
14. Полностью удалить каталог *ГГНТУ*, предварительно выполнив переход в корневой каталог *C:*
15. Подготовить отчет о выполненных заданиях (приложение 1)

3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. В каких вариантах возможно использовать команду *cd*?
2. Способы создания текстовых файлов?
3. Как создать каталог (папку)?

ЛАБОРАТОРНАЯ РАБОТА 3. ПАКЕТНЫЕ ФАЙЛЫ

Цель работы: приобрести теоретические знания и практические умения при создании и работе пакетных файлов.

1. ТЕОРЕТИЧЕСКИЙ МАТЕРИАЛ

Пакетный, файл (или командный файл) – это текстовый файл с расширением **.bat** (batch-пакет, пакетный), содержащий команды процессора `cmd.exe`, выполняемые операционной системой автоматически в порядке, определяемым управляющими командами в этом же файле. Пакетные файлы являются **программами-сценариями** (“скриптами”), автоматизирующими выполнение часто повторяющихся действий по администрированию компьютерных систем, а также разнообразных действий обычных пользователей. Сценарии на основе интерпретатора `cmd.exe` являются наиболее простым для освоения и одновременно достаточно функциональным средством для решения на компьютере разнообразных задач с использованием возможностей самой операционной системы без привлечения дополнительного программного обеспечения. В состав современных операционных систем семейства Windows входят также более мощные оболочки и языки сценариев - сервер сценариев Windows Script Host (WSH) и оболочка Windows PowerShell [4], предназначенные главным образом для профессионалов компьютерщиков. В состав Windows 10 входит также пакетный интерпретатор среды Bash, широко используемый в операционных системах UNIX/Linux. Скриптовый язык среды Bash сходен с языком командного процессора Windows, но в целом эта оболочка обладает значительно большими возможностями, в том числе за счёт наличия в ней мощных утилит командной строки (`grep`, `awk`, `sed` и др.), отсутствующих в оболочке `cmd.exe`.

Синтаксис пакетных файлов

Команды записываются построчно и в простых случаях выглядят так же, как если бы они вводились непосредственно в окне `cmd.exe`. Для расширения функциональности пакетные файлы могут содержать специальные символы, и команды, управляющие ходом выполнения. Кроме того, пакетный файл, как любая программа, может запускаться с параметрами, вызывать на исполнение другие пакетные файлы, и организовывать подпрограммы. В целом правила синтаксиса полнофункциональных пакетных файлов весьма обширны.

Ниже приведены отдельные, необходимые для лабораторной работы сведения.

Запуск пакетного файла

Пакетный файл является исполняемым файлом. Чтобы запустить пакетный файл из командной строки, нужно **ввести его имя и нажать Enter**. При этом расширение указывать не обязательно. Для запуска файла или команды в отдельном окне используется команда **start**,

например: *start dir*

Использование русскоязычных букв (кодировка)

По умолчанию любой текст, набираемый в окне командной строки, кодируется в соответствии с кодовой страницей **866 (DOS)**. Текстовые документы, набираемые в стандартных текстовых редакторах Windows (блокнот, Wordpad), сохраняются в кодировке **1251 (Windows-кодировка (Кириллица))** или **65001 (Кодировка UTF-8)**. Чтобы пакетный файл с русскими буквами правильно интерпретировался и воспроизводился в окне `cmd`, следует:

- либо явно задать его сохранение в **кодировке 866**, тогда не придётся менять кодовую страницу командной строки,
- либо перед его запуском (или в самом пакетном файле) сменить кодовую страницу на **1251** командой

chcp 1251>nul

Практика показывает, что команды не всегда правильно обрабатывают русскоязычный текст, поэтому результат следует оценивать критически.

Пример пакетного файла

Каждая из команд, считываемых из пакетного файла и выполняемых, дублируется на экран. Чтобы запретить дублирование конкретной команды, в начале строки пакетного файла ставится специальный символ **@**. Если нужно подавить дублирование на экран всех команд, используется команда **@echo off**.

Одиночный символ **%** используется для взятия значений параметров пакетного файла (см. рис.19 и рис.20). Парные символы **%** используются для взятия значений переменных в пакетных файлах.

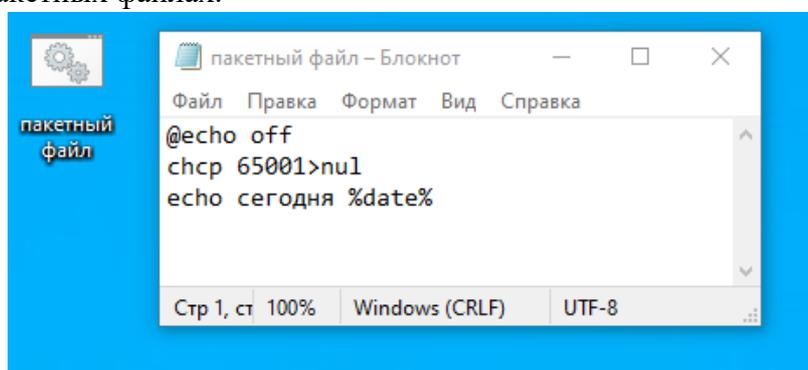


Рис. 19. Создание пакетного файла

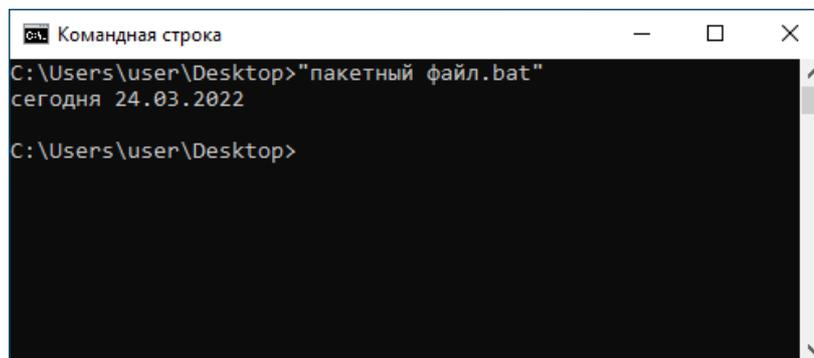


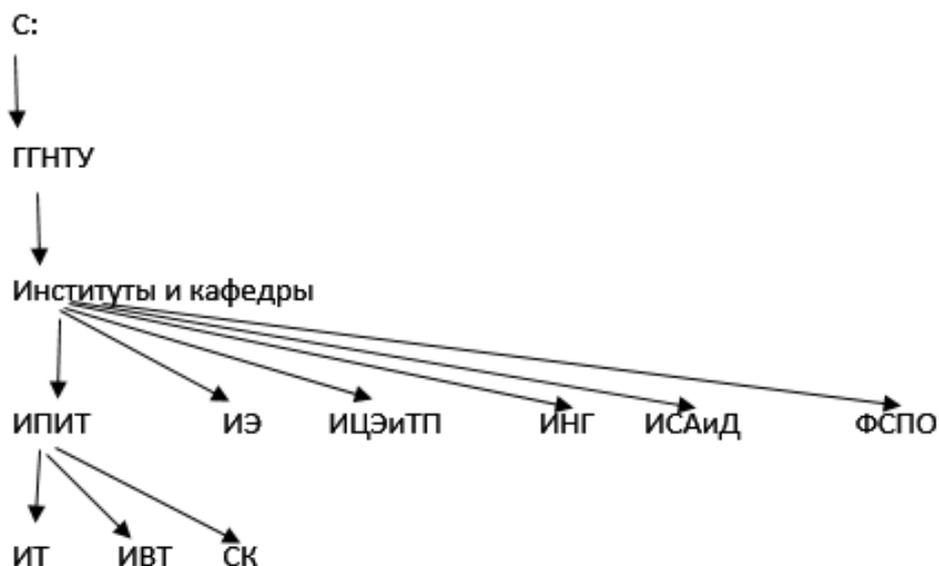
Рис.20. Запуск пакетного файла

2. ЗАДАНИЕ

1. Запустить командную оболочку **cmd** и перейти в текущий диск. Запустить блокнот (**notepad**) или редактор **edit** из командной строки с параметром в виде имени файла: **notepad mycmd.bat** или **edit mycmd.bat**. При этом откроется пустое окно для ввода текста файла. Ввести команды (по одной на строку), показанные в виде примера на рисунке 19. Далее запустить командный файл на выполнение, набрав его имя в командной строке и нажав **ENTER**.

