

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования
**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ**
Кафедра автоматизации обработки информации (АОИ)

Ю. Б. Гриценко

**Методические указания к выполнению
лабораторных работ по дисциплине
«Вычислительные системы, сети
и телекоммуникации»**

Для студентов направления подготовки Бизнес-информатика
(квалификация (степень) "бакалавр")

2015

Гриценко Ю. Б.

Методические указания к выполнению лабораторных работ по дисциплине «Вычислительные системы, сети и телекоммуникации» Для студентов направления подготовки Бизнес-информатика (квалификация (степень) "бакалавр"). — Томск: Факультет дистанционного обучения, ТУСУР, 2015. — 32 с.

СОДЕРЖАНИЕ

Введение.....	4
1 Интерфейс командной строки ОС Windows.....	5
2 Задание к лабораторной работе	26
3 Варианты заданий на выполнение.....	27
Список литературы	32

ВВЕДЕНИЕ

Настоящие методические указания соответствуют требованиям Федерального Государственного образовательного стандарта высшего профессионального образования (ФГОС ВПО) третьего поколения и содержанию рабочих программ по дисциплине «Вычислительные системы, сети и телекоммуникации».

Целью дисциплины «Вычислительные системы, сети и телекоммуникации» является формирование у студента профессиональных знаний по теоретическим основам построения и функционирования компьютеров вычислительных систем, телекоммуникационных вычислительных сетей и коммуникаций, их структурной и функциональной организации, программному обеспечению, эффективности и перспективам развития.

В рамках изучения дисциплины необходимо выполнить одну лабораторную работу.

Выбор варианта лабораторной работы осуществляется по общим правилам с использованием следующей формулы:

$$V = (N \times K) \operatorname{div} 100,$$

где V — искомый номер варианта,

N — общее количество вариантов,

div — целочисленное деление,

при $V=0$ выбирается максимальный вариант,

K — код варианта.

Варианты заданий представлены в разделе 3 методических указаний.

1 ИНТЕРФЕЙС КОМАНДНОЙ СТРОКИ ОС WINDOWS

Интерфейсы операционных систем можно разделить на два класса: графические интерфейсы пользователя (GUI) и **интерфейсы командной строки (CPI — Command Prompt Interface)**.

Графический интерфейс пользователя в Windows обеспечивается процессом Explorer и, как правило, хорошо знаком большинству пользователей персональных компьютеров. Графические интерфейсы систем на платформе Unix бывают различными, что связано с большим количеством версий, но работа в них мало чем отличается от работы в операционной системе Windows. Графический интерфейс в Linux чаще всего реализуется с помощью графических оболочек KDE и Gnome. Загрузка операционной системы Linux Red Hat заканчивается тем, что на экране появляется окно графической оболочки Gnome.

Эффективная профессиональная работа опытного пользователя с операционной системой компьютера немислима без овладения интерфейсом, обеспечиваемым командной строкой [1]. Преимуществом данного интерфейса служит возможность более гибко управлять ресурсами системы, чем с помощью графического интерфейса.

Интерфейс командной строки в ОС Windows присутствует, но играет для пользователей вспомогательную роль. В свое время он формировался как некое подмножество команд интерфейса Unix-подобных систем и особого развития не получил. Однако следует сказать, что интерфейс командной строки во многих нестандартных ситуациях остается единственным средством определения рассогласований и «тонкой настройки» аппаратно-программных средств. В последних версиях операционных систем Microsoft (Windows Server 2008) интерфейс командной строки получил даль-

нейшее развитие и превратился в мощный инструмент администрирования системы — оболочку Power Shell.

В новых версиях операционных систем Windows с учетом роста сложности аппаратной и программной частей компьютерных систем добавлен ряд команд, позволяющих решать задачи администрирования системы. Часть команд, заимствованных из MS DOS, получили дополнительные возможности. Например, такие команды, как `dir`, `copy`, `xcopy`, `rename` и др., в новых редакциях Windows могут работать с длинными именами файлов.

Включение режима командной строки может быть выполнено двумя способами (рис. 1.1):

1. Нажать на экране кнопку «Пуск» → «Выполнить», затем в появившемся окне набрать `cmd`.

2. Выбрать из главного меню: «Пуск» → «Программы» → «Стандартные» → «Командная строка».

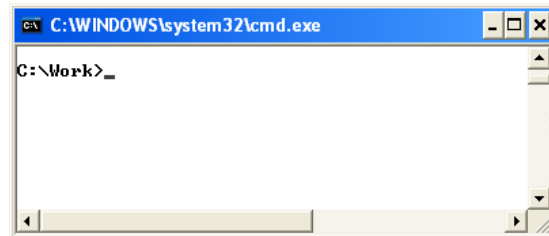


Рис. 1.1 — Окно «Командная строка»

Здесь указываются версия работающей операционной системы и строка приглашения с активным диском и рабочим каталогом (папкой). Выключение режима обеспечивается набором в строке приглашения команды `exit` и ее выполнением при нажатии клавиши `<Enter>`.

Смена текущего диска указывается путем указания его имени и двоеточия на конце. Например, чтобы перейти на диск D, необходимо указать `D:` и нажать клавишу `<Enter>`.

