

Министерство образования и науки Челябинской области
Государственное бюджетное профессиональное образовательное учреждение
«Южно-Уральский государственный технический колледж»

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ
по выполнению практических работ
по учебной дисциплине
«ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»
(1 семестр)

для студентов специальности
09.02.06 Сетевое и системное администрирование
Квалификация: Сетевой и системный администратор

Челябинск, 2022

СОДЕРЖАНИЕ

| | |
|----------------------------------|--|
| СОДЕРЖАНИЕ | 2 |
| ПОЯСНИТЕЛЬНАЯ ЗАПИСКА..... | 3 |
| ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ РАБОТ..... | 5 |
| Практическая работа №1 | 7 |
| Практическая работа №2 | 27 |
| Практическая работа №4 | Ошибка! Закладка не определена. |
| Практическая работа №5 | Ошибка! Закладка не определена. |
| Практическая работа №6 | Ошибка! Закладка не определена. |
| Практическая работа №7 | Ошибка! Закладка не определена. |
| Практическая работа №8 | Ошибка! Закладка не определена. |
| Практическая работа №9 | Ошибка! Закладка не определена. |
| Практическая работа №10 | Ошибка! Закладка не определена. |
| Практическая работа №11 | Ошибка! Закладка не определена. |
| Практическая работа №12 | Ошибка! Закладка не определена. |
| Практическая работа №13 | Ошибка! Закладка не определена. |
| Практическая работа №14 | Ошибка! Закладка не определена. |
| Практическая работа №15 | Ошибка! Закладка не определена. |
| Практическая работа №16 | Ошибка! Закладка не определена. |
| Практическая работа №17 | Ошибка! Закладка не определена. |
| Информационные источники..... | Ошибка! Закладка не определена. |
| Приложение 1 | 27 |
| Приложение 2 | 28 |
| Приложение 3 | 29 |

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Методические рекомендации по выполнению практических работ по учебной дисциплине «Основы теории информации» предназначены для студентов специальности 09.02.06 Сетевое и системное администрирование, квалификация – Сетевой и системный администратор (перечень ТОП-50).

Практические занятия являются важным элементом учебной дисциплины. В процессе выполнения практических работ, обучающиеся систематизируют и закрепляют полученные теоретические знания, развивают интеллектуальные и профессиональные умения, формируют элементы общих и профессиональных компетенций.

Программой учебной дисциплины «Информационные технологии» предусмотрено выполнение 35 практических работ в 3-ем семестре, направленных **на формирование элементов следующих компетенций:**

ОК 1. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 2. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 4. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 5. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 9. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке

ПК 3.1. Устанавливать, настраивать, эксплуатировать и обслуживать технические и программно-аппаратные средства компьютерных сетей.

ПК 3.5. Организовывать инвентаризацию технических средств сетевой инфраструктуры, осуществлять контроль оборудования после его ремонта.

ПК 3.6. Выполнять замену расходных материалов и мелкий ремонт периферийного оборудования, определять устаревшее оборудование и программные средства сетевой инфраструктуры.

ПК 5.2. Разрабатывать предложения по совершенствованию и повышению эффективности работы сетевой инфраструктуры.

умений:

- обрабатывать текстовую и числовую информацию;
- применять мультимедийные технологии обработки и представления информации;
- обрабатывать экономическую и статистическую информацию, используя средства пакета прикладных программ.

обобщение, систематизацию, углубление и закрепление знаний:

- назначение и виды информационных технологий, технологии сбора, накопления, обработки, передачи и распространения информации;
- состав, структуру, принципы реализации и функционирования информационных технологий;
- базовые и прикладные информационные технологии;
- инструментальные средства информационных технологий.

Описание каждой практической работы содержит номер, название и цель работы, формируемые в процессе выполнения работы умения и элементы компетенций, изложение необходимого теоретического материала (при необходимости примеры выполнения заданий), варианты заданий, описание алгоритма выполнения работы и контрольные вопросы (с целью выявить и устранить недочеты в освоении материала).

Для получения дополнительной, более подробной информации по основным вопросам учебной дисциплины в конце методических рекомендаций приведен перечень информационных источников.

Отчеты студентов по практическим работам должны содержать номер, название и цель работы, выполненные задания и их результаты, ответы на контрольные вопросы и выводы по проделанной работе.

Титульный лист должен быть оформлен в соответствии с приложением 1.

Отчет должен быть оформлен в соответствии с приложением 2.

ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ РАБОТ

| № | Наименование работы | Количество часов |
|-----|--|------------------|
| 1. | Работа с консольным интерфейсом ОС WINDOWS | 2 |
| 2. | Изучение графического интерфейса и настроек ОС LINUX | 2 |
| 3. | Работа с файлами в операционной системе LINUX | 2 |
| 4. | Редактирование и форматирование документов в среде текстового процессора | 2 |
| 5. | Создание документов, содержащих таблицы и формулы | 2 |
| 6. | Создание и редактирование таблиц в среде текстового процессора | 2 |
| 7. | Создание электронных документов в MSWord | 2 |
| 8. | Создание вычисляемых таблиц в MS Word | 2 |
| 9. | Оформление научной документации средствами MS Word | 2 |
| 10. | Слияние документов в MS Word. Создание структурированного документа | 2 |
| 11. | Технология OLE: внедрение и связывание объектов. Создание составного документа | 2 |
| 12. | Создание и редактирование графических объектов в MS Word | 2 |
| 13. | Обработка текста различной сложности в MS Word. Допечатная подготовка | 2 |
| 14. | Создание и обработка таблиц в MS Excel | 2 |
| 15. | Адресация в Excel. Работа с листами | 2 |
| 16. | Обработка списков | 2 |
| 17. | Представление числовой информации в графическом виде в MS Excel | 2 |
| 18. | Работа со встроенными функциями | 2 |
| 19. | Работа с шаблонами в MS Excel | 2 |
| 20. | MS Excel. Сортировка и фильтрация данных | 2 |
| 21. | Анализ данных: диспетчер сценариев, составление сводных таблиц | 2 |
| 22. | Выполнение консолидации данных и промежуточных итогов | 2 |