2. Создать пакетный файл (присвоив ему имя ФИО и группу студента) со следующими командами:

2.1. Создать на диске дерево каталогов с нижеприведенной структурой:



2.2. Вывести на экран графическое представление структуры каталога ГГНТУ

2.3. Создать в каталоге ИТ файл file1.txt, содержащий следующий текст:

Информационные системы и технологии, Информационные технологии в образовании, Информационные технологии в дизайне

2.4. Вывести на экран графическое представление структуры каталога ГГНТУ, со всеми файлами

2.5. Переименовать файл «file1.txt» на «Список направлений.doc/.docx» и вывести на экран командной строки содержимое этого файла.

2.6. Полностью удалить каталог ГГНТУ, предварительно выполнив переход в корневой каталог C:\

2.7. Подготовить отчет о выполненных заданиях (приложение 1)

3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое пакетный файл?
2. Как создать пакетный файл? Какое расширение он имеет?
3. Какие виды кодировки существуют? Как их менять?

ЛАБОРАТОРНАЯ РАБОТА 4. ЭЛЕМЕНТЫ ПРОГРАММИРОВАНИЯ В CMD

Цель работы: научиться использовать программные элементы языка пакетных (командных) файлов.

1. ТЕОРЕТИЧЕСКИЙ МАТЕРИАЛ

Как нам известно пакетный файл (или командный файл) может выступать как обычный сценарий, представляющий набор последовательно выполняющихся программ. Однако он, также, может содержать программные конструкции в виде условий, циклов, переходов по меткам по аналогии с программами.

В отличие от программы на языке программирования высокого уровня (C++, Basic, Pascal), командный файл устроен проще и может содержать ограниченное количество программных конструкций. От версии к версии операционной системы Windows язык пакетных файлов становится богаче.

Рассмотрим простейшие, базовые конструкции.

1.1. Переменные

Переменные в командной строке представляют собой именованные значения, которые могут использоваться в командной строке операционной системы для хранения информации и передачи ее между различными процессами и командами. Эти переменные используются для настройки окружения, управления работой команд и программ, а также для хранения информации о системе и пользователях.

Основные команды, часто применяемые в командных файлах для создания и манипулирования переменными:

Rem – строка комментария в пакетном файле

Set – присвоить значение переменной всем, что следует за знаком равенства (=). При этом использования дополнительных параметров расширяет возможности данной команды: например, Set /p позволяет ввести данные.

Call — вызов одного пакетного файла из другого (пример: CALL C:\SCRIPTS\test.bat)

Пример применения вышеупомянутых команд показан на рисунке 21 и 22

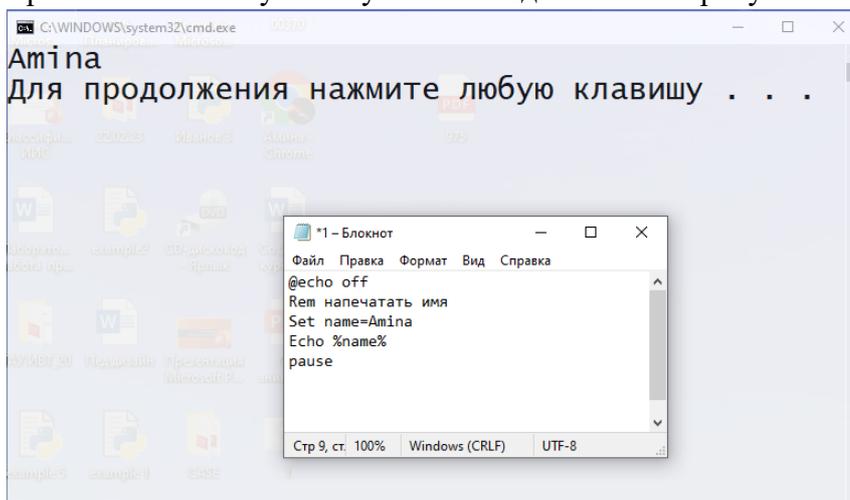


Рис.21. Пример использования Set

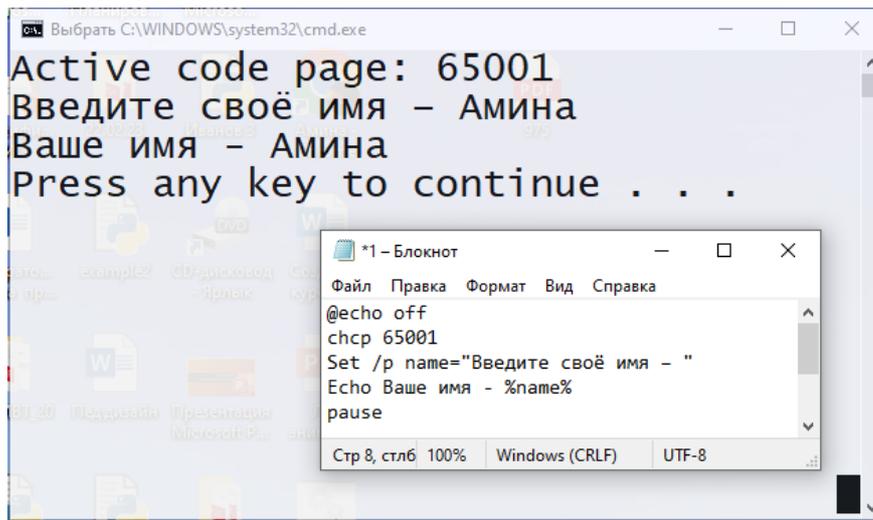


Рис.22. Пример использования Set /p

Вспомним, что одиночный символ % используется для взятия значений параметров пакетного файла, а парные символы % используются для взятия значений переменных в пакетных файлах (например, %CD% - Принимает значение строки текущей директории; %DATE% - Принимает значение текущей даты и т.д)

Таким образом, переменные могут быть использованы для различных целей, таких как настройка путей поиска файлов, установка параметров программ, передача информации скриптам и многое другое. Их значение можно прочитать и изменить в командной строке или в скриптах, чтобы влиять на поведение системы и программ.

1.2. Условные операторы

Условные операторы позволяют выполнять команды только в том случае, если определенное условие истинно.