Перечень команд. В состав внутренних команд Windows входит около 70 команд. Перечень команд можно посмотреть с помощью команды HELP.

Поскольку весь перечень команд перекрывает размер экрана дисплея, то для ознакомления с каждым элементом перечня следует использовать полосу прокрутки окна или вызов на экран частей перечня постранично. Для этого следует набрать более сложную команду, состоящую из конвейера двух команд **HELP / MORE** (рис. 1.2).

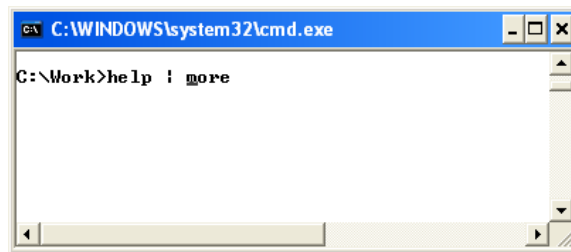


Рис. 1.2 — Ввод команд в командной строке

Список команд лучше рассмотреть по группам:

- команды справочной системы (табл. 1.1);
- команды файловой системы (табл. 1.2);
- команды управления работой ОС (табл. 1.3);
- команды пакетных (командных) файлов (табл. 1.4).

Таблица 1.1 — Команды справочной системы

HELP	Выводит справочную информацию о системе команд с версии Windows 2000
HELP имя_команды	Выводит справочную информацию о набранной команде
имя_команды /?	Выводит справочную информацию о набранной команде

Таблица 1.2 — Команды файловой системы

ATTRIB	Отображение и изменение атрибутов файлов
CD	Вывод имени либо смена текущей папки
CHDIR	
CHKDSK	Проверка диска и вывод статистики

Окончание табл. 1.2

COMP	Сравнение содержимого двух файлов или двух наборов файлов
COPY	Копирование одного или нескольких файлов в другое место
DEL	Удаление одного или нескольких файлов
DIR	Вывод списка файлов и подпапок из указанной папки
DISKCOMP	Сравнение содержимого двух гибких дисков
DISKCOPY	Копирование содержимого одного гибкого диска на другой
ERASE	Удаление одного или нескольких файлов
FC	Сравнение двух файлов или двух наборов файлов и вывод различий между ними
FIND	Поиск текстовой строки в одном или нескольких файлах
FINDSTR	Поиск строк в файлах
FORMAT	Форматирование диска
LABEL	Создание, изменение и удаление меток тома для дисков
MD	Создание папки
MKDIR	
MOVE	Перемещение одного или нескольких файлов из одной папки в другую
PUSHD	Сохранение значения текущей активной папки и переход к другой папке
POPD	Восстановление предыдущего значения текущей активной папки, сохраненного с помощью команды PUSHD
PRINT	Вывод на печать содержимого текстовых файлов
RD	Удаление папки
REN	Переименование файлов и папок
RENAME	
REPLACE	Замещение файлов
RMDIR	Удаление папки
SORT	Сортировка ввода
TREE	Графическое отображение структуры папок заданного диска или заданной папки
TYPE	Вывод на экран содержимого текстовых файлов
VERIFY	Установка режима проверки записи файлов на диск
VOL	Вывод метки и серийного номера тома для диска
XCOPY	Копирование файлов и дерева папок

Таблица 1.3 — Команды управления работой ОС

ASSOC	Вывод или изменение связи между расширениями имени и типами файлов
AT	Выполнение команд и запуск программ по расписанию
BAEAK	Включение/выключение режима обработки комбинации клавиш CTRL+C
CACLS	Отображение/редактирование списков управления доступом к файлам различных пользователей
CHCP	Просмотр номера текущей кодовой страницы или изменение текущей кодовой страницы консоли
CHKNTFS	Отображение или изменение выполнения проверки диска во время загрузки

Окончание табл. 1.3

CLS	Очистка экрана на консоли
CMD	Запуск еще одного интерпретатора командных строк
COLOR	Установка цвета текста и фона, используемых по умолчанию
COMPACT	Отображение/изменение сжатия файлов в разделах NTFS
CONVERT	Преобразование дисковых томов FAT в NTFS
DATE	Вывод либо установка текущей даты
DOSKEY	Редактирование и повторный вызов командных строк. Создание макросов
FTYPE	Вывод либо изменение типов файлов, используемых при сопоставлении по расширениям имен файлов
GRAFTABL	Позволяет отображать расширенный набор символов в графическом режиме
MODE	Конфигурирование системных устройств
MORE	Последовательный вывод данных по частям размером в один экран
PATH	Вывод либо установка пути поиска исполняемых файлов
PROMPT	Изменение приглашения в командной строке
RECOVER	Восстановление читаемой информации с плохого или поврежденного диска
SET	Вывод, установка и удаление переменных среды
START	Запуск программы или команды в отдельном окне
SUBST	Сопоставляет заданному пути имя диска
VER	Вывод сведений о версии операционной системы

Таблица 1.4 — Команды пакетных (командных) файлов

CALL	Вызов одного пакетного файла из другого
ECHO	Вывод сообщений и переключение режима отображения команд на экране
ENDLOCAL	Конец локальных изменений среды для пакетного файла
EXIT	Завершение работы программы
FOR	Организация циклов для обработки наборов файлов или строк в файле
GOTO	Передача управления в отмеченную строку пакетного файла
IF	Оператор условного выполнения команд в пакетном файле
PAUSE	Приостановка выполнения пакетного файла и вывод сообщения
REM	Помещение комментариев в пакетные файлы
SETLOCAL	Начало локальных изменений среды для пакетного файла
SHIFT	Изменение содержимого (сдвиг) замещаемых параметров для пакетного файла

Кроме команд, перечисленных в таблицах, имеется еще одна группа для работы в компьютерных сетях. Перечень этих команд может быть получен командой *NET /?* (рис. 1.3) (табл. 1.5).