| | | |
|--------------------|--|-----------|
| 23. | Технология динамического обмена данными (DDE) между Excel и другими приложениями Windows | 2 |
| 24. | Создание компьютерной презентации в MS PowerPoint | 2 |
| 25. | Управление презентацией | 2 |
| 26. | Монтаж видеоклипов | 2 |
| 27. | Монтаж видеоклипов | 2 |
| 28. | Подготовка видеоролика с использованием программы захвата видео и видеоредактора | 2 |
| 29. | Освоение интерфейса и возможностей HTML-редактора | 2 |
| 30. | Создание web-сайта средствами HTML-редактора | 2 |
| 31. | Создание таблиц, списков средствами HTML-редактора | 2 |
| 32. | Создание ссылок. Работа с объектами web-сайта | 2 |
| 33. | Создание тематического сайта | 2 |
| 34. | Использование бесплатных online-сервисов для создания мультимедийных объектов | 2 |
| 35. | Создание компьютерной презентации в MS PowerPoint | 2 |
| Всего часов | | 70 |

Практическая работа №1

Название практической работы: Работа с консольным интерфейсом ОС WINDOWS

Цель работы: сформировать умения систематизировать и упорядочивать документы на ПК, организовывать их размещение, хранение, обработку, поиск и передачу файлов

знания (актуализация):

- виды и формы представления информации;
- методы и средства определения количества информации.

умения:

- применять закон аддитивности информации.

элементы компетенций:

ОК 1. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 2. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 9. Использовать информационные технологии в профессиональной деятельности.

ПК 3.1. Устанавливать, настраивать, эксплуатировать и обслуживать технические и программно-аппаратные средства компьютерных сетей.

Теоретический материал:

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, используемый оператором в качестве средства общения с компьютером.

Ход работы:

1. Запуск командной строки.

Для запуска командной строки можно воспользоваться одним из следующих способов:

Пуск - Выполнить (или клавиши Win+R) введите **cmd** и нажмите клавишу Enter;

Пуск - Все программы - Стандартные - Командная строка»;

Пуск - Поиск - Командная строка. Также, для запуска командной строки можно использовать заранее подготовленный ярлык, ссылающийся на исполняемый файл **%SystemRoot%\system32\cmd.exe** (обычно – C:\Windows\system32\cmd.exe). Выполнение некоторых команд требует повышенных привилегий, поэтому, приложение командной строки должно быть запущено с использованием пункта контекстного меню «Запуск от имени администратора», вызываемого правой кнопкой мышки.

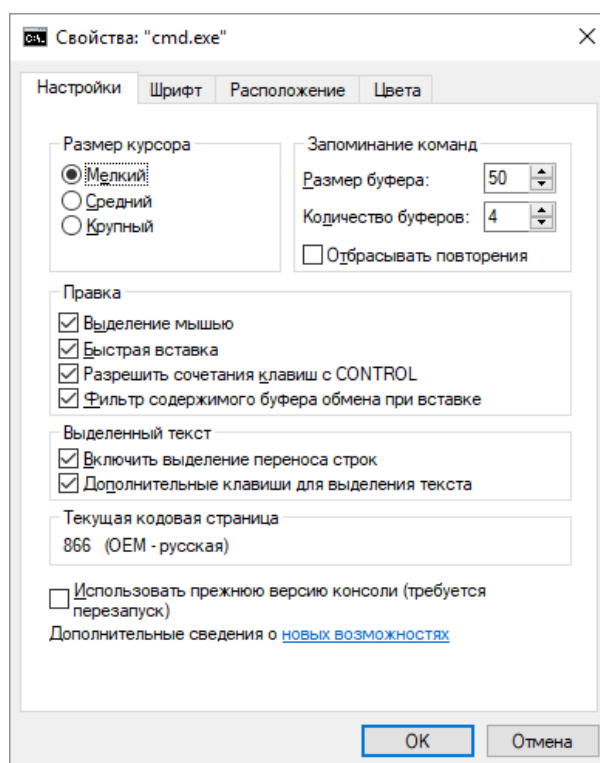
2. Настройка командной строки.

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. Настройки можно выполнить изменяя свойства ярлыка, с помощью которого выполняется запуск приложения командной строки, либо правкой параметров реестра, относящихся к командному процессору.

Работа в командной строке предполагает ввод данных с клавиатуры, причем, иногда довольно большого объема. Этот объем можно значительно сократить, используя некоторые приемы:

Использование буфера обмена.

Текст, выделенный в окне приложения, можно скопировать в буфер обмена, а также вставить его из буфера обмена в поле ввода. При стандартных настройках командной строки для выделения текста используется пункт «Пометить» контекстного меню, вызываемого правой кнопкой мышки. При желании, в свойствах ярлыка, с помощью которого выполняется запуск можно включить режим выделения мышью:



В режиме выделения мышью, контекстное меню не используется, а текст выделяется с использованием левой кнопки мышки. Копирование и вставка выполняется при нажатии правой кнопки мышки. При включенном режиме «Разрешить сочетания клавиш с CONTROL», можно использовать стандартные сочетания клавиш:

- CTRL+C (CTRL+Insert) – скопировать выделенный текст.
- CTRL+V (Shift+Insert) – вставить выделенный текст.

Использование истории команд и горячих клавиш.

Для вызова ранее введенных команд используются клавиши со стрелками Стрелка Вверх - на одну команду назад и Стрелка Вниз - на одну команду вперед. Кроме того, можно использовать функциональные клавиши:

F1 - посимвольный вызов последней введенной команды. Каждое нажатие F1 приводит к последовательной подстановке в поле ввода одного символа из предыдущей команды.