В пакетных файлах Windows оператор if используется для проверки условий и выполнения определенных действий в зависимости от результатов проверки. Некоторые из ключевых слов, которые можно использовать с оператором if в пакетном файле Windows, включают:

- exist - проверяет, существует ли файл или папка.
- not - инвертирует результат проверки условия.
- errorlevel - проверяет, была ли обнаружена ошибка при выполнении предыдущей команды.
- equ - проверяет, равны ли два значения
- neq - проверяет, не равны ли два значения
- defined - проверяет, определена ли переменная
- LSS - проверяет, меньше ли первое значение, чем второе значение
- LEQ - проверяет, меньше или равно ли первое значение, чем второе значение
- GTR - проверяет, больше ли первое значение, чем второе значение
- GEQ - проверяет, больше или равно ли первое значение, чем второе значение
- NOT - инвертирует результат проверки условия

Пример:

```
@echo off
if exist "C:\Temp\file.txt" (
```

```
    echo The file exists.
) else (
    echo The file does not exist.
)
```

Обратите внимание на скобки, которые окружают команды, выполняемые при истинном или ложном условии. В этом примере, если файл существует, выполняется команда "echo The file exists.". Скобки используются для группировки этой команды, чтобы она выполнялась только в том случае, если условие истинно. Если условие ложно, то выполняется команда "echo The file does not exist." в блоке else.

Операторы символы:

« :оператор » позволяет сравнивать строки на основе различных критериев, таких как наличие, отсутствие, содержание и другие (пример: string1:E string2 - выполняет сравнение строк на равенство.

= позволяет проверить равны ли два значения (например, string1==string2 - проверяет, равны ли две строки)

!= позволяет проверить не равны ли два значения (например, string1!=string2 - проверяет, не равны ли две строки)

1.3.Циклы

В пакетных файлах командной строки Windows можно использовать циклы для автоматизации повторяющихся задач. Циклы позволяют выполнять определенный набор команд несколько раз в зависимости от заданных условий. В командной строке Windows для этого используют оператор "for", который позволяет выполнять набор команд для каждого элемента в заданном списке или диапазоне значений.

2. ЗАДАНИЕ

Необходимо создать 4 пакетных файла, которые будут выполнять следующие действия:

1. Первый пакетный файл: Проверка наличия файла, имя которого запрашивается у пользователя (т.е. вводится после запуска пакетного файла) и его копирование в резервную директорию (с учетом создания (любым способом) ее с произвольным именем в корневом каталоге (на диске), если ее еще нет). ". Если же файла не существует, выводится сообщение об ошибке с помощью команды "echo". Добавьте в конце паузу для того, чтобы увидеть результат (pause). Обратите внимание на то, что в имени файла, введенном пользователем, необходимо указывать путь к файлу, если он находится в другой директории. Если файл находится в той же директории, что и пакетный файл, то достаточно указать только имя файла.

2. Второй пакетный файл: Запрос у пользователя его фамилии, имени, отчества, название студенческой группы, кафедры и института. Если имя задано необходимо вывести на экран приветствие с использованием введенных всех значений переменных, в противном случае вывести ошибку или отказ доступе. Добавьте в конце паузу для того, чтобы увидеть результат (pause).

3. Третий пакетный файл: Проверка пользователя на совершеннолетие, возраст которого запрашивается у пользователя. Если пользователь совершеннолетний (т.е. введено числа больше или равно 18) , то должен выводиться текст приветствия, если нет – выводится

текст «Вы слишком молоды». Добавьте в конце паузу для того, чтобы увидеть результат (pause).

4. Добавьте во второй пакетный файл команду вызова третьего пакетного файла

5. Четвертый пакетный файл: Запрос у пользователя его фамилии, имени, отчества, название студенческой группы. Проверка введенных значений переменных по очереди: если имя пользователем не задано (т.е. результат пустой) – выводится текст об ошибке; если фамилия не задана, то на экран выводится текст приветствия с заданным именем; если отчество не задано, то на экран выводится текст приветствия с заданным именем и фамилией; если группа не задана, то на экран выводится текст приветствия с заданным именем, фамилией и отчеством; если все четыре переменные содержат значения, на экран выводится текст приветствия со всеми значениями. Далее создайте другой пакетный файл, который содержит проверку является ли пользователь совершеннолетним (для этого должен быть задан запрос возраста у пользователя), если да – текст приветствие, если нет – текст “вы слишком молоды”. Запустите второй пакетный файл через первый. Добавьте в конце паузу для того, чтобы увидеть результат (pause).

3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое переменная и как ее объявить?
2. Что такое условный оператор?
3. Какой оператор используется для проверки условий?
4. Что такое циклы?
5. Для чего используется команда «Call»?

ЛАБОРАТОРНАЯ РАБОТА 5. ПЛАНИРОВАНИЕ ЗАДАЧ

Цель работы: приобрести практические навыки планирования задач в операционной системе Windows.

1. ТЕОРЕТИЧЕСКИЙ МАТЕРИАЛ

В системах Windows обновлены два основных средства планирования задач: автономная оснастка Планировщик заданий (Task Scheduler) и утилита командной строки schtasks.exe. С их помощью можно составлять расписание для запуска командных файлов, прикладных программ или различных утилит для обслуживания системы. Возможны сложные расписания для запуска и останова задач, привязанные как ко времени, так и к системам событий; одновременно могут запускаться несколько программ, и возможны различные действия по их завершению.

Для запуска программ существуют различные регламенты: однократно, ежедневно, еженедельно или ежемесячно в заданные дни, при загрузке системы или регистрации в ней, а также при бездействии системы (idle state). Планировщик позволяет задавать достаточно сложное расписание для выполнения заданий, в котором задаются продолжительность задания, время его окончания, количество повторов и т. п.

Утилита schtasks, запущенная в окне командной строки без параметров, позволяет видеть список заданий, выполняющихся в системе в данный момент (в системах Windows 7 даже по умолчанию этот список достаточно внушительный — об этом следует помнить, если возникнет желание отключить сервис планировщика, поскольку может быть заблокирован запуск важных задач).

1.1. Автономная оснастка «Планировщик Заданий» (Task Scheduler)

Служба *Планировщик заданий* (Task Scheduler; имя сервиса Schedule) инсталлируется вместе с системой и автоматически запускается при ее загрузке.

С помощью Планировщика заданий Windows можно запланировать запуск любой программы в определенное время с определенными условиями. Также можно запланировать отправку электронной почты и даже вывод на экран определенного сообщения.

В Windows существуют два основных типа планируемых заданий

Стандартные задания. Задания этого типа используются для автоматизации повседневных задач обслуживания компьютерной системы. Пользователь может видеть эти задания и модифицировать их в случае необходимости.

Скрытые задания. Задания этого типа используются для автоматизации выполнения специальных системных задач. Эти задания в стандартном режиме просмотра скрыты от пользователей и их изменение нежелательно.

Каждое задание планировщика имеет определенные свойства:

Триггеры - задают условия начала и завершения выполнения задания. Выполнение задания можно начинать по расписанию, а также по событиям - при входе пользователя в систему, при запуске компьютера, при подключении/отключении терминальной сессии или при записи в журналы системы событий с определенным кодом.

Действия - определяют операцию, которую должно выполнять запущенное на выполнение задание, в качестве которого может использоваться приложение или сценарий.

Условия - определяют условия, при которых активированное задание запускается или останавливается. Например, условия можно использовать, чтобы выполнение задания было возможным только в случае наличия определенного сетевого подключения. С помощью условий можно запускать, останавливать и перезапускать задания.

Параметры - определяют дополнительные параметры, влияющие на выполнение задания – перезапуск после сбоя, выполнение просроченного задания, поведение при наличии уже выполняющегося задания.

Чтобы запустить Планировщик заданий, можно воспользоваться главным меню *Панель управления - Администрирование - Управление компьютером - Планировщик заданий*, или *Выполнить* (комбинация Win+R) - *taskschd.msc*.

После запуска открывается окно для работы с планировщиком (рис. 23).