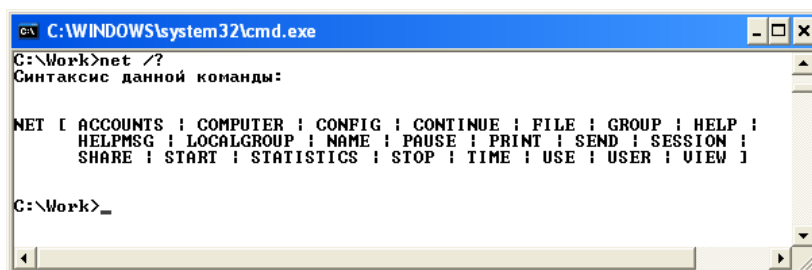


Рис. 1.3 — Вызов помощи по команде /?

Таблица 1.5 — Сетевые команды

NET ACCOUNTS	Обновление учетной базы пользователей, паролей и параметров подключения
NET COMPUTER	Добавление и удаление имени компьютера в базе данных домена
NET CONFIG	Сведения о настраиваемых службах и их изменение
NET CONTINUE	Активизация приостановленной службы, имя которой указано в качестве параметра
NET FILE	Вывод имен открытых файлов на сервере и количества их блокировок
NET GROUP	Вывод, добавление и изменение глобальных групп на сервере домена
NET HELPMMSG	Выдача справок об ошибках и предупреждающих сообщениях
NET LOCALGROUP	Отображение и изменение локальных групп
NET NAME	Добавление и удаление имени, называемого псевдонимом. Псевдоним — имя компьютера, принимающего сообщения
NET PAUSE	Приостановка работы службы, указанной параметром в команде
NET PRINT	Отражение состояния, управление заданиями и очередями принтеров
NET SEND	Пересылка сообщения адресату: пользователю, компьютеру, псевдониму
NET SESSION	Вывод списка подключенных к компьютеру пользователей и его изменение
NET SHARE	Создание и удаление совместно используемых ресурсов сети
NET START	Вывод списка запущенных служб и его изменение
NET STATISTICS	Вывод содержимого журнала статистики для служб компьютера или сервера
NET STOP	Остановка работы службы, указанной параметром в команде
NET TIME	Синхронизация часов компьютеров, включенных в сеть
NET USE	Подключение компьютеров сети к сетевым ресурсам
NET USER	Добавление, редактирование и просмотр учетных сведений пользователей
NET VIEW	Просмотр списков компьютеров, доменов и общих ресурсов на указанном компьютере

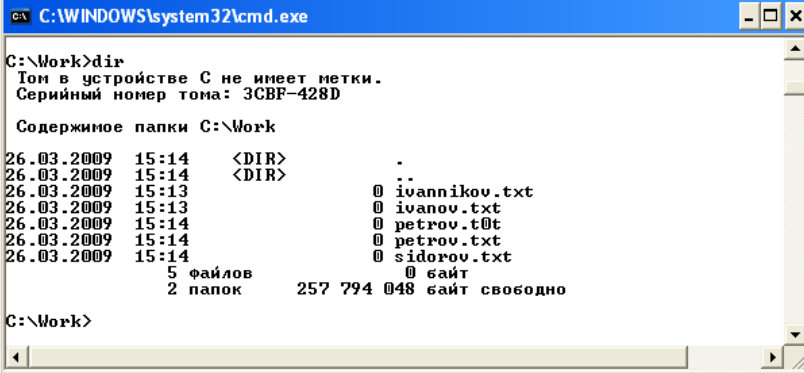
Для вызова помощи для конкретных сетевых команд следует набирать `net имя_команды /?`.

Справочная информация по различным командам свидетельствует, что командой, набираемой в командной строке, является собственно имя команды, за которым могут следовать ключи (опции) — указания, модифицирующие поведение команды. Квадратные скобки в пояснениях обозначают, что эта информация не является обязательной при наборе команды. Ключи начинаются со знака / (слэша) и состоят из одного или нескольких символов. Кроме ключей, после команды могут следовать аргументы (параметры) — названия объектов, над которыми должна быть выполнена команда. Очень часто аргументами служат имена файлов и каталогов.

Ввод команды заканчивается нажатием клавиши <Enter>, после чего команда передается на исполнение командному процессору. В результате выполнения команды на экране дисплея могут появиться сообщения о ходе выполнения команды или об ошибках, а появление очередного приглашения (мигающего курсора) свидетельствует об успешном выполнении введенной команды и ожидании ввода следующей [1].