F2 - копирование до символа. После нажатия на F2 на экране отображается запрос символа, до которого будет скопирована предыдущая команда. Например, если предыдущая команда была **ping yandex.ru**, и в

качестве символа задана точка, то в строку ввода команды будет занесено **ping yandex**, если задан пробел, то - **ping**.

F3 - вызов предыдущей команды.

F4 - удалить до символа. Выполняется удаление текста от текущей позиции курсора до заданного символа.

F5 и **F8** - вызов буфера ранее введенных команд. Вывод по нажатию клавиши **F5** прекращается когда будет отображена первая введенная команда текущей сессии.

F7 - отображение ранее введенных команд в отдельном окне в виде списка. Для выполнения нужной команды нужно выбрать ее, используя клавиши со стрелками, и нажать **ENTER**

F9 - выполнить команду, номер которой запрашивается. Номер команды в списке истории можно получить при использовании **F7**.

В Windows 10 / Windows Server 2016 появились возможности, отсутствующие в предыдущих реализациях командной строки:

- Изменение прозрачности окна консоли при нажатии комбинаций **CTRL+Shift+-** или **CTRL+Shift++**.

- Включение / выключение полноэкранного режима при нажатии **CTRL+Enter**.

- Расширение возможностей выделения текста и редактирования:

Shift+Home – выделить текст от текущего положения курсора до начала строки.

Shift+End – выделить текст от текущего положения курсора до конца строки.

- Быстрое перемещение по буферу экрана и поиск по контексту:

CTRL+Home – перейти в начало буфера экрана

CTRL+End – перейти в конец буфера экрана.

CTRL+Стрелка Вверх - перемещение на 1 строку вверх.

CTRL+Стрелка Вниз - перемещение на 1 строку вниз.

CTRL+F - открыть диалог поиска текста в буфере экрана.

Окно командной строки в Windows 10 можно закрыть с помощью стандартной комбинации клавиш - **ALT+F4**.

Для отключения новых возможностей CMD, необходимо в свойствах приложения установить галочку «Использовать прежнюю версию консоли (требуется перезапуск)».

3. Автодополнение путей файлов

В командной строке Windows можно выполнять подстановку имен файлов и каталогов при нажатии клавиши **Tab**. Например, для перехода в каталог с длинным именем **Program Files** наберите начальную часть имени каталога

CD Pro и нажмите **Tab**. В строке ввода должна появиться команда :

CD "Program Files"

Как видим, выполняется подстановка недостающей части имени каталога, а в тех случаях, когда в нем имеется символ пробела, добавляются еще и двойные кавычки.

Если имеется несколько совпадающих начальных частей имен файлов или каталогов, при каждом нажатии клавиши **Tab** будет подставляться следующее по алфавитному порядку имя. Для возврата на предыдущее - используется комбинация **Shift+Tab**

4. Перенаправление данных стандартного ввода /вывода консоли.

В качестве стандартного устройства ввода командной строки используется клавиатура, а в качестве устройства вывода – монитор. Однако, существует возможность **перенаправления** ввода-вывода на другие устройства с использованием операторов перенаправления:

а) **>** - перенаправление вывода. Выходные данные записываются в файл или передаются на указанное устройство.

ping -n 5 localhost > nul - выполнить пинг петлевого интерфейса 5 раз с перенаправлением вывода в фиктивное устройство **nul**. Вывод результатов выполнения команды подавляется. Подобный прием используется для организации задержки в командных файлах, поскольку пинг петлевого

интерфейса выполняется почти мгновенно, а интервал между пингами равен одной секунде, время выполнения данной команды определяется значением параметра **-n**

ping -n 100 yandex.ru > C:\ping-ya.txt - выполнить 100 раз пинг узла yandex.ru с записью результатов выполнения команды в файл C:\ping-ya.txt. Если файл не существует, он будет создан, а если существует, его содержимое будет затерто.

б) **>>** - то же, что и в предыдущем случае, но данные записываются в конец файла.

ping -n 100 yandex.ru >> C:\ping-ya.txt - то же, что и в предыдущем примере, но, если файл не существует, то он будет создан, а если существует, то запись результатов будет выполняться в конец файла.

в) **<** - перенаправление ввода. Данные считываются не с клавиатуры, а из файла или другого устройства.

cmd < 1.txt - запустить командный процессор CMD и выполнить ввод данных из файла 1.txt. Если в файле поместить строку **ping -n 100 yandex.ru**, то выполнится команда, рассмотренная выше.

г) вывод первой команды перенаправить на вход следующей за ней.

Нередко, вывод одной команды нужно передать в качестве вводимых данных для другой, т.е. объединить команды в последовательную цепочку:

ping -n 100 microsoft.com | find "Превышен интервал" - результат выполнения команды **ping -n 100 microsoft.com** передается в виде входных данных для команды поиска строк (**find**), содержащих текст "Превышен интервал".

ping -n 100 microsoft.com | find "Превышен интервал" > C:\ping-ya.txt - то же, что и в предыдущем примере, но с перенаправлением выводимых результатов выполнения команды в текстовый файл.

д) Использование дескрипторов ввода-вывода консоли.

Каждому открытому файлу или устройству соответствует свой дескриптор (**handle**) который представляет собой неотрицательное число, значение которого используется породившим поток ввода-вывода процессом. По

умолчанию, для всех процессов, в том числе и для командного интерпретатора **cmd.exe** :

0 (STDIN) – дескриптор стандартного ввода (ввод с клавиатуры).

1 (STDOUT) – дескриптор стандартного вывода (вывод на экран).

2 (STDERR) – дескриптор вывода диагностических сообщений (сообщений об ошибках на экран).