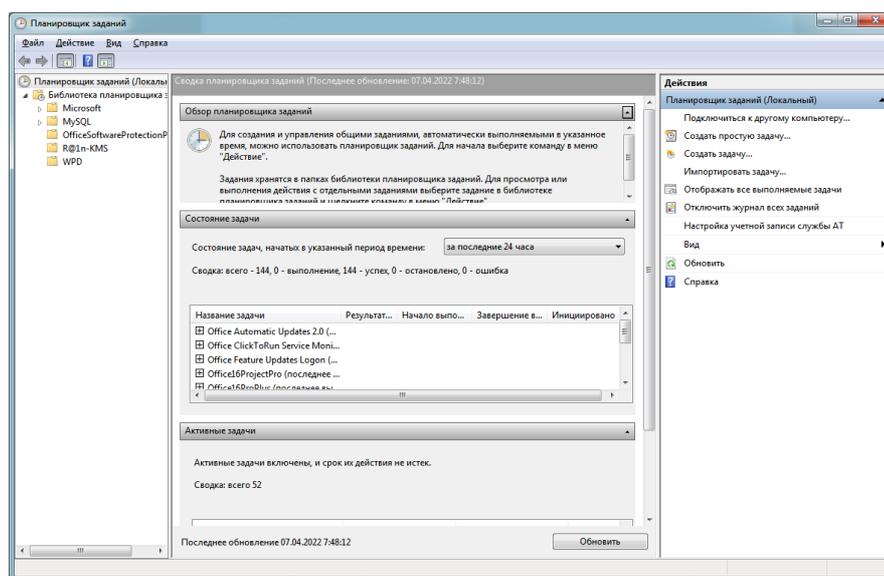


Рис. 23. Главное окно

В левой части основного окна отображаются списки заданий, упорядоченные в соответствии с назначением, в виде структуры папок. В средней части отображается информация о состоянии заданий и их свойствах. В правой части отображается меню действий, допустимых по отношению к заданиям. По умолчанию отображаются задания, относящиеся к локальному компьютеру.

Библиотека планировщика заданий, отображаемая в левой части окна оснастки планировщика, имеет довольно непростую иерархическую структуру, поэтому, для освоения приемов работы с запланированными задачами, можно создать отдельную папку, с использованием контекстного меню, вызываемого правой кнопкой мышки и пункта *Создать папку*, ввести имя папки, и в дальнейшем, именно в ней создавать свои тестовые или рабочие задания.

В расположенном справа меню *Действия* нажмите *Создать задачу* или *Создать простую задачу*. (рис.24)

В окне *Создание задачи* введите название новой задачи и ее описание. Если нужно запускать программу с повышенными правами администратора, то активируйте функцию *Выполнить с наивысшими правами*.

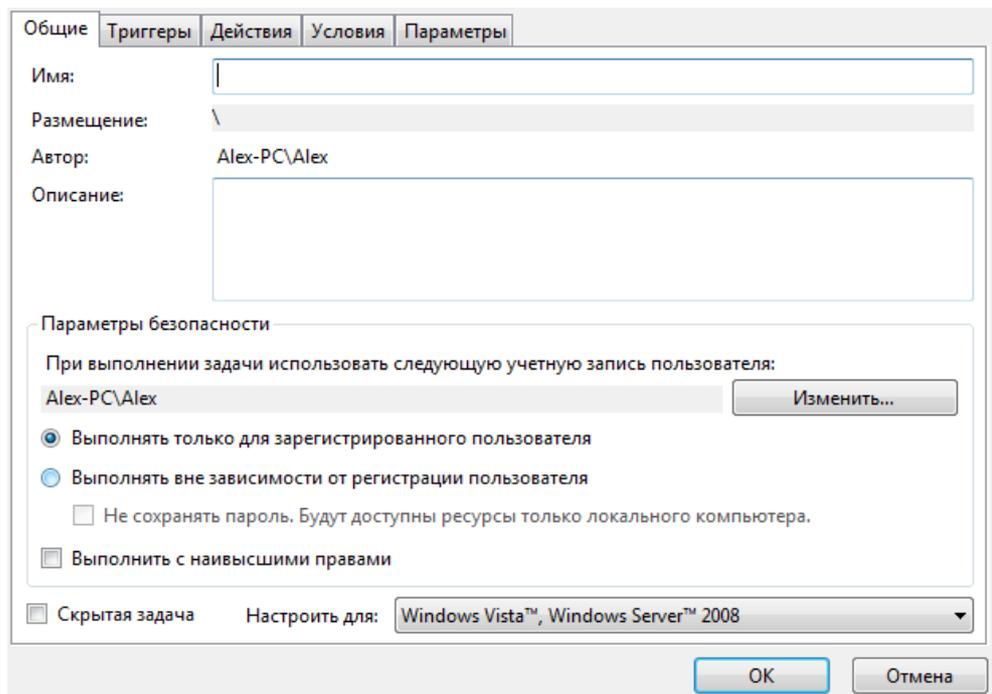


Рис. 24. Окно создания задачи

Переключитесь на вкладку *Триггеры*, нажмите кнопку *Создат* и назначьте дату, время и частоту выполнения задачи. Чтобы установить дату, когда выполнение задачи по расписанию будет прекращено, активируйте функцию *Срок действия* и укажите дату и время.

Перейдите на вкладку *Действия* и нажмите кнопку *Создать*. Чтобы запланировать выполнение программы по указанному расписанию, в раскрывающемся списке *Действие* выберите *Запуск программы* и с помощью кнопки *Обзор* укажите местоположение исполняемого файла.

Перейдите на вкладку *Условия* и укажите условия выполнения задачи. Также можно задать дополнительные параметры выполнения задачи на вкладке *Параметры*.

После завершения всех этих действий, нажмите *ОК*, чтобы создать задачу.

Чтобы изменить созданное задание, откройте *Планировщик заданий*, выделите задачу в списке, в правом меню *Действия* выберите *Свойства*, отредактируйте нужные параметры и нажмите *ОК*.

Чтобы удалить запланированное задание, откройте *Планировщик заданий*, выделите задачу в списке, в правом меню *Действия* выберите *Удалить* и подтвердите свои намерения.

Примечание. Чтобы *Планировщик заданий* функционировал, служба планировщика заданий должна автоматически запускаться вместе с Windows. Откройте *Панель управления (расширенный вид)* -> *Администрирование* -> *Службы*. В списке служб найдите *Планировщик заданий*, дважды щелкните по ней, на вкладке *Общие* установите тип запуска *Авто* и нажмите *ОК*. (рис.25)