Замещаемые символы (метасимволы). Параметр в командной строке команды может включать замещаемые символы «?» и «*». Символ вопросительный знак заменяет один любой символ. Символ звездочка может заменять любую последовательность символов.

Пусть в текущем каталоге содержится произвольный набор файлов. Команда `DIR` без параметров по умолчанию покажет нам весь перечень файлов в директории (рис. 1.4).



```

C:\WINDOWS\system32\cmd.exe
C:\Work>dir
Том в устройстве C не имеет метки.
Серийный номер тома: 3CBF-428D

Содержимое папки C:\Work

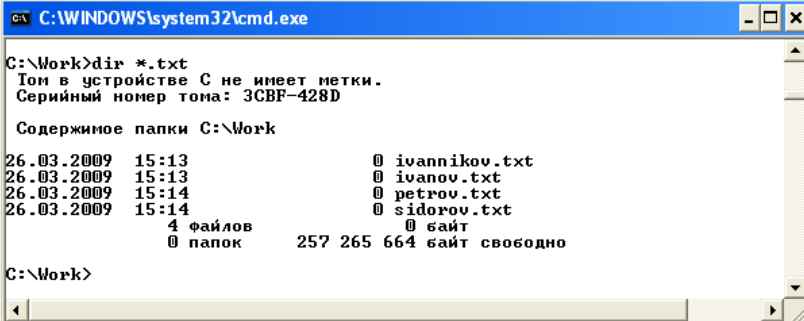
26.03.2009  15:14    <DIR>          .
26.03.2009  15:14    <DIR>          ..
26.03.2009  15:13                0 ivannikov.txt
26.03.2009  15:13                0 ivanov.txt
26.03.2009  15:14                0 petrov.t0t
26.03.2009  15:14                0 petrov.txt
26.03.2009  15:14                0 sidorov.txt
                5 файлов          0 байт
                2 папок          257 794 048 байт свободно

C:\Work>

```

Рис. 1.4 — Результат выполнения команды DIR

Применим для вывода команды замещаемые параметры. Результат выполнения команды DIR *.TXT будет следующим (рис. 1.5). Строка с файлом PETROV.T0T не будет отображена, так как мы указали команде DIR показать все файлы с любым именем (символ *), но имеющим только расширение .TXT.



```

C:\WINDOWS\system32\cmd.exe
C:\Work>dir *.txt
Том в устройстве C не имеет метки.
Серийный номер тома: 3CBF-428D

Содержимое папки C:\Work

26.03.2009  15:13                0 ivannikov.txt
26.03.2009  15:13                0 ivanov.txt
26.03.2009  15:14                0 petrov.txt
26.03.2009  15:14                0 sidorov.txt
                4 файлов          0 байт
                0 папок          257 265 664 байт свободно

C:\Work>

```

Рис. 1.5 — Результат выполнения команды DIR *.TXT

Выполним последовательно команды:

1. DIR PETROV.T* (рис. 1.6).
2. DIR PETROV.T? (рис. 1.7).
3. DIR PETROV.T?? (рис. 1.8).

```

C:\WINDOWS\system32\cmd.exe
C:\Work>dir petrov.t*
Том в устройстве C не имеет метки.
Серийный номер тома: 3CBF-428D

Содержимое папки C:\Work
26.03.2009  15:14                0 petrov.t0t
26.03.2009  15:14                0 petrov.txt
                2 файлов                0 байт
                0 папок          256 532 480 байт свободно

C:\Work>_

```

Рис. 1.6 — Результат выполнения команды DIR PETROV.T*

Первая команда говорит, что необходимо показать файлы, у которых имя «petrov», расширение начинается с символа «t», далее могут идти любые символы.

Вторая команда говорит, что необходимо показать файлы, у которых имя «petrov», расширение начинается с символа «t», далее может быть только один любой символ.

Третья команда говорит, что необходимо показать файлы, у которых имя «petrov», расширение начинается с символа «t», далее может быть только два любых символа.

```

C:\WINDOWS\system32\cmd.exe
C:\Work>dir petrov.t?
Том в устройстве C не имеет метки.
Серийный номер тома: 3CBF-428D

Содержимое папки C:\Work
Файл не найден

C:\Work>

```

Рис. 1.7 — Результат выполнения команды DIR PETROV.T?

```

C:\WINDOWS\system32\cmd.exe
C:\Work>dir petrov.t??
Том в устройстве C не имеет метки.
Серийный номер тома: 3CBF-428D

Содержимое папки C:\Work
26.03.2009  15:14          0 petrov.t0t
26.03.2009  15:14          0 petrov.txt
              2 файлов              0 байт
              0 папок            253 173 760 байт свободно

C:\Work>_

```

Рис. 1.8 — Результат выполнения команды DIR PETROV.T??

Как видно из рисунков, результаты выполнения первой и третьей команд совпали, так как расширение состоит из трех символов.

А результат выполнения второй команды не вывел ни одного файла, удовлетворяющего заданному условию.

Стандартные потоки ввода-вывода и перенаправление потоков.