Дескрипторы можно использовать в тех случаях, когда требуется перенаправить (изменить) источники и приемники данных в стандартных потоках ввода-вывода. Например:

ping.exe -n 100 yandex.ru 2> C:\pinglog.txt - стандартный поток сообщений программы **ping.exe** будет выводиться на экран, а ошибки (стандартный вывод с дескриптором = 2) будут записаны в файл **C:\pinglog.txt** . В реальном случае для программы **ping.exe** приведенная конструкция значения не имеет, поскольку она выводит и диагностику, и результаты на экран.

Для задания перенаправления в существующие дескрипторы используется амперсанд (&), затем номер требуемого дескриптора (например, &1):

ping -n 100 yandex.ru >log.txt 2>&1 - стандартный поток сообщений об ошибках (дескриптор=2) перенаправляется в стандартный поток вывода (дескриптор = 1) и все это перенаправляется в файл **log.txt** текущего каталога.

ping -n 100 yandex.ru >log.txt 1>&2 - стандартный вывод (дескриптор = 1) перенаправляется в вывод сообщений об ошибках (дескриптор=2) и все это записывается в текстовый файл.

Если дескриптор не определен, то по умолчанию оператором перенаправления ввода < будет ноль (0), а оператором перенаправления вывода > будет единица.

5. Объединение нескольких команд в цепочку

В командной строке Windows существует возможность последовательного выполнения нескольких команд в зависимости от результатов их выполнения. Для чего используются символы объединения команд - **&** (амперсанд) и **|** (вертикальная черта)

& - одиночный амперсанд используется для разделения нескольких команд в одной командной строке. Например:

echo ping ya.ru - вывод на экран строки **ping ya.ru**

echo &ping ya.ru - последовательное выполнение команды **echo** без параметров и команды **ping ya.ru**

&& - условное выполнение второй команды. Она будет выполнена, если код завершения (значение которого передается в стандартную переменную **ERRORLEVEL**) первой команды равен нулю, т.е. команда выполнена успешно.

команда1 && команда2 - выполняется **команда1**, а **команда2** выполняется, только если первая была выполнена успешно. Например,:

ping ya.ru -err & ping -n 2 ya.ru - при использовании единичного амперсанда, первая команда **ping ya.ru -err** выдаст сообщение о неверном параметре **-err**, а вторая, **ping -n 2 ya.ru** выполнит двукратный пинг узла **ya.ru**

ping ya.ru -err && ping -n 2 ya.ru - при использовании двойного амперсанда, первая команда **ping ya.ru -err** выдаст сообщение о неверном параметре **-err** и соответственно, сформирует код завершения не равный нулю (ошибка), что приведет к тому, что вторая команда (**ping ya.ru -n 2**) выполнена не будет.

Двойная вертикальная черта **||** - условное выполнение второй команды. Если первая команда завершилась с кодом возврата не равным нулю (неуспешно), то выполняется команда, следующая за двойной вертикальной чертой.

команда1 || команда2 - если **команда1** выполнена неуспешно, то запускается на выполнение **команда2**

ping -n 1 ya.ru && ping -n2 ya.ru || ping -n 3 ya.ru - сначала выполнится 1-я команда **ping ya.ru -n 1** - однократный пинг узла **ya.ru**, затем выполнится команда **ping ya.ru -n 2** - двукратный пинг узла **ya.ru**. Третья команда **ping ya.ru -n 3**, трехкратный пинг узла **ya.ru** выполнена не будет.

ping -n 1 -err ya.ru && ping ya.ru -n 2 || ping ya.ru -n 3 - первая команда **ping ya.ru -n 1 -err** выполнится с ошибкой из-за неверного параметра **-err**, вторая

команда **ping -n 2 ya.ru** - двукратный пинг узла **ya.ru** не будет выполнена из-за условия двойного амперсанда, и в результате выполнится третья команда **ping -n 3 ya.ru**, трехкратный пинг узла **ya.ru**.

В некоторых случаях может возникнуть необходимость запрета обработки служебных символов и трактовки их в качестве простого текста. Например, если ввести в командной строке:

ECHO ping -n 1 ya.ru & ping -n 2 ya.ru, то вместо текста ping -n 1 ya.ru & ping -n 2 ya.ru будет выведена его часть до символа амперсанда ping -n 1 ya.ru и затем выполнена команда после амперсанда - ping -n 2 ya.ru . Проблему решает использование специального экранирующего символа ^ , который позволяет использовать служебные символы как текст:

ECHO ping -n 1 ya.ru ^& ping -n 2 ya.ru - вывод на экран текста ping -n 1 ya.ru & ping -n 2 ya.ru

Логика условной обработки команд, реализуемая с помощью конструкций && и || действует только на ближайшую команду, то есть, при вводе команды

```
TYPE C:\plan.txt && DIR & COPY /?
```

команда COPY /? запустится в любом случае, независимо от результата выполнения команды TYPE C:\plan.txt. Но, несколько команд можно сгруппировать с помощью скобок. Например, есть 2 командные строки:

```
TYPE C:\plan.txt && DIR & COPY /?
```

```
TYPE C:\plan.txt && (DIR & COPY /?)
```

В первой из них символ условной обработки && действует только на команду DIR, во второй — одновременно на две команды: DIR и COPY. В качестве наглядного эксперимента, попробуйте выполнить вторую команду при условиях наличия и отсутствия файла C:\plan.txt. Для создания пустого файла можно воспользоваться копированием из фиктивного устройства nul:

```
copy nul C:\plan.txt
```

Для удаления файла используется команда erase c:\plan.txt или del C:\plan.txt

6. Командные файлы

Командные файлы (сценарии, скрипты) – это обычные текстовые файлы с заранее подготовленным набором команд для их выполнения командным процессором cmd.exe. Стандартно, такие файлы имеют расширение .bat или .cmd. Строки командных файлов могут содержать специфические команды самого процессора команд, например - FOR, ECHO, REM и т.п. или имена исполняемых модулей – reg.exe, sc.exe, auditpol.exe., которые можно использовать без расширения – reg, sc, auditpol. Пример простого командного файла:

REM Создается текстовый файл со списком каталога Windows

dir C:\Windows > %TEMP%\winlist.txt

REM выполняется задержка на 5 секунд

ping -n 5 localhost > nul

REM Файл открывается в редакторе WordPad

write %TEMP%\winlist.txt

REM После завершения работы Wordpad, текстовый файл удаляется.

erase C:\winlist.txt

Строки, начинающиеся с REM являются комментариями. В качестве примера, используются команды для работы с файловой системой и выполняется запуск приложения графической среды – текстового редактора Wordpad (write.exe) с передачей ему параметра командной строки (имя файла). Язык командных файлов довольно примитивен и не в полной мере соответствует требованиям сегодняшнего дня, однако, он является самым простым средством автоматизации рутинных действий и используется большинством системных администраторов и грамотных пользователей.

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

Задачи лабораторной работы Задачами лабораторной работы являются освоение следующих навыков: □ работа с файловой системой в командной строке; □ создание архивных и пакетных файлов; □ настройка переменных среды окружения.

Командная строка Интерфейс операционной системы (ОС) – это средство общения пользователя с ОС. С помощью интерфейса пользователь обращается к различным функциям ОС, таким как запуск программ или копирование файлов, и получает сообщения ОС. Командная строка – это интерфейс ОС, работающий в отдельном текстовом режиме следующим образом: □ ОС выводит на экран приглашение; □ пользователь вводит команду и нажимает [Enter]; □ ОС выводит результаты на экран. Приглашение к вводу команд содержит имя диска и текущий каталог, например: C:\User> Пользователь вводит с клавиатуры команду и ее параметры, разделенные пробелами, например: dir /w /o:n c:\ Параметры командной строки – это части строки, разделенные пробелами. Их нумеруют, начиная с нуля. Нулевой параметр – это имя команды или исполняемого файла. Следующие за ним параметры указывают режим выполнения команды и сообщают имя объекта, с которым будет выполняться операция.

6

Команды ОС, например dir, выполняет сама ОС, поэтому на диске нет отдельных файлов для каждой команды, например dir.exe. К исполняемым файлам относятся прикладные программы (приложения) *.com и *.exe, а также пакетные файлы *.bat и *.cmd. Пакетные файлы – это текстовые файлы, содержащие последовательность команд ОС.

Запуск командной строки Командную строку можно запустить следующими способами: 1) через главное меню Пуск: Пуск ► Программы ► Стандартные ► Командная строка; 2) через главное меню Пуск: Пуск ► Выполнить введите имя программы cmd. 3) через главное меню Пуск: нажмите кнопку [Win] выберите пункт Выполнить введите имя программы cmd. 4) с помощью «горячих клавиш»: □ нажмите комбинацию клавиш [Win + R]; □ откроется окно Запуск программы; □ введите cmd и нажмите [OK] или [Enter] (рис. 1.1).

Рис. 1.1. Запуск командной строки

Окно командной строки содержит приглашение, содержащее полный путь к текущему каталогу. Мигающий курсор указывает место для ввода следующей команды (рис. 1.2). Чтобы просмотреть предысторию, можно использовать полосу прокрутки.

7

Нажав кнопку в левом верхнем углу окна, можно изменить настройки шрифта, а также копировать и вставлять текст через буфер обмена. Чтобы закрыть командное окно, нажмите кнопку [x] в правом верхнем углу окна либо введите команду exit с клавиатуры.

Рис. 2.1. Окно командной строки

Задание □ Запустите командную строку описанными способами. □ Настройте размер шрифта и цвет текста и фона. □ Введите команду ver и нажмите [Enter]. □ Запишите версию ОС. □ Найдите в командном окне все элементы интерфейса: □ Приглашение □ Мигающий курсор □ Команду □ Ответ ОС □ Полосу прокрутки □ Закройте командное окно.

Справочная система Команда help выводит на экран описание указанной команды. Например, чтобы изучить команду dir, нужно ввести команду: help dir. Длинный текст справки выводится на экран постранично. Некоторые команды и программы сами могут выводить справку при указании вопросительного знака в качестве параметра:

8

dir /?. Справку можно также получить через меню Пуск ► Справка и поддержка. В строке поиска справочной системы введите имя команды, например, dir. Квадратные скобки в тексте справки отмечают необязательные аргументы. Например, описание dir [диск:] означает, что допускаются следующие варианты использования команды dir: dir dir c: dir d: Задание □ Вызовите справку для команды dir описанными способами.

История ввода команд Все команды, введенные течение текущего сеанса работы командного окна, сохраняются в памяти. Их можно вернуть стрелками [↑] и [↓] и снова выполнить. Содержимое командной строки можно редактировать клавишами [←], [→], [Home], [End], [Del], [BackSpace]. Клавиша [Backspace] (англ. Back Space – «пробел назад») находится на клавиатуре над клавишей [Enter]. Эта клавиша удаляет символ слева от курсора. Клавиша [Del] удаляет символ справа от курсора. Для очистки текущей команды нажмите

[Esc]. Задание □ Используя клавиши [□] и [□], просмотрите историю ввода. □ Верните команду `help dir` □ Отредактируйте команду, чтобы получить `help help` □ Нажмите [Enter] □ Верните предыдущую команду □ Нажмите [Esc] □ Прочитайте описание [Backspace] на ru.wikipedia.org

Команда DIR Для просмотра содержимого каталога используется команда `dir` (англ. Directory – каталог, папка, директория). На экран выводится

9

список всех каталогов и файлов, находящихся в текущем каталоге диска с указанием их расширения и размеров, даты и времени создания (рис. 1.3). Каталоги помечаются символами <DIR>. Примеры использования команды: `dir /b` – вывод только названий файлов; `dir /d` – вывод списка в несколько колонок; `dir /oe-s` – отсортированный по расширениям список имен каталогов и файлов, а среди них – сортировка от больших файлов к меньшим.