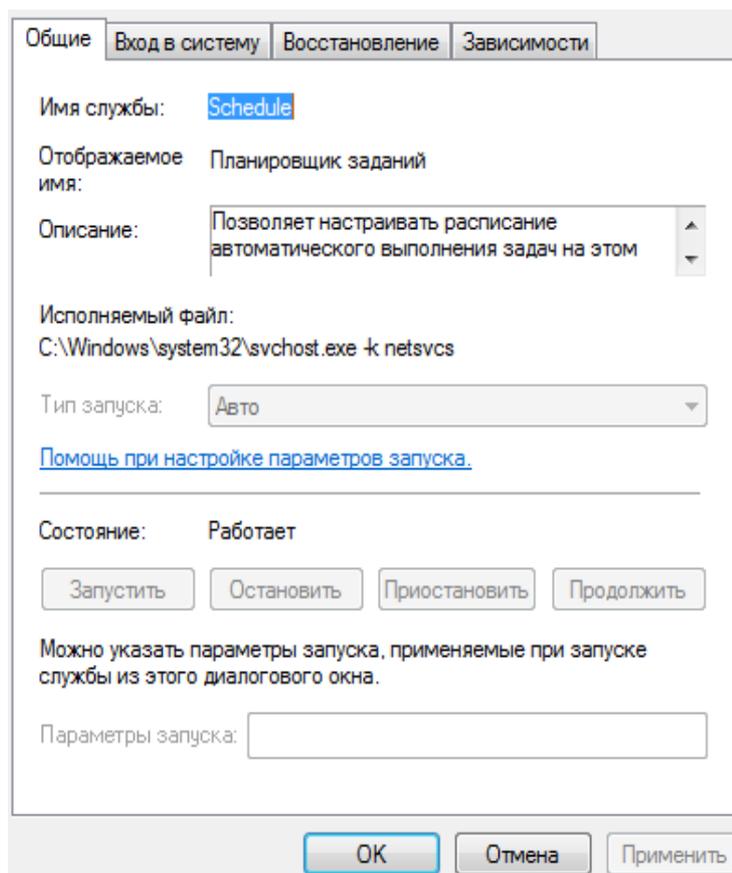


Рис.25. Вкладка «Общие» Планировщика заданий

Для работы с созданным заданием щелкните по нему правой кнопкой мышки в основной панели и в контекстном меню выберите одну из следующих команд:

- Удалить (Delete) — полностью удалить задание;
- Отключить (Disable) — временно отключить задание;
- Свойства (Properties) — просмотреть и/или редактировать свойства задания.
- Экспортировать (Export) — экспортировать задание в файл, данные из которого можно импортировать на другом компьютере.
- Выполнить (Run) — запустить задачу на выполнение;
- Завершить (End) — если задача выполняется, остановить ее выполнение.

Задания, созданные пользователями и прикладными программами, можно изменять или удалять без каких-либо особых проблем, но большинство **заданий, созданных самой операционной системой, трогать не стоит**, поскольку это может иметь неприятные последствия. Для отображения системных задач в меню "Вид" необходимо установить флажок "Отобразить скрытые задачи" (Show hidden tasks).

1.2. Утилита Schtasks

Утилита *schtasks.exe* является стандартным средством управления планировщиком и позволяет создавать, удалять, изменять и опрашивать запланированные задачи на локальном или удаленном компьютере.

СИНТАКСИС: **SCHTASKS /параметр [аргументы]**

ПАРАМЕТРЫ:

/Create	Создание запланированной задачи.
/Delete	Удаление запланированной задачи или задач.
/Query	Вывод данных о всех запланированных задачах.
/Change	Изменение свойств запланированной задачи.
/Run	Выполнение запланированной задачи по требованию.
/End	Остановка выполняющейся запланированной задачи.
/ShowSid	Отображение идентификатора безопасности, соответствующего имени запланированной задачи.
/?	Вывод справки по использованию

Пример вывода справки по отдельным параметрам команды: *SCHTASKS/ShowSid/?*

Изучим для примера более детально параметр Create (Создание). Для создания новых задач используется следующий формат командной строки:

SCHTASKS /Create

```
[/S <система> [/U <пользователь> [/P [<пароль>]]]]  
  [/RU <имя_пользователя> [/RP <пароль>]]  
/SC <расписание> [/MO <модификатор>] [/D <день>] [/M <месяцы>] [/I <простой>]  
/TN <имя_задания>  
/TR <выполнение>  
  [/ST <время_начала>] [/RI <интервал>]  
[ {/ET <время_окончания> | /DU <длительность>} [/K] [/XML xmlfile] [/V1]]  
  [/SD <дата_запуска>] [/ED <дата_окончания>] [/IT | /NP] [/Z] [/F]
```

Параметры и их значения представлены в таблице 6.

Таблица 6. – Параметры и их значения

Параметр	Значение
/S	Удаленная система для подключения. Если параметр не задан по умолчанию система считается локальной.
/U	Контекст, в котором будет выполняться команда SchTasks.exe
/P	Пароль для указанного пользовательского контекста. Запрос ввода, если не указан
/RU	Учетная запись (пользовательский контекст), под которой выполняется задача. Для системной учетной записи допустимыми значениями являются "", "NT AUTHORITY\SYSTEM" или "SYSTEM". Кроме того, для задач v2 доступны параметры: NT AUTHORITY\LOCALSERVICE и NT AUTHORITY\NETWORKSERVICE, также, как и известные идентификаторы безопасности для всех трех параметров.
/RP	Пароль указанного пользователя. Для запроса пароля следует указать "*" или пустую строку. Этот пароль не используется для системной учетной записи. Необходимо сочетать с параметрами /RU или ключ /XML.
/SC	Задаёт частоту повторения задачи. Допустимы следующие типы расписания: MINUTE, HOURLY, DAILY, WEEKLY, MONTHLY, ONCE, ONSTART, ONLOGON, ONIDLE, ONEVENT
/MO	Уточнение типа расписания для более детального контроля за периодичностью. Допустимые значения перечислены в одноименном разделе
/D	День недели, на который запланировано выполнение задачи. Допустимые значения: MON, TUE, WED, THU, FRI, SAT, SUN и для расписаний типа

	MONTHLY - с 1 по 31 (дни месяца). Подстановочный знак "*" указывает все дни
/M	Месяцы в году. По умолчанию запуск выполняется в первый день месяца. Допустимые значения: JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC. Подстановочный знак "*" указывает все месяцы
/I	Время простоя, которое необходимо выждать, прежде чем запускать запланированную задачу ONIDLE. Допустимые значения: 1-999 мин
/TN	Имя, однозначно идентифицирующее запланированную задачу
/TR	Путь и имя файла программы, которая должна быть запущена в назначенное время. Пример: C:\windows\system32\calc.exe
/ST	Время запуска задачи. Формат указания времени: ЧЧ:мм (24-часовой), например, 14:30 вместо 2:30 PM. Если параметр /ST не задан, то по умолчанию используется текущее время. Этот параметр должен использоваться вместе с /SC ONCE
/RI	Интервал повторения в минутах. Этот параметр неприменим для следующих типов расписания: MINUTE, HOURLY, ONSTART, ONLOGON, ONIDLE, ONEVENT. Допустимый диапазон значений: 1-599940 мин Если указан параметр /ET или /DU, то по умолчанию используется 10 мин
/ET	Время завершения выполнения задачи. Формат указания времени: ЧЧ:мм (24-часовой), например, 14:50 вместо 2:50 PM. Этот параметр неприменим для следующих типов расписания: ONSTART, ONLOGON, ONIDLE, ONEVENT
/DU	Длительность выполнения задачи. Формат указания времени: ЧЧ:мм. Не применяется с ключом /ET и расписаниями следующих типов: ONSTART, ONLOGON, ONIDLE, ONEVENT. Для задач /V1, если указан параметр /RI, по умолчанию используется до 1 часа.
/K	Завершение задания по достижении времени завершения или при превышении длительности. Этот параметр неприменим для следующих типов расписания: ONSTART, ONLOGON, ONIDLE, ONEVENT. Параметр /ET или /DU должен быть указано.
/SD	Дата первого запуска задачи. Формат: dd/mm/yyyy. По умолчанию используется текущая дата. Этот параметр неприменим для следующих типов расписания: ONCE, ONSTART, ONLOGON, ONIDLE, ONEVENT
/ED	Дата последнего запуска задачи. Формат: dd/mm/yyyy. Не применяется для расписаний следующих типов: ONCE, ONSTART, ONLOGON, ONIDLE, ONEVENT
/EC	Канал событий для триггеров OnEvent
/IT	Разрешает интерактивное выполнение задачи в том случае, если пользователь /RU выполнил вход в систему во время выполнения задания. Задание выполняется, только если пользователь вошел в систему
/NP	Пароль не сохраняется. Задача выполняется в неинтерактивном режиме. как указанный пользователь. Доступны только локальные ресурсы
/Z	Пометка задачи для удаления, если не предусмотрено ее повторное выполнение
/XML	Создание задачи из указанного файла XML. Нельзя сочетать с переключателями /RU и /RP или /RP один, если XML-файл задачи уже содержит участника
/V1	Создание задачи, видимой для платформ, предшествовавших версии Vista. Несовместим с параметром /XML
/F	Принудительное создание задачи и подавление предупреждающих сообщений, если указанная задача уже существует
/RL	Уровень выполнения задачи. Допустимые значения: LIMITED и HIGHEST. По умолчанию используется LIMITED
/DELAY	Время ожидания перед запуском задачи после включения триггера. Формат указания времени: ммм:сс. Этот параметр применим только к типам расписания ONSTART, ONLOGON, ONEVENT

