# Лабораторная работа № 1

# ОСНОВЫ РАБОТЫ В ОС LINUX

## **1 Цель работы**

Целью работы является изучение архитектуры и принципов функционирования многопользовательской многозадачной операционной системы Linux.

## **2 Теоретическая часть**

**2.1 Выполнение простых команд**

;

Формат команд в ОС LINUX следующий:

**имя команды [аргументы] [параметры] [метасимволы]**

Имя команды может содержать любое допустимое имя файла; аргументы - одна или несколько букв со знаком минус (-); параметры - передаваемые значения для обработки; метасимволы интерпретируются как специальные операции. В квадратных скобках указываются необязательные части команд.

Введите команду **echo**, которая выдает на экран свои параметры:

**echo good morning**

и нажмите клавишу *Enter*. На экране появится приветствие "*good morning*" – параметр команды **echo.** Командный интерпретатор *shell* вызвал команду **echo**, реализованную в виде программы на языке СИ, и передал ей параметры. После этого интерпретатор команд вывел знак-приглашение. Синтаксис команды **echo**:

**echo [-n] [arg1] [arg2] [arg3]...**

Команда помещает в стандартный вывод свои параметры, разделенные пробелами и завершаемые символом перевода строки. При наличии флага -*n* символ перевода строки исключается.

**who**  - получение информации о работающих пользователях.

В квадратных скобках указываются параметры команды, которые можно опустить. Ответ представляется в виде таблицы, которая содержит следующую информацию:

- идентификатор пользователя;

- идентификатор терминала;

- дата подключения;

- время подключения.

**date** - вывод на экран текущей даты и текущего времени.

**cal [[месяц]год]** - календарь; если календарь не помещается на одном экране, то используется команда **cal год | more** и клавишей пробела производится постраничный вывод информации.

**man <название команды>** - вызов электронного справочника об указанной команде. Выход из справочника - нажатие клавиши Q.

Команда **man man** сообщает информацию о том, как пользоваться справочником.

**tty** - сообщение имени специального файла стандартного вывода, соответствующего терминалу пользователя.

**cat <имя файла>** - вывод содержимого файла на экран. Команда **cat > text.1** создает новый файл с именем text.1, который можно заполнить символьными строками, вводя их с клавиатуры. Нажатие клавиши *Enter* создает новую строку. Завершение ввода - нажатие *Ctrl - d*. Команда **cat text.1 > text.2** пересылает содержимое файла text.1 в файл text.2. Слияние файлов осуществляется командой **cat text.1 text.2 > text.3**.

**ls [-alrstu] [имя]** - вывод содержимого каталога на экран. Если параметр не указан, выдается содержимое текущего каталога.

Аргументы команды:

-a - выводит список всех файлов и каталогов, в том числе и скрытых;

-l - выводит список файлов в расширенном формате, показывая тип каждого элемента, полномочия, владельца, размер и дату последней модификации;

- r - выводит список в порядке, обратном заданному;

- s - выводит размеры каждого файла;

- t - перечисляет файлы и каталоги в соответствии с датой их последней модификации;

- u - перечисляет файлы и каталоги в порядке, обратном их последней модификации.

**rm <имя файла>** - удаление файла (файлов). Команда **rm text.1 text.2 text.3** удаляет файлы text.1, text.2, text.3. Другие варианты этой команды - **rm text.[123]** или **rm text.[1-3].**

**wc [имя файла]** - вывод числа строк, слов и символов в файле.

**clear** - очистка экрана.

**2.2 Группирование команд**

Группы команд или сложные команды могут формироваться с помощью специальных символов (метасимволов):

& - процесс выполняется в фоновом режиме, не дожидаясь окончания предыдущих процессов;

? - шаблон, распространяется только на один символ;

\* - шаблон, распространяется на все оставшиеся символы;

| - программный канал - стандартный вывод одного процесса является стандартным вводом другого;

> - переадресация вывода в файл;

< - переадресация ввода из файла;

; - если в списке команд команды отделяются друг от друга точкой с запятой, то они выполняются друг за другом;

&& - эта конструкция между командами означает, что последующая команда выполняется только при нормальном завершении предыдущей команды ( код возврата 0 );

|| - последующая команда выполняется только, если не выполнилась предыдущая команда ( код возврата 1 );

() - группирование команд в скобки;

{ } - группирование команд с объединенным выводом;

[] - указание диапазона или явное перечисление (без запятых);

>> - добавление содержимого файла в конец другого файла.

Примеры.