Рис. 1.3. Содержимое каталога

Задание □ Введите следующие команды: □ `dir` □ `dir c:\` □ `dir /b c:\` □ Обратите внимание на результаты выполнения команд □ Вызовите справку для команды `dir` □ Выведите на экран список файлов и каталогов диска D: , отсортированных по расширению в обратном алфавитном порядке;

10

□ Выведите на экран список файлов и каталогов диска D: , отсортированных по дате – от ранних к поздним □ Выведите на экран список файлов и каталогов диска D: , сгруппировав каталоги после файлов □ Выведите на экран в алфавитном порядке имена скрытых файлов □ Выведите на экран в алфавитном порядке имена системных файлов

Команда CD Для перехода из каталога в каталог используйте команду `cd` (англ. Change Directory – сменить каталог). Команда `cd` без параметров выводит имена текущего диска и папки. Примеры использования команды: □ `cd c:\tmp` – переход в каталог `c:\tmp` из любого каталога диска `c:`; □ `cd /d c:\tmp` – переход в каталог `c:\tmp` из любого каталога любого диска; □ `cd tmp` – переход в подкаталог `tmp` в текущем каталоге; □ `cd ..` – переход на один уровень выше по дереву файлов; □ `cd \` – переход в корневой каталог текущего диска; □ `d:` – переход на диск D:. В приглашении ОС каждый раз после выполнения команды выводится имя текущего диска и каталога. Задание □ Вызовите справку для команды `cd` □ Обратите внимание на использование кавычек в команде `cd` □ Используя команды `cd` и `dir`, переходите вверх и вниз

по файловому дереву диска c: □ Сделайте зарисовку фрагмента файлового дерева □ Установите текущий диск d: □ Переходите вверх и вниз по файловому дереву диска d: □ Перейдите в каталог C:\Program Files с одновременной сменой диска и каталога □ Перейдите в корневой каталог

11

Команда TREE Команда `tree` (англ. Tree – Дерево) выводит на экран схематичное изображение дерева каталогов. Если не указаны диск или путь, на экран выводится часть дерева, начиная с текущего каталога. Чтобы прервать вывод на экран, нажмите [Ctrl + C]. Примеры: □ `tree \` – вывод дерева каталогов всего диска C:; □ `tree c:\ /f | more` – постраничный вывод дерева диска C:. Задание □ Вызовите справку для команды `tree` □ Введите команду `tree` □ Введите команду `tree` и нажмите [Ctrl + C] □ Введите команду `tree` с постраничным выводом □ Сделайте зарисовку части дерева диска C:, начиная с корневого каталога

Команда SET Команда `set` (англ. Set – установить) позволяет просматривать и изменять переменные среды окружения (англ. Environment variables – параметры окружения). Это настройки ОС, доступные для прикладных программ в виде текстовых строк в оперативной памяти. К переменным окружения обращаются по имени. Команда `set` без параметров выводит текущие значения переменных среды окружения (см. рис. 1.4.), в том числе: □ `HOMEDRIVE` – диск, на котором находится домашний каталог; □ `HOMEPATH` – путь к домашнему каталогу; □ `NUMBER_OF_PROCESSORS` – число процессоров; □ `OS` – операционная система; □ `PATH` – путь поиска исполняемых файлов; □ `PROMPT` – формат приглашения командной строки; □ `USERNAME` – имя пользователя.

12

Рис. 1.4. Результаты вызова команды `set`

Для обращения к значению переменной окружения используют символ процента. Например, `%os%` – значение переменной `os`. Примеры: □ `set path=C:\Temp;%path%` – добавление каталога C:\Temp в путь поиска □ `echo %os%` – вывод на экран значения переменной `OS` Задание □ Выведите полный список переменных окружения □ Определите число процессоров □ Определите тип операционной системы □ Выясните в Википедии, что такое Windows NT □ Определите имя пользователя □ Выведите путь для поиска □ Добавьте каталог

d:\ в путь для поиска □ Выведите путь для поиска □ Выведите полный список переменных окружения □ Убедитесь, что значение переменной path изменилось

Команда PATH Команда path (англ. Path – Путь) устанавливает пути поиска исполняемых файлов. Если при запуске приложения полный путь к

13

файлу не указан, ОС пытается найти исполняемый файл в текущем каталоге, а затем просматривает пути к каталогам из заранее заданного списка path. Список путей для поиска хранится в переменной среды окружения PATH. Для просмотра списка путей используется команда path без параметров. Примеры: □ path □ path c:\usr; d:\bin – задание двух путей для поиска; □ path %path%; d:\s – добавляет каталог d:\s к пути поиска. Изменение set-параметров из командной строки действует только для текущего командного окна. После изменения путей для поиска ОС не сможет выполнять некоторые команды. Для восстановления работоспособности закройте текущее командное окно и откройте новое. Задание □ Выведите список путей поиска □ Добавьте каталог c:\1 в список путей □ Выведите список путей поиска □ Убедитесь, что каталог добавлен в список

Создание файлов Для обеспечения безопасности текущие настройки ОС в дисплейном классе не позволяют пользователям создавать файлы в корневом каталоге диска c:\. Рекомендуется проводить опыты с файлами в каталоге C:\Student. Для создания текстовых файлов в командной строке можно использовать два способа. Способ 1. Копирование ввода с клавиатуры в файл: copy con file.txt При выполнении этой команды данные с клавиатуры записываются в указанный файл. Для завершения ввода текста нажмите комбинацию [Ctrl + Z] , а затем [Enter]. con (англ. Console – Консоль) – устройство стандартного вводавывода информации. «Стандартный ввод» выполняется с клавиатуры, а «стандартный вывод» – на экран.