Модификаторы: ключ /МО предусматривает следующие значения:

MINUTE: 1 - 1439 мин.

HOURLY: 1 - 23 ч.

DAILY: 1 - 365 дней.

WEEKLY: недели с 1 по 52.

ONCE: без модификаторов.

ONSTART: без модификаторов.

ONLOGON: без модификаторов.

ONIDLE: без модификаторов.

MONTHLY: 1 - 12 или

FIRST, SECOND, THIRD, FOURTH, LAST, LASTDAY.

ONEVENT: строка запроса событий XPath.

2. ЗАДАНИЕ

1. С помощью Планировщика заданий, создайте задачу в предварительно созданной вами папке (в Библиотеке планировщика заданий) с произвольным именем, которая будет запускать браузер при входе в систему пользователя.

2. С помощью Планировщика заданий, создайте задачу с произвольным именем в своей папке, которая будет запускать любой ваш (работающий) пакетный файл ежедневно в 12:00.

3. Отредактируйте первую задачу, установив ей срок действия (на ваш выбор).

4. Просмотрите список активных задач на вашем компьютере как с помощью автономной оснастки Планировщик заданий, так и утилиту командной строки schtasks.exe

5. С помощью командной строки, создайте задачу с произвольным именем, который будет запускать Microsoft Word при входе любого пользователя в систему.

6. Создайте пакетный файл, который будет копировать все файлы с расширение «.txt» из одной папки в другую (папки выбираете или создаёте на ваше усмотрение). Далее запланируйте запуск пакетного файла в первый понедельник каждого месяца, сначала через автономную оснастку, потом через утилиту командной строки (при этой присвоив произвольное имя для запланированной задачи).

7. Просмотрите свойства запланированных вами задач через автономную оснастку и утилиту командной строки.

8. Показав созданные задачи преподавателю, удалите все созданные задачи таким же образом, как и были созданы

9. Подготовить отчет о выполненных заданиях (приложение 1)

3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Для чего предназначено планирование заданий?

2. Охарактеризовать каждый этап создания задания.

3. Каким образом создать несколько заданий?

4. Каким образом можно пробудить компьютер для выполнения задания?

ЛАБОРАТОРНАЯ РАБОТА 6. РАЗГРАНИЧЕНИЕ ПРАВ ПОЛЬЗОВАТЕЛЕЙ

Цель работы: приобрести навыки разграничения прав пользователя в ОС Windows

1. ТЕОРЕТИЧЕСКИЙ МАТЕРИАЛ

Учетная запись пользователя Windows представляет собой набор данных, определяющих, к каким папкам и файлам пользователь имеет доступ, какие изменения могут вноситься пользователем в работу компьютера, а также персональные настройки пользователя, такие как фон рабочего стола и экранная заставка. Применение учетных записей позволяет нескольким пользователям работать на одном компьютере с использованием собственных файлов и параметров. Для доступа к учетной записи используется имя пользователя и пароль.

В операционной системе Windows для компьютеров, которые состоят в рабочей или домашней группе, учетные записи можно создавать следующими способами:

1. Создание учетной записи при помощи диалога «Управление учетными записями пользователей»
2. Создание учетной записи при помощи диалога «Учетные записи пользователей»
3. Создание учетной записи при помощи оснастки «Локальные пользователи и группы»
4. Создание учетной записи при помощи командной строки

1.1. Создание учетной записи при помощи оснастки «Локальные пользователи и группы»

Оснастка «*Локальные пользователи и группы*» расположена в компоненте «*Управление компьютером*», представляющем собой набор средств администрирования, с помощью которых можно управлять одним компьютером, локальным или удаленным. Оснастка «*Локальные пользователи и группы*» служит для защиты и управления учетными записями пользователей и групп, размещенных локально на компьютере. Можно назначать разрешения и права для учетной записи локального пользователя или группы на определенном компьютере (и только на этом компьютере).

Использование оснастки «*Локальные пользователи и группы*» позволяет ограничить возможные действия пользователей и групп путем назначения им прав и разрешений. Право дает возможность пользователю выполнять на компьютере определенные действия, такие как архивирование файлов и папок или завершение работы компьютера. Разрешение представляет собой правило, связанное с объектом (обычно с файлом, папкой или принтером), которое определяет, каким пользователям, и какой доступ к объекту разрешен.

Для того чтобы создать локальную учетную запись пользователя при помощи оснастки «*Локальные пользователи и группы*», нужно сделать следующее:

1. Откройте оснастку «*Локальные пользователи и группы*» одним из следующих способов:

А. Нажмите на кнопку «*Пуск*» для открытия меню, откройте «*Панель управления*» и из списка компонентов панели управления выберите «*Администрирование*», затем откройте компонент «*Управление компьютером*». В «*Управлении компьютером*» откройте «*Локальные пользователи и группы*»;

В. Открыть «*Консоль управления MMC*». Для этого нажмите на кнопку «*Пуск*», в поле поиска введите *mmc*, а затем нажмите на кнопку «*Enter*». Откроется пустая консоль

ММС. В меню «Файл» выберите команду «Добавить или удалить оснастку» или воспользуйтесь комбинацией клавиш *Ctrl+M*. В диалоге «Добавление и удаление оснасток» выберите оснастку «Локальные пользователи и группы» и нажмите на кнопку «Добавить». Затем нажмите на кнопку «Готово», а после этого - кнопку «ОК». В дереве консоли откройте узел «Локальные пользователи и группы (локально)»;

С. Воспользоваться комбинацией клавиш *+R* для открытия диалога «Выполнить». В диалоговом окне «Выполнить», в поле «Открыть» введите *lusrmgr.msc* и нажмите на кнопку «ОК».

1.2. Создание учетной записи при помощи командной строки

NET – управление сетевой конфигурацией в командной строке Windows. При использовании команды без параметров или с параметром *help* отображается краткая подсказка по использованию.

Синтаксис: **NET**

[ACCOUNTS | COMPUTER | CONFIG | CONTINUE | FILE | GROUP | HELP | HELPMMSG | LOCALGROUP | PAUSE | SESSION | SHARE | START | STATISTICS | STOP | TIME | USE | USER | VIEW]

Для получения краткой подсказки по подкоманде, можно использовать синтаксис: *net share /?* - отобразить подсказку по использованию команды *NET SHARE*

Для получения более подробной справочной информации используется подкоманда **HELP**:

net help share - отобразить подробную справку по использованию команды *NET SHARE*

Как правило, подробная подсказка содержит не только описание синтаксиса команды, но еще и описание отдельных параметров, ее назначение и примеры использования.