Термин CONsole используется для обозначения стандартных потоков ввода-вывода. Когда говорят о вводе с консоли, подразумевается ввод с клавиатуры. Когда говорят о выводе на консоль, подразумевают вывод на экран монитора. Существуют специальные символы для перенаправления стандартных потоков ввода-вывода:

> *приемник* — перенаправить стандартный вывод в приемник (если файл-приемник существует, то он будет создан заново);

>> *приемник* — перенаправить стандартный вывод в приемник (если файл-приемник существует, то он будет сохранен, а информация будет записана в конец файла);

< *источник* — перенаправить стандартный ввод из источника;

передатчик | *приемник* — передает вывод одной команды на вход другой.

Приведем несколько примеров с использованием перенаправления потоков:

`DIR > FILES.TXT` — содержимое текущего каталога записать в текстовый файл;

`TYPE FILE.TXT >> ARXIV.TXT` — добавить в конец файла `ARXIV.TXT` содержимое файла `FILE.TXT`;

`DATE < DATE.TXT` — установить новую системную дату, значение взять из файла `DATE.TXT`;

`TYPE PETROV.TXT | SORT` — распечатывает на экране файл `petrov.txt`, отсортировав его строки.

Возможна комбинация символов перенаправления потоков:

`TYPE PETROV.TXT | SORT > PETROV_SORT.TXT` — записывает в файл `PETROV_SORT.TXT` содержимое файла `PETROV.TXT`, отсортировав его строки.

Создание текстовых файлов можно выполнить следующей командой:

`COPY CON ДИСК:ПУТЬ\ИМЯ_ФАЙЛА.РАСШ`

Например, для создания файла `F1.TXT` в текущем каталоге необходимо выполнить команду `COPY CON F1.TXT`.

Этой командой ввод текста с клавиатуры (консоли `con`) осуществляется в новый, создаваемый этой операцией файл `F1.TXT`. После набора каждой строки следует нажимать клавишу `<Enter>`. Окончание набора файла должно заканчиваться нажатием клавиш `<Ctrl>+Z` или `F6` (знаком окончания файла), а затем `<Enter>`. Недостатком применения этой команды является то, что редактировать можно только текущую строку файла. После нажатия клавиши `<Enter>` ранее введенные строки уже недоступны. Таким образом, команда `copy` использует простейший однострочный редактор.

Атрибуты файлов. Каждый файл и каталог, находящиеся в компьютере, могут иметь атрибуты — характеристики, отражающие свойства объ-

екта, которые используются операционной системой для корректной работы с ними. Атрибутами файла (файлов) могут быть следующие значения:

R — «только для чтения», то есть нельзя модифицировать файл и уничтожить его;

A — «архивный», т. е. +A обозначает, что снимались копии данного файла; соответственно, -A — файл является вновь созданным;

H — «скрытый», скрытые файлы не показываются командой DIR и некоторыми программными оболочками;

S — «системный», этот атрибут показывает, что файл является принадлежностью операционной системы.

Изменение атрибутов файлов осуществляется командой ATTRIB. Формат команды

ATTRIB +|-АТРИБУТ ДИСК:\МАРШРУТ\ИМЯ ФАЙЛА /S

Установка любого атрибута производится знаком + (плюс), отмена — знаком - (минус). Можно задавать изменение сразу нескольких атрибутов в любой последовательности. Ключ /S, стоящий в конце формата, указывает, что процесс изменения атрибутов файла (файлов) распространяется не только на текущий каталог, но и на все каталоги, подчиненные текущему.

Примеры:

ATTRIB +A +H +R PRIMER.TXT — присваивает файлу сразу три атрибута: только для чтения, архивный и скрытый файлу PRIMER.TXT.

ATTRIB -R A:*.* /S — с использованием шаблона снимает атрибут «только для чтения» у всех файлов диска A, т. е. находящихся на дискете.

Разработка командных файлов. Командный файл — это группа последовательных команд настройки компьютера на определенный режим или выполнение определенных операций. В простейшем случае командный файл может быть представлен в виде определенной последовательности отдельных команд операционной системы. Разработка командных

файлов является мощным средством автоматизации подготовительных работ пользователей по настройке среды их работы [1].

При разработке командных файлов следует руководствоваться следующими правилами [1]:

1. Вызов на исполнение командного файла осуществляется командой следующего формата:

ДИСК:\ПОЛНЫЙ_ПУТЬ\ИМЯ_КОМ.ФАЙЛА [P1 P2 ... P10]

Содержимое в квадратных скобках указывает, что командный файл может иметь до 10 фактических параметров, замещающих формальные параметры, присутствующие в тексте файла. Команда SHIFT позволяет снять это ограничение.

2. Имя командного файла образуется по обычным правилам. Расширением должно быть только сочетание BAT или CMD.

3. Если текущим является каталог (папка), содержащий командный файл, то полный путь к командному файлу можно не указывать.

4. Командный файл выполняется командным процессором строка за строкой.

5. Выполнение командного файла может быть прекращено командами <Ctrl>+<Break> или <Ctrl>+C.

6. Из командного файла можно вызывать другой командный файл командой CALL (с возвратом) или обычной командой вызова (без возврата).

7. Командный файл может содержать любые внешние и внутренние команды операционной системы, а также специальные внутренние команды.

8. Формальные параметры, включаемые в строки командного файла, имеют вид %0, %1 и т. д. до %9. Фактические значения параметров вводятся в строке вызова командного файла; вводимые параметры подставляются на место формальных параметров %1, %2 и т. д. по порядку. На место

формального параметра %0, если он встречается в тексте командного файла, подставляется имя самого командного файла.