14

Способ 2. Перенаправление стандартного вывода: echo текст > file.txt Команда echo (англ. Echo – эхо) выводит текст, напечатанный после команды, на экран. Знак > направляет вывод вместо экрана в файл. В результате, при выполнении этой команды будет создан файл file.txt, который будет содержать в себе текст, напечатанный после команды echo. Задание □ Создайте текстовый файл a.txt с помощью копирования с консоли в файл. □ Создайте текстовый файл b.txt с помощью перенаправления стандартного вывода.

Вывод на экран Команда type выводит на экран содержимое текстового файла (например, *.txt или *.bat), например: type f.txt Можно также выполнить копирование файла на экран (устройство стандартного вывода, консоль), например copy f.txt con Задание □ Выведите на экран файл a.txt командой type □ Выведите на экран файл b.txt командой copy

Команда COPY Команда copy (англ. Copy – Копировать) используется, чтобы скопировать один или несколько файлов. Первый параметр команды – исходный файл (источник). Вторым параметром – результат копирования (каталог или новый файл). Например, команда copy f1.txt d:\t копирует файл f1.txt в каталог d:\t Для объединения нескольких файлов в один файл между именами файлов ставится знак +. Например команда copy f1+f2+f3 f4 объединяет файлы f1, f2 и f3 в файл f4 Вторым параметром команды может содержать имя файла или путь к файлу. Примеры:

15

□ 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:\t\2.txt – копирование файла 1.txt в файл 2.txt в каталоге d:\t Звездочка * заменяет любой символ или несколько символов в названии файла. Например, команда copy *.txt d:\ копирует все файлы с расширением txt в каталог d:\ Задания □ Скопируйте файл a.txt в каталог d:\ □ Выведите на экран файл d:\a.txt □ Скопируйте файл a.txt в d:\z.txt □ Выведите на экран файл d:\z.txt □ Скопируйте файл b.txt в файл c.txt □ Выведите на экран файл c.txt □ Скопируйте файл a.txt в файл c.txt □ Выведите на экран файл c.txt □ Скопируйте файлы a.txt и b.txt в файл c.txt □ Выведите на экран файл c.txt □ Скопируйте все файлы типа txt в файл d.txt □ Выведите на экран файл d.txt

Работа с архивами Архив – это файл, внутри которого упакованы каталоги и файлы. Архиватор – это программа, которая выполняет упаковку и распаковку архивов. Многотомный архив – это большой архив, разбитый на несколько частей. Такие архивы можно записать на несколько дисков или отправить частями по сети. Самораспаковывающийся архив (SFX, Self-Extracting archive) – это исполняемый файл *.exe, содержащий архив и программу распаковки. При запуске такого файла производится распаковка архива без использования архиватора.

16

В лабораторной работе используется программа 7-zip – свободно распространяемый архиватор. Сайт программы: 7-zip.org. Чтобы запускать архиватор из командной строки, необходимо добавить в путь поиска каталог C:\Program Files\7-Zip. Примеры использования архиватора: `7z a x1 b.txt` – упаковка файла b.txt в архив x1.7z `7z l 1.7z` – вывод списка файлов из архива 1.7z; `7z e 1.7z` – извлечение файлов из архива 1.7z `7z e 1.7z -od:\` – извлечение файлов в каталог d:\ `7z a -sfx rr *.txt` – сжатие всех текстовых файлов в самораспаковывающийся архив rr.exe В описании архиватора используются следующие обозначения. В квадратных скобках даются необязательные параметры. Угловые скобки служат для обозначения параметров. При вводе команды квадратные и угловые скобки не используют. Например, следующее описание `7z <command> [<switches>...] <archive_name> [<file_names>...]` означает, что пользователь может вводить такие команды: `7z a ghjk 7z l ghjk 7z a ghjk a.txt 7z a -sfx ghjk b.doc b.txt`

Задания ☐ Выведите на экран путь поиска ☐ Добавьте путь для поиска архиватора `set path=%path%;C:\Program Files\7-Zip` ☐ Выведите на экран путь поиска ☐ Убедитесь, что путь добавлен ☐ Введите команду 7z и прочитайте справку по использованию архиватора ☐ Выясните назначение и перевод команд `a`, `e`, `l` ☐ Упакуйте файл c.txt в архив ss.7z ☐ Сравните размеры исходного и сжатого файлов ☐ Создайте текстовый файл tt.doc в редакторе MS Word

17

☐ Упакуйте файл tt.doc в архив t1.7z ☐ Сравните размеры исходного и сжатого файлов ☐ Упакуйте все файлы с расширением *.txt в архив ttt.7z ☐ Выведите список файлов из архива ttt.7z; ☐ Распакуйте архив ttt.7z в каталог d:\ ☐ Убедитесь, что операция выполнена успешно ☐ Упакуйте файл tt.doc в самораспаковывающийся архив в каталоге d:\ ☐ Сравните размеры исходного и сжатого файлов ☐ Перейдите в каталог d:\ ☐ Запустите созданный архивный файл на выполнение ☐ Убедитесь, что операция выполнена успешно

Перенаправление ввода-вывода При выполнении программ в командной строке используется так называемый стандартный ввод-вывод. Это стандартные встроенные функции ОС для ввода текста с клавиатуры и вывода текста на экран. Такие действия выполняются, например, при использовании команд ReadLn и WriteLn в программе на языке Pascal. Для перенаправления стандартного ввода-вывода используют следующие символы: `>` – записать в файл (вместо вывода на экран); `>>` – добавить к файлу (вместо вывода на экран); `<` – прочитать из файла (вместо ввода с клавиатуры); `|` – объединить две

команды в конвейер (вместо вывода на экран и 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.