Рассмотрим более подробно необходимые нам команды.

Команда *NET USER* предназначена для добавления, редактирования или просмотра учетных записей пользователей на компьютерах. При выполнении команды в командной строке без параметров отображается список учетных записей пользователей Windows, присутствующих на компьютере. Информация об учетных записях пользователей хранится в базе данных Windows.

К основным возможностям команды *Net User* относятся:

1. Добавление учетной записи

Синтаксис: **net user имя_пользователя /add**

2. Добавление пароля учетной записи

Синтаксис: **net user имя_пользователя {пароль | *} /add**

3. Просмотр свойств учетной записи

Синтаксис: **net user имя_пользователя**

4. Удаление учетной записи

Синтаксис: **net user имя_пользователя /delete**

Дополнительную информацию можно получить, набрав **net help user** или **net user /?**

2. ЗАДАНИЕ

2.1. Выполните следующие действия при помощи оснастки «Локальные пользователи и группы»:

1. Средства регистрации пользователей

- открыть список зарегистрированных пользователей (Панель управления | Администрирование | Управление компьютером | Локальные пользователи и группы | Пользователи);
- с помощью команды контекстного меню (Новый пользователь) создать для себя учетную запись с вашим именем; самостоятельно изучить смысл четырех дополнительных параметров создаваемой учетной записи;
- выделить имя вновь зарегистрированного пользователя и с помощью команды контекстного меню (Свойства) просмотреть ее свойства;
- на вкладке «Общие» изучить разницу между отключением и блокировкой учетной записи;
- перейти на вкладку «Членство в группах» и подготовить ответ на вопрос, в какую группу по умолчанию включается вновь созданный пользователь; с помощью кнопок «Добавить», «Дополнительно» и «Поиск» включить вновь созданного пользователя также в группу «Опытные пользователи»;
- изучить список команд контекстного меню при не выбранном и при выбранном имени учетной записи вместе с пояснениями их смысла, а также изучить следующее:
 1. когда должна применяться команда «Задать пароль»,
 2. в чем опасность ее применения,
 3. как должна происходить смена пароля пользователем.

2. Средства работы с группами:

- открыть список групп (Панель управления | Администрирование | Управление компьютером | Локальные пользователи и группы | Группы);
- изучить сведения об автоматически создаваемых группах пользователей, их именах и характеристиках прав их членов;
- создать новую группу в системе с именем «Начинающие пользователи», а также ответить на вопрос, в чем целесообразность разбиения множества пользователей на группы.

3. Порядок назначения прав пользователям:

- открыть окно настройки прав пользователей (Панель управления | Администрирование | Локальная политика безопасности | Локальные политики | Назначение прав пользователя);
- добавить группу «Начинающие пользователи» к списку пользователей, обладающих правом «Локальный вход в систему»;
- добавить созданную учетную запись в число пользователей, обладающих правом «Доступ к компьютеру из сети»;
- пояснить алгоритм исключения пользователя\группы из числа пользователей\групп, обладающим каким-либо правом;

4. Определение параметров политики безопасности, относящихся к аутентификации и авторизации пользователей при интерактивном входе:

- открыть окно определения параметров безопасности для паролей (Панель управления | Администрирование | Локальная политика безопасности | Политики учетных записей | Политика паролей);
- самостоятельно изучить сведения о порядке назначения максимального и минимального сроков действия паролей и ответ на вопрос о смысле подобных ограничений;
- самостоятельно изучить сведения о порядке назначения минимальной длины и ограничений на сложность паролей, а также ответы на вопросы, какие и почему требования по сложности предъявляются к паролям в операционной системе Windows (с помощью справочной подсистемы);
- самостоятельно изучить сведения о назначении параметров «Требовать неповторяемости паролей» и «Хранить пароли всех пользователей в домене, используя обратимое шифрование» (с помощью справки Windows);
- самостоятельно изучить параметры политики безопасности, относящихся к паролям; открыть окно определения параметров безопасности для политики блокировки учетных записей (Панель управления | Администрирование | Локальная политика безопасности | Политики учетных записей | Политика блокировки учетных записей);

5. Удаление пользователя и (или) группы:

- удалите созданную вами учетную запись и группу «Начинающие пользователи»

2.2. Выполните следующие действия при помощи командной строки:

Средства регистрации пользователей через net user:

- открыть и изучить дополнительную информацию по команде net help user;
- открыть список зарегистрированных пользователей, через командную строку;
- с помощью командной строки создать для себя учетную запись с следующими данными:
 - имя_пользователя - произвольное логическое имя;
 - полное_имя – свое ФИО
 - описание – указать свою группу
 - пароль - случайным образом сгенерировать пароль, состоящий из 8 символов, с помощью переключателя /Random
- через командную строку просмотреть свойства вновь зарегистрированного пользователя;
- изменить описание учетной записи на «группа, кафедра, институт», далее проверить внесенные изменения через свойства зарегистрированного пользователя;
- удалите созданную вами учетную запись.

2.3. Подготовить отчет (приложение 1)

3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Как может быть повышена надежность аутентификации с помощью паролей?
2. Какой может быть реакция системы на попытку подбора паролей?
3. Как должны храниться пароли в базе учетных записей пользователей?
4. В чем смысл объединения пользователей в группы?

ЛАБОРАТОРНАЯ РАБОТА 7. РЕАЛИЗАЦИЯ ПОЛИТИКИ БЕЗОПАСНОСТИ

Цель работы: приобрести навыки реализации политики безопасности в защищенных версиях ОС Windows

1. ТЕОРЕТИЧЕСКИЙ МАТЕРИАЛ

Параметры политики безопасности — это правила, которые администраторы настраивают на компьютере или нескольких устройствах с целью защиты ресурсов на устройстве или сети. Расширение *Параметры редактора локальной групповой политики* позволяет определять конфигурации безопасности как часть объекта групповой политики (GPO). GPOs связаны с контейнерами Active Directory, такими как сайты, домены или организационные подразделения, и они позволяют управлять настройками безопасности для нескольких устройств с любого устройства, подключенного к домену. Политики параметров безопасности используются в рамках общей реализации системы безопасности, чтобы обеспечить безопасность контроллеров доменов, серверов, клиентов и других ресурсов в организации.

Расширение *Параметры безопасности редактора локальной групповой политики* включает следующие типы политик безопасности:

– Политики учетных записей. Эти полицейские службы определяются на устройствах; они влияют на взаимодействие учетных записей пользователей с компьютером или доменом.

– Локальные политики. Эти политики применяются к компьютеру и включают следующие типы параметров политики:

- Политика аудита. Укажите параметры безопасности, которые контролируют ведение журнала событий безопасности в журнале безопасности на компьютере, и укажите, какие типы событий безопасности необходимо входить в журнал (успех, сбой или оба).
- Назначение прав пользователей. Укажите пользователей или группы, которые имеют права или привилегии логотипа на устройстве
- Параметры безопасности. Укажите параметры безопасности компьютера, такие как имена администратора и гостевой учетной записи; доступ к дискетным дискам и дискам CD-ROM; установка драйверов; подсказки logon; и так далее.

– Windows Брандмауэр с расширенным обеспечением безопасности. Укажите параметры для защиты устройства в сети с помощью государственного брандмауэра, который позволяет определить, какой сетевой трафик разрешен для прохода между устройством и сетью.

– Политики диспетчера списков сети. Укажите параметры, которые можно использовать для настройки различных аспектов перечисления и отображения сетей на одном устройстве или на многих устройствах.