**who | wc** - подсчет количества работающих пользователей командой **wс** (word count - счет слов);

**cat text.1 > text.2** - содержимое файла text.1 пересылается в файл text.2;

**cat text.1,text.2** - просматриваются файлы text.1 и text.2;

**cat text.1 >> text.2** - добавление файла text.1 в конец файла text.2;

**cc primer.c &** - трансляция Си - программы в фоновом режиме. Имя выполняемой программы по умолчанию a.out.

**gcc -o primer.o primer.c** - трансляция Си-программы с образованием файла выполняемой программы с именем primer.o;

**rm text.\*** - удаление всех файлов с именем text;

**{cat text.1; cat text.2} | lpr** - пpосмотp файлов text.1 и text.2 и вывод их на печать;

**ps [-al] [number]** - команда для вывода информации о процессах:

-a - вывод информации обо всех активных процессах, запущенных с вашего терминала;

-l - полная информация о процессах;

number - номер процесса.

Команда **ps** без параметров выводит информацию только об активных процессах, запущенных с данного терминала, в том числе и фоновых. На экран выводится подробная информация обо всех активных процессах в следующей форме:

F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY TIME CMD

1 S 200 210 7 0 2 20 80 30 703a 03 0:07 cc

1 R 12 419 7 11 5 20 56 20 03 0:12 ps

F - флаг процесса (1 - в оперативной памяти, 2 - системный процесс, 4 - заблокирован в ОЗУ, 20 - находится под управлением другого процесса, 10 - подвергнут свопингу);

S - состояние процесса (O - выполняется процессором , S - задержан, R - готов к выполнению, I - создается);

UID - идентификатор пользователя;

PID - идентификатор процесса;

PPID - номер родительского процесса;

C - степень загруженности процессора;

PRI - приоритет процесса, вычисляется по значению переменной NICE и чем больше число, тем меньше его приоритет;

NI - значение переменной NICE для вычисления динамического приоритета, принимает величины от 0 до 39;

ADDR - адрес процесса в памяти;

SZ - объем ОЗУ, занимаемый процессом;

WCHAN - имя события, до которого процесс задержан, для активного процесса - пробел;

TTY - номер управляющего терминала для процесса;

TIME - время выполнения процесса;

CMD - команда, которая породила процесс.

**nice [-приращение приоритета] команда[аргументы]** - команда изменения приоритета. Каждое запущенное задание (процесс) имеет номер приоритета в диапазоне от 0 до 39, на основе которого ядро вычисляет фактический приоритет, используемый для планирования процесса. Значение 0 представляет наивысший приоритет, а 39 - самый низший. Увеличение номера приоритета приводит к понижению приоритета, присвоенного процессу. Команда **nice -10 ls -l** увеличивает номер приоритета, присвоенный процессу **ls -l** на 10.

**renice 5 1836** - команда устанавливает значение номера приоритета процесса с идентификатором 1836 равным 5. Увеличить приоритет процесса может только администратор системы.

**kill [-sig] <идентификатор процесса>** - прекращение процесса до его программного завершения. Sig - номер сигнала. Sig = -15 означает программное (нормальное) завершение процесса, номер сигнала = -9 - уничтожение процесса. По умолчанию sig= -9. Вывести себя из системы можно командой kill -9 0. Пользователь с низким приоритетом может прервать процессы, связанные только с его терминалом.

**mc** - вызов файлового менеджера (программы - оболочки) *Midnight Commander*, аналогичного *Norton Commander*.

**sort [-dr]** - сортировка входных файлов и вывод результата на экран.

## **2.3 Краткое описание командного интерпретатора Shell**

Интерпретатор команд **Shell** анализирует команды, вводимые с терминала либо из командного файла, и передает их для выполнения в ядро системы. Команды обычно имеют аргументы и параметры, которые обеспечивают модернизацию выполняемых действий. **Shell** является также языком программирования, на котором можно создавать командные файлы (shell-файлы). При входе в ОС пользователь получает копию интерпретатора **Shell** в качестве родительского процесса. Далее, после ввода команды пользователем создается порожденный процесс, называемый процессом-потомком. Т.е. после запуска ОС каждый новый процесс функционирует только как процесс - потомок уже существующего процесса. В ОС Linux имеется возможность динамического порождения и управления процессами.

Обязательным в системе является интерпретатор **Bash**, полностью соответствующий стандарту POSIX. В качестве **Shell** может быть использована оболочка **mc** с интерфейсом, подобным Norton Commander.

## **2.4 Редактор vi**

В составе ОС LINUX обычно поставляются текстовые редакторы: **ed** - интерактивный строковый редактор, **vi** и **ех** - его расширенные версии. Под именем **vi** (**visual interpretator** - визуальный интерпретатор) эта программа работает как экранно-ориентированный редактор, а под именем **ех** - как строчно-ориентированный.