9. Для обращения к переменным окружения их имена следует заключать в знаки %, например %ТЕХТ%.

10. Перед выполнением очередной строки командного файла ее значение выводится на экран. Вывод любой строки командного файла на экран подавляется, если строка начинается с символа @.

Рассмотрим *особенности применения специальных команд*:

Команда ECHO предназначена для отключения «эха» на экране дисплея, то есть она не позволяет выводить лишнюю информацию на монитор (блокирует выдачу на экран последовательностей команд, включенных в командный файл, и текстовых сообщений при выполнении этих команд). Форматы команды:

ECHO OFF — запрет вывода на экран;

ECHO ON — разрешение вывода на экран;

ECHO (без параметров) — запрос состояния эха (ON или OFF);

ECHO + текстовое сообщение — вывод текстового сообщения на экран;

ECHO %имя переменной окружения% — вывод текущего значения переменной окружения;

ECHO. (с точкой) — вывод пустой строки.

При использовании команды ECHO следует помнить:

– при запуске системы по умолчанию устанавливается режим «ECHO ON»;

– режим «ECHO OFF» действует только до конца командного файла или до очередного переключения режима командой «ECHO ON»;

– ECHO влияет только на вывод сообщений командного файла, но не влияет на вывод сообщений из программ пользователей, даже если они используют команды операционной системы;

– для подавления самой команды «ECHO OFF» надо поставить впереди знак @.

Для лучшего понимания содержимого командного файла используются комментарии, вводимые с помощью *команды REM* (remark — примечание). Командный процессор полностью игнорирует всю информацию, которая размещается за словом REM. Команда очень полезна, когда в командный файл включаются пояснения, описания работы файла или отдельных его команд, тестирования и отладки.

Для приостановки выполнения командного файла используется *команда PAUSE*. Команда имеет формат:

PAUSE сообщение

При остановке работы командного файла на экране появляется текст строки сообщения в режиме ECHO ON, а под ним фраза «Press any key to continue» — Нажмите любую клавишу для продолжения (для продолжения работы файла).

Команду полезно использовать в тех случаях, когда, например, на экран дисплея выводится большое количество информации порциями по страницам, чтобы пользователь мог ее прочитать, осмыслить и перейти к следующему фрагменту. Команда полезна и в случаях, когда необходимо выполнить какие-то вспомогательные действия, например:

@ECHO ON

PAUSE Установите дискету с на дисковод A:

@ECHO OFF

Кроме того, команду PAUSE можно использовать и для управления работой командного файла. Если в ответ на команду PAUSE нажать <Ctrl>+C, то появляется вопрос «Terminate batch job (Y/N)?» — Завершить выполнение задания (командного файла)? Выбор Y — останавливает выполнение командного файла, а N — обеспечивает продолжение его выполнения. В случаях когда командный файл выполняется с частыми остановками и появление множества фраз «Press any key to continue» нежелательно, строка с командой может выглядеть как PAUSE> NUL, то есть вывод переадресуется в несуществующее устройство nul.

Команда GOTO позволяет изменить привычную последовательность выполнения операторов (команд) командного файла. Когда командный процессор встречает строку с оператором GOTO, то он просматривает все строки файла и отыскивает соответствующую метку — строку с двоеточием. Двоеточие может быть и не в первой позиции строки. Идентификатор метки должен иметь до восьми символов. Больше, чем восемь символов, в идентификаторе не воспринимается. Команда GOTO может использоваться самостоятельно или совместно с операцией IF.

Команда IF — условное выполнение команд, организует разветвление при выполнении командного файла. Формат оператора IF

IF условие команда.

В качестве условия обычно используются:

- проверка наличия файла. В этом случае в качестве условия записывается фраза exist диск:путь\имя_файла.расш;
- проверка кода завершения отдельных программ по значению внутренней переменной системы с именем ERRORLEVEL. В этом случае в качестве условия записывается фраза «ERRORLEVEL значение». Условие

считается истинным, если код завершения равен или больше параметра значение. Значение переменной `errorlevel` может формироваться многими утилитами и прикладными программами;

– проверка идентичности двух символьных строк. Строка условие при этом записывается в виде

`строка_1==строка_2` (двойной знак =)

Предваряя любому из перечисленных условий слово `NOT`, можно проверять противоположное условие.

Для многократного выполнения отдельных команд применяется команда `FOR`. Она позволяет обрабатывать целые группы файлов. Команда имеет следующие форматы:

`FOR %%переменная IN (набор) DO команда` — для строк командных файлов;

`FOR %переменная IN (набор) DO команда` — для режима командной строки (автономного выполнения команды).

В качестве параметров команды используются:

– `переменная` — однобуквенная переменная, последовательно принимающая значения слов или имен файлов, перечисленных в параметре (набор);

– (набор) — одно или несколько символьных слов или спецификаций файлов. Спецификация файла имеет вид `диск:путь\имя_файла.расш`. Допускаются шаблоны групповых операций. Слова и спецификации файлов разделяются пробелами или запятыми. Максимальная длина строки набора — не более 127 символов;

– `команда` — команда `DOS`, выполняемая для каждого слова или файла из параметра набор.