– Политики общедоступных ключей. Укажите параметры для управления шифрованием файловой системы, защиты данных и шифрования диска BitLocker в дополнение к определенным путям сертификата и настройкам служб.

– Политики ограничения программного обеспечения. Укажите параметры для определения программного обеспечения и управления его возможностью запуска на локальном устройстве, организационном подразделении, домене или сайте.

– Политики управления приложениями. Укажите параметры, чтобы контролировать, какие пользователи или группы могут запускать определенные приложения в организации на основе уникальных удостоверений файлов.

– Политики безопасности IP на локальном компьютере. Укажите параметры для обеспечения конфиденциальной и безопасной связи через IP-сети с помощью служб криптографической безопасности. IPsec устанавливает доверие и безопасность с исходных IP-адресов на ip-адрес назначения.

– Расширенные настройки политики аудита. Укажите параметры, которые контролируют ведение журнала событий безопасности в журнале безопасности на устройстве. Параметры в статье Advanced Audit Policy Configuration обеспечивают более тонкий контроль за действиями, которые необходимо отслеживать, а не настройками политики аудита в соответствии с локальными политиками.

2. ЗАДАНИЕ

1. Средства определения политики безопасности:

- открыть окно определения параметров политики безопасности (Панель управления | Администрирование | Локальная политика безопасности | Локальные политики | Параметры безопасности);
- установить заголовок «ПРЕДУПРЕЖДЕНИЕ» в качестве значения параметра «Интерактивный вход в систему: заголовок сообщения для пользователей при входе в систему»;
- установить текст «На этом компьютере могут работать только зарегистрированные пользователи!» в качестве значения параметра «Интерактивный вход в систему: текст сообщения для пользователей при входе в систему»;
- установить значение «Отключен» для параметра «Интерактивный вход в систему: не требовать нажатия CTRL+ALT+DEL»;
- установить значение «Включен» для параметра «Интерактивный вход в систему: не отображать последнего имени пользователя»;
- установить значение «7 дней» для параметра «Интерактивный вход в систему: напоминать пользователям об истечении срока действия пароля заранее»;
- включить в отчет о лабораторной работе сведения о порядке назначения параметров политики безопасности, относящихся к интерактивному входу, и ответ на вопрос о смысле этих параметров;
- включить в отчет о лабораторной работе копии экранных форм, используемых при определении параметров политики безопасности, относящихся к интерактивному входу;

2. Средства определение политики аудита:

- открыть окно определения параметров политики аудита (Панель управления | Администрирование | Локальная политика безопасности | Локальные политики | Политика аудита) и включить в отчет о лабораторной работе ответ на вопрос, какие параметры политики аудита могут быть определены;
- с помощью параметров политики аудита установить регистрацию в журнале аудита успешных и неудачных попыток
 - ◆ входа в систему,
 - ◆ изменения политики,

- ◆ использования привилегий,
 - ◆ событий входа в систему,
 - ◆ управления учетными записями;
- открыть окно просмотра журнала аудита событий безопасности (Панель управления | Администрирование | Просмотр событий | Безопасность), выполнить команду «Свойства» контекстного меню (или команду Действие | Свойства) и включить в отчет о лабораторной работе ответы на вопросы
 - ◆ какие еще параметры политики аудита могут быть изменены,
 - ◆ где расположен журнал аудита событий безопасности;
 - включить в отчет о лабораторной работе сведения о порядке назначения параметров политики аудита и ответ на вопрос о смысле этих параметров;
 - включить в отчет о лабораторной работе копии экранных форм, используемых при определении параметров политики аудита.
3. Средства просмотра журнала аудита событий безопасности:
- открыть окно просмотра журнала аудита событий безопасности (Панель управления | Просмотр событий | Безопасность);
 - включить в отчет о лабораторной работе копии экранных форм с краткой и полной информацией о просматриваемом событии безопасности;
4. Средства определения политики ограниченного использования программ:
- открыть окно определения уровней безопасности политики ограниченного использования программ (Панель управления | Администрирование | Локальная политика безопасности | Политики ограниченного использования программ | Уровни безопасности);
 - включить в отчет о лабораторной работе пояснения к возможным уровням безопасности при запуске программ и копии соответствующих экранных форм;
 - открыть окно определения дополнительных правил политики ограниченного использования программ (Панель управления | Администрирование | Локальная политика безопасности | Политики ограниченного использования программ | Дополнительные правила);
 - включить в отчет о лабораторной работе ответы на вопросы, какие дополнительные правила для работы с программами могут быть определены (с помощью команд контекстного меню или меню «Действие») и в чем их смысл, а также копии соответствующих экранных форм.
5. Подготовить отчет (приложение 1)

3.КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие параметры определяют политику аудита?
2. Целесообразно ли с точки зрения безопасности компьютерной системы объединение в одном лице функций администратора и аудитора?
3. Как должен передаваться по сети (с точки зрения безопасности компьютерной системы) пароль пользователя (или другая аутентифицирующая информация)?
4. Нужно ли ограничивать права пользователей по запуску прикладных программ и почему?

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Системное администрирование персонального компьютера [Электронный ресурс] / Режим доступа: <https://ab57.ru/index.html> (дата обращения: 03.04.2022)
2. Техническая документация Microsoft Windows [Электронный ресурс] / Режим доступа: <https://learn.microsoft.com/ru-ru/docs/> (дата обращения: 05.04.2022)

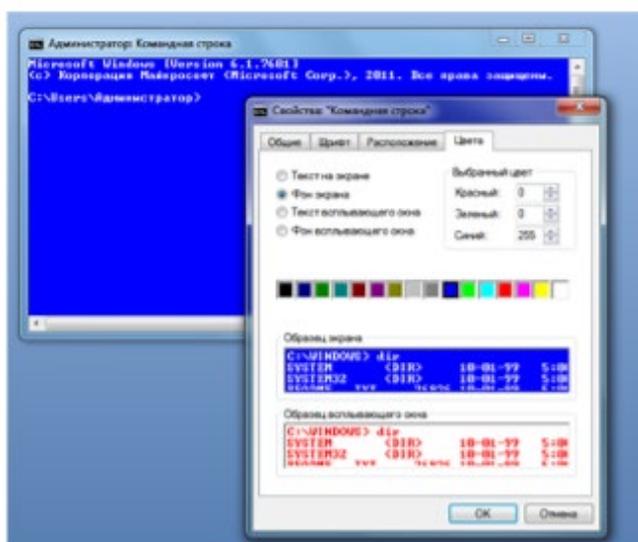
ПРИЛОЖЕНИЕ 1. ОБРАЗЕЦ ОФОРМЛЕНИЯ ОТЧЕТА

Отчет по лабораторной работе «Основы командной строки» по дисциплине «Операционные системы»

ФИО студента _____

Группа _____

1. Запустить окно командной оболочки cmd.exe и изменить общие настройки (вид окна, цвет фона и шрифта).



2. Сменить текущий диск на другой (предварительно проверив на компьютере название дополнительного диска);
3. Произвести очистку экрана;
4. Назначить заголовком окна для текущего сеанса интерпретатора командных строк CMD.EXE свою фамилию и группу;
5. Вывести на экран версию операционной системы, установленную в компьютере;
6. Вывести на экран информации о конфигурациях системы;
7. Используя оператор для разделения нескольких команд в одной командной строке, вывести на экран дату, а затем время, установленные в компьютере; при этом на запрос ввода новых параметров (даты и времени) нажмите Enter;
8. Проверить действие вышеперечисленных клавиш управления списком команд в следующем порядке: \uparrow , \downarrow , F7, F9, Alt + F7.