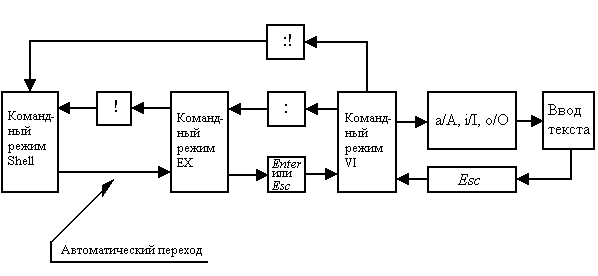
Для вызова редактора **vi** используется команда **vi**:

**vi [+line] [-R] [-х] [-r] [-t] file...**

где +line - номер строки, с которой Вы хотите начать редактирование; R - читать; это означает, что файл можно только просматривать, но не модифицировать; х - расшифровывающее чтение т.е. просмотр файла, зашифрованного командой **сryрt**, или редактирование обычного текста с последующим шифрованием по мере записи на диск; r - восстановление файла после системного или программного крахов; t - вызов для редактирования файла, который содержит названный (в поле filе команды **vi**) тег (tag). Тег - это список символов, с которых начинается раздел в текстовом файле. Теги разных файлов объединяют в один файл - файл тегов с именем tags. Опцией -t обеспечивается вызов файла tags, который содержит названный тег и имя редактируемого файла, в котором тег находится. Команду вызова редактора можно использовать в форме **vi +/word/file** - начало редактирования файла file с первой строки, которая содержит слово word, или в форме **vi +file** - начало редактирования файла с последней строки.

**Структура редактора**

Работая с редактором, пользователь находится или в одном из его командных режимов, или в режиме ввода текста. Ниже приведенная схема иллюстрирует взаимодействие этих режимов и способы перехода редактора между ними.



В простейшем случае для вызова редактора нужно ввести команду **vi** техт и нажать клавишу *Enter*. На экране появится:

$ vi tехt

\_

~

.

.

"text"

Строка начинается знаком ~, знак \_ определяет положение курсора. В данный момент пользователь находится в командном режиме **vi**. Перейти в режим ввода текста можно с помощью команд добавления текста, которые не отображаются на экране после их ввода:

а/А - ввод текста после курсора/после конца строки (аррend - присоединение);

i/I - вставка текста перед курсором/с 1-й позиции данной строки (insert - вставить);

о/О - образовать пустую строку ниже имеющейся / выше имеющейся.

Для выполнения команд (например, записи в файл, перемещения курсора) после введения текста или его части нужно перейти снова в командный режим **vi**, нажав клавишу *Еsс*. После вызова **vi** нажмите клавишу а (ввод текста после курсора), не нажимая после этого клавишу *Enter*, и Вы попадете в режим ввода текста. Вводите текст, нажимая клавишу *Enter* в конце каждой строки (курсор в режиме ввода текста можно перемещать вправо, используя клавишу "*пробел*", и влево, используя клавишу *ВаскSрасе*.

**Переход в командный режим vi**. Для перехода в командный режим vi нужно нажать клавишу *Еsс*. Теперь редактор находится в командном режиме vi. В этом режиме выполняются следующие команды:

. - повторение последней команды;

u - аннулирование действия последней команды;

**Переход в режим ex**. Чтобы перейти к группе команд редактора ех (под именем ех редактор работает как строчно-ориентированный), нужно ввести символ : (двоеточие), команду и нажать <Enter> или Еsс. Команды редактора ех начинаются с символа : и отображаются в нижней части экрана. После нажатия клавиши Еsс или <Enter> происходит возврат (назад) в командный режим. Команды режима ех:

:w - запись текста в файл;

:r - чтение файла;

:е - редактирование нового файла;

:е! - выход без сохранения данного файла и редактирование нового;

:n - авторедактирование;

:wq - запись текста и выход из редактора;

:x - запись текста только при наличии в нем изменений;

:q! - оставить текст в рабочей области и закончить редактирование;

:аb - присвоение сокращений;

:mар - определение ключей;

:set - изменение установочных режимов;

:s - выполнение замещений.

**Переход в Shell.** Редактор позволяет в процессе работы с ним выполнять команды ОС LINUX. Для этого нужно перейти в командный режим Shell с помощью команды !.

Рассмотрим пример. Определите текущее время командой date (вывод и установка даты) :! date. Здесь символ : означает переход в командный режим ех, а символ ! дает доступ к Shell. Для продолжительной работы с командами Shell можно вызвать командой :bash и после окончания работы вернуться в редактор vi, набрав СТRL-D.

Для возврата в командный режим vi нажмите клавишу *Enter*.