18

При организации конвейера стандартный вывод первой команды (вместо вывода на экран) направляется на стандартный ввод второй команды (вместо чтения с клавиатуры). Например, конвейер `dir | sort` создает список файлов, сортирует его и выводит на экран. Команда `more` организует постраничный вывод на экран. При организации конвейера в виде `tree | more`, команда `more` перехватывает вывод на экран результатов работы команды `tree` и выводит длинный текст по страницам. При работе с командой `more` для вывода на экран следующей строки нажимают `[Enter]`, для вывода следующей страницы – пробел `[Space]`, для выхода из страничного просмотра – клавишу `[Q]` (англ. Quit – Выход). Задания □ Вызовите команду `dir` и запишите результаты в файл `d4.txt` □ Выведите файл `d4.txt` на экран □ Вызовите команду `ver` и добавьте результаты к файлу `d4.txt` □ Выведите файл `d4.txt` на экран □ Выведите файловое дерево в постраничном режиме

Пакетные файлы Пакетный файл – это текстовый файл с расширением `*.bat` (англ. Batch – Пакет). В пакетном файле записывают последовательность команд ОС. Имеется также возможность составления несложных программ. 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` Задание □ Выведите справку по команде `date` □ Командой `echo` создайте пакетный файл `da.bat`, в котором содержится команда вывода на экран текущей даты □ Запустите файл `da.bat` на выполнение, указав расширение □ Запустите файл `da.bat` на выполнение, не указав расширение

19

□ Запустите файл `da.bat` на выполнение в отдельном окне □ Перейдите в корневой каталог диска `D:` □ Запустите на исполнение файл `da.bat` без указания

пути □ Ознакомьтесь с сообщением ОС □ Добавьте к пути поиска полный путь к каталогу, где находится файл da.bat □ Убедитесь, что путь поиска изменен □ Запустите на исполнение файл da.bat без указания пути □ Убедитесь, что файл успешно запущен

Работа с файлами и каталогами Для создания каталогов используется команда `mkdir` (англ. Make Directory – Создать каталог). Для удаления каталогов используется команда `rmdir` (англ. Remove Directory – Удалить каталог). Для удаления файлов используется команда `del` (англ. Delete – Удалить). Задания □ Вызовите справку по `mkdir` □ Создайте каталог (в названии используйте номер группы) □ Перейдите в новый каталог □ Создайте каталог (в названии используйте фамилию) □ Перейдите в новый каталог □ Создайте текстовый файл □ Вызовите справку по `del` □ Удалите текстовый файл □ Поднимитесь на один уровень вверх по файловому дереву □ Вызовите справку по `rmdir` □ Удалите каталог □ Создайте каталог □ Поднимитесь на один уровень вверх по файловому дереву □ Удалите каталог

Управление процессами Для вывода списка процессов используется команда `tasklist` (англ. Task List – Список задач).

20

Для остановки выполнения процесса используется команда `taskkill` (англ. Task Kill – «Убить» задачу). Для остановки процесса требуется указать идентификатор процесса PID (англ. Process Identifier – Идентификатор процесса). PID – это «уникальное» целое число, назначенное каждому процессу. У всех процессов эти номера разные. Задание □ Выведите список процессов □ Определите, какой процесс занимает больше всего памяти □ Определите номер процесса `cmd.exe` □ Запустите второе командное окно □ Выведите список процессов □ Определите номер второго процесса `cmd.exe` □ Определите, сколько памяти занимает каждый процесс `cmd.exe` □ Введите в первом командном окне несколько команд □ Выведите список процессов □ Определите, сколько памяти занимает каждый процесс `cmd.exe` □ Остановите процесс `cmd.exe`, который занимает больше памяти □ Выведите список процессов □ Убедитесь, что запущен только один процесс `cmd.exe` □ Удалите все файлы и каталоги, созданные за время выполнения лабораторной работы

1. Ответьте на контрольные вопросы:

- Как происходит сбор и хранение данных?
- Как происходит передача данных?

- Из каких технологических процессов состоит процесс обработки информации?
- Как осуществляется вывод данных?

2. Оформите и сдайте отчет преподавателю (образец отчета представлен в Приложении 2).

Практическая работа №2

Название практической работы: Измерение количества информации

Приложение 1

Министерство образования и науки Челябинской области
Государственное бюджетное профессиональное образовательное учреждение
«Южно-Уральский государственный технический колледж»

ОТЧЕТ

по практическим работам

учебная дисциплина

«ОСНОВЫ ТЕОРИИ ИНФОРМАЦИИ»

специальность 09.02.06

Сетевое и системное администрирование

Квалификация: сетевой и системный администратор

Выполнил: _____

Группа: _____

Проверил: _____

Челябинск, год

Приложение 2

Отчет по практической работе

Практическая работа № 1

Название практической работы: Способы хранения обработки и передачи информации

Цель работы: сформировать умения систематизировать и упорядочивать документы на ПК, организовывать их размещение, хранение, обработку, поиск и передачу файлов

Ход работы:

1. Выполните вставку скриншотов файлов: *Моя профессия, Фамилия, Моя группа.*

2. Опишите алгоритм создания файла *Моя группа*

3. Ответы на контрольные вопросы:

– ...

– ...

4. Вывод по работе: ...

Приложение 3

Отчет по практической работе

Практическая работа № 2

Название практической работы: Измерение количества информации

Цель работы: научиться измерять информацию различными методами, использовать правила перевода информации из одних единиц измерения в другие.

Ход работы:

1. Выполните перевод единиц измерения информации:

Расчеты

2. Используя содержательный подход к измерению информации, решите задачи:

Условия задач и решение

3. Используя алфавитный подход к измерению информации, решите задачи:

Условия задач и решение

4. Решите задачи на измерение информации:

Условия задач и решение

5. Ответы на контрольные вопросы:

– ...

– ...

6. Вывод по работе: ...