Обычно в командный файл можно передавать до 9 параметров, иногда этого недостаточно. *Команда SHIFT* (сдвиг) позволяет сдвигать строку параметров влево на один параметр. Применение данной команды позволяет снять ограничение на число параметров.

В учебном пособии приведены лишь фрагменты описания команды, более подробную информацию по командам лучше смотреть непосредственно, вызывая помощь в командной строке операционной системы.

Команда CHOICE — ожидает ответа пользователя. Данная команда является внешней, то есть необходимо наличие файла choice.exe. Данный файл должен располагаться в текущей папке или в какой-либо системной, чтобы операционная система могла найти его. Формат команды выглядит следующим образом:

`CHOICE [/C[:]варианты] [/N] [/S] [/T[:]с,nn] [текст]`

`/C[:]варианты` — варианты ответа пользователя.

По умолчанию строка включает два варианта: YN

`/N` Ни сами варианты, ни знак вопроса в строке приглашения не отображаются.

`/S` Учитывать регистр символов.

`/T[:]с,nn` Ответ «с» выбирается автоматически после nn секунд ожидания текста Строка приглашения

После выполнения команды переменная `ERRORLEVEL` приобретает значение, равное номеру выбранного варианта ответа.

Приведем *несколько примеров командных файлов.*

Пример 1 [1]. Пусть требуется создать командный файл `test1.bat`, который будет копировать из текущего каталога на дискету ряд текстовых файлов с проверкой правильности записи и удалением исходных файлов.

Перед каждым удалением файла должно выдаваться предупреждающее сообщение. В момент приостановки можно прервать дальнейшее выполнение командного файла, нажав клавиши <Ctrl>+C.

```
:LOOP
COPY %1.TXT A:/V
PAUSE УДАЛЯЮ СКОПИРОВАННЫЙ ФАЙЛ
DEL %1.TXT
SHIFT
IF NOT %1.==. GOTO LOOP
```

Запуск этого файла следует выполнять командой

Test1.bat 01 02 03 04 05 06 07 08 09 10 11 12 и т. д., если текстовые файлы имеют имена 01.txt, 02.txt, 03.txt и т. д. Обратите внимание, что:

- по умолчанию здесь используется режим ECHO ON. В противном случае сообщения команды PAUSE были бы не видны;
- расширения текстовых файлов присоединяются к имени непосредственно в командах выполняемого файла.

Пример 2 [1]. Создадим файл test2.bat таким образом, чтобы можно было отыскивать и просматривать нужный файл в любом каталоге. Учитывая, что файлы могут иметь большие размеры, превышающие емкость одного экрана, обеспечим поэкранный просмотр файлов. Имя нужного файла будем задавать в качестве параметра в строке вызова файла test2.bat. Например: test2.bat proba.txt

Здесь имя искомого файла proba.txt служит фактическим параметром, значение которого должно заменить формальные параметры %1 внут-

ри командного файла. Таких параметров строка вызова может иметь от %1 до %9. Командный файл test2.bat будет иметь следующее содержание.

```
ECHO OFF
CLS
IF /%1==/ GOTO ERROR1
IF NOT EXIST %1 GOTO ERROR2
TYPE %1 | MORE
GOTO END
:ERROR1
ECHO ВЫ забыли указать имя искомого файла!
:GOTO END
:ERROR2
ECHO ФАЙЛА %1 на этом диске нет!
:END
```

В этом фрагменте два слэша / в операторе IF играют роль скобок.

Пример 3. Командный файл просит ввести цифру, соответствующую имени пользователя, и в зависимости от того, какая цифра была введена, устанавливает текущим тот или иной рабочий каталог и открывает окно CMD.

```
:BEGIN
    ECHO Введите номер пользователя
    ECHO 1 – Алексей 2 – Петр 3 – Иван 4 – остальные
    CHOICE /C:1234
    IF ERRORLEVEL 4 GOTO WORK
    IF ERRORLEVEL 3 GOTO IVAN
```



```
IF ERRORLEVEL 2 GOTO PETER
IF ERRORLEVEL 1 GOTO ALEX
GOTO BEGIN
```

```
:IVAN
```

```
    CD IVAN
```

```
    START
```

```
:PETER
```

```
    CD PETER
```

```
    START
```

```
:ALEX
```

```
    CD ALEX
```

```
    START
```

```
:WORK
```

```
    CD WORK
```

2 ЗАДАНИЕ К ЛАБОРАТОРНОЙ РАБОТЕ

Лабораторная работа по курсу «Вычислительные системы, сети телекоммуникации» состоит из двух частей: практическая часть и теоретическая часть.

В *практической части* студенты должны разработать командные файлы.

При разработке необходимо учесть возможность обработки различных ошибок ввода данных, например, неправильного запуска ваших программ (с недостаточным количеством параметров или с неправильными параметрами), и предусмотреть вывод сообщения об ошибке и подсказки.

Программа должна быть реализована в виде командных файлов ОС Windows. Файлы должны быть самостоятельными, а не в тексте отчета (документе Microsoft Word). Каждая строчка командного файла должна сопровождаться подробными комментариями.

Теоретическая часть содержит ряд вопросов по учебному пособию, на которые студент должен дать развернутый и аргументированный ответ.

3 ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ

Вариант 1

Практическая часть

Разработать командный файл, создающий, копирующий или удаляющий файл, указанный в параметре строки при запуске командного файла, в зависимости от выбранного ключа /n , /c , /d.

Теоретическая часть

1. Дайте определение термину «Система». Какие характеристики используют для описания систем?
2. Опишите основные характеристики накопителей на жестком магнитном диске.

Вариант 2

Практическая часть

Разработать командный файл, который бы проверял событие: «Запускали сегодня его уже или нет». Если файл уже запускали, то выйти из программы, если нет, то файл должен запустить какой-либо текстовый редактор. Для определения события выполните сравнение дат (последнего запуска и текущей) через переменные, а не через файлы. Вам поможет системная переменная %DATE% и команда SET (под ОС Windows).

Теоретическая часть

1. Что понимают под определением «Вычислительная система»?
2. Чем отличается физическая организация магнитного диска от логической?

Вариант 3

Практическая часть

Разработать командный файл, добавляющий вводом с клавиатуры содержимое текстового файла (в начало или в конец в зависимости от ключей /b /e). Используйте команды перенаправления потоков ввода-вывода.

Теоретическая часть

1. Расскажите историю развития вычислительной техники.
2. Приведите достоинства и недостатки различных файловых систем.

Вариант 4

Практическая часть

Разработать командный файл, который бы проверял событие: «Запускали сегодня его уже или нет». Если файл уже запускали, то выйти из программы, если нет, то файл должен запустить какой-либо текстовый редактор. Для определения события сохраните текущую дату и дату последнего запуска в файлы и выполните сравнение файлов командой FC. Результат сравнения можно определить, используя команду IF ERRORLEVEL (под ОС Windows).

Теоретическая часть

1. Опишите электронные вычислительные машины и приведите их классификацию.
2. Как в ОС Windows на платформе NT можно управлять дисками и файловыми системами?

Вариант 5

Практическая часть

Разработать командный файл, регистрирующий время своего запуска в файле протокола gun.log и автоматически запускающий некоторую программу (например, антивирусную и т. п.) по пятницам или 13 числам. Определение даты запуска можно выполнить одним из двух возможных способов: использовать сравнение переменных (вырезать подстроку из системной переменной %DATE% и сравнить с числом. См. команду SET); сохранить текущую дату в файл и выполнить в нем поиск.

Теоретическая часть

1. Сравните определения архитектуры вычислительной системы и архитектуры электронной вычислительной машины.
2. Какие преимущества в ОС Windows на платформе NT дает использование файловой системы NTFS пятой версии?

Вариант 6

Практическая часть

Разработать командный файл, который дописывал бы имя файла, полученного входным параметром в сам файл N количество раз. N — также задается параметром.

Теоретическая часть

1. В чем отличие принстонской архитектуры ЭВМ от гарвардской архитектуры ЭВМ?
2. Назовите последовательно уровни сетевой модели OSI.

Вариант 7

Практическая часть

Разработать командный файл, который в интерактивном режиме (командный файл «задает вопросы», а пользователь на них отвечает) мог бы дописывать в файл текст, удалять строки из файла и распечатывать на экране содержимое файла. Удаление строк можно реализовать либо через команду поиска строк в файле, либо через команду организации циклов FOR.

Теоретическая часть

1. Расскажите, какие свойства ЭВМ относятся к общим, а какие к индивидуальным свойствам?
2. Что входит в состав физической инфраструктуры сети?

Вариант 8

Практическая часть

Разработать командный файл, который бы получал в качестве параметра какое-либо имя и проверял, определена ли такая переменная среды или нет, и выводил соответствующее сообщение.

Теоретическая часть

1. Какие свойства имеет CISC-архитектура процессора?
2. В чем состоит отличие локальной вычислительной сети от глобальной компьютерной сети?

Вариант 9

Практическая часть

В некотором файле хранится список пользователей ПК и имена их домашних каталогов. Каждый пользователь и имя его каталога в отдельной строке. Необходимо разработать программу, которая просматривает данный файл и в интерактивном режиме (командный файл «задает вопросы», а пользователь на них отвечает) задает вопрос — копировать текущему пользователю (в его домашний каталог) какой-либо заданный файл в качестве параметра или нет. Если «Да» то программа копирует файл.

Теоретическая часть

1. Какие свойства имеет RISC-архитектура процессора?
2. Что такое сеть периметра?

Вариант 10

Практическая часть

Разработать командный файл, который помещает список файлов текущего каталога в текстовый файл и в зависимости от ключа сортирует по какому-либо полю. Реализовать два варианта: с использованием только команды DIR, с использованием команд DIR и SORT.

Теоретическая часть

1. Опишите основные «вехи» развития микропроцессоров семейства x86-64.
2. В чем состоит отличие технологии VPN от Dial-up сервиса?

СПИСОК ЛИТЕРАТУРЫ

1 Назаров С. В. Операционные системы : практикум / С. В. Назаров, Л. П. Гудыно, А. А. Кириченко; под ред. С. В. Назарова — М.: КУДИЦ-ПРЕСС, 2008. — 464 с. : ил.