**Команды, выполняемые в командном режиме VI**

Изучим группу команд режима vi: перемещения курсора, добавления текста, поиска (частично), изменения и смещения текста, удаления, замены букв. Команды vi не отображаются на экране, кроме команд поиска, начинающихся со знаков / ? перемещение курсора, управление экраном дисплея, добавление текста.

Многие команды редактора выполняются только при определенном положении курсора, и нужно уметь пользоваться клавишами управления курсором (клавиши со стрелками <- , -> и т.д.). Кроме клавиши со стрелками для перемещения курсора можно использовать клавиши: СТRL-Н - влево; СТRL-N - вниз; СТRL-Р - вверх; SРАСЕ - вправо.

**Команды перемещения курсора:**

h - на одну позицию влево;

l - на одну позицию вправо;

j - на одну позицию вниз;

k - на одну позицию вверх;

b - к первому символу предыдущего слова;

В - то же самое, что b, но игнорируются знаки пунктуации;

w - к первому символу следующего слова;

W - то же самое, что w, но игнорируются знаки пунктуации;

е - к последнему символу следующего слова;

Е - то же самое, что е, но игнорируются знаки пунктуации;

( - к началу текущего предложения (предложение считается законченным, если после него есть два пробела или пустая строка);

) - к концу текущего предложения;

{ - к началу текущего раздела (разделителем раздела является пустая строка);

} - к концу текущего раздела;

[ - к началу текущей секции;

] - к концу текущей секции;

^ - к первому отображаемому символу на текущей строке;

О - к началу текущей строки;

$ - к концу текущей строки;

Н - к началу экрана;

М - на середину экрана;

L - к концу экрана;

nG - к строке с номером n (на последнюю строку, если номера n нет); % - к символу парной скобки, если курсор находится под одной из них.

**Команды управления экраном:**

^U - смещение текста на одну строку вверх (СТRL-U);

^D - смещение текста на одну строку вниз (СТRL-D);

^В - смещение текста на один кадр назад (СТRL-В);

^F - смещение текста на один кадр вперед (СТRL-F).

Чтобы переместить текущую строку:

 в верхнюю часть экрана нужно ввести команду z и нажать клавишу *Enter*;

 в середину экрана z;

 в нижнюю часть экрана z- .

Для очистки экрана от сообщений нужно использовать команды СТRL-R и СТRL-L; тексты в рабочей области при этом сохраняются.

**Команды изменения текста:**

сw - изменение слова;

сW - то же самое, что и сw, но игнорируются знаки пунктуации;

сО - от начала текущей строки;

с$ - до конца текущей строки;

сс - изменение всей строки;

с( - от начала текущего предложения;

с) - до конца текущего предложения;

с{ - от начала текущего раздела;

с} - до конца текущего раздела.

Для внесения изменений в текст необходимо: переместить курсор в нужную позицию; ввести команду изменения; без пробела набрать новый текст; нажать клавишу ЕSС.

Во всех командах можно использовать множители n, например для изменения пяти слов используется команда с5w.

Команды поиска начинаются косой чертой / (поиск вперед по тексту) или знаком ? (поиск назад); далее следует номер строки или ключевое слово. Команда заканчивается нажатием клавиши *Enter*.

**Команды смещения текста:**

<(или>( - к началу текущего предложения;

<)или>) - к концу текущего предложения;

<{или>{ - к началу текущего раздела;

<}или>} - к концу текущего раздела.

В командах смещения текста можно использовать множители, например может использоваться команда 2>> (сдвиг вправо). Смещение устанавливается командой: set sw=m. По умолчанию m=8. После того как курсор подведен к требуемой строке, нужно набрать символы << или >>.

**Удаление, замена строчных букв на прописные и наоборот**. Для удаления текста/фрагмента нужно переместить курсор в требуемую позицию и ввести команду удаления.

dw - до конца текущего слова;

dW - то же, что и dw, но игнорируются знаки пунктуации;

d^ - до 1-го видимого символа текущей строки;

dО - удаление начала строки;

d$ - удаление конца строки;

d( - до начала текущего предложения;

d) - до конца текущего предложения;

d{ - до начала текущего раздела;

d} - до конца текущего раздела;

dd - удаление всей строки;

dkw - удаление k слов;

dk)/dk} - удаление k предложений, k разделов;

kdd - удаление k строк.

Для удаления одиночного символа нужно подвести к нему курсор и набрать х (не d), а для удаления нескольких символов подряд набрать команду nх.

Для удаления текста от начала строки до определенного места и от определенного места до конца строки используются команды d^ и d$ соответственно.

Символ ~ используется для замены строчных букв на прописные и наоборот. Замена 1-й буквы в последней строке текста:

 Введите символ ( (к началу текущего предложения).

 Наберите команду .~

 Восстановите текст командой u.

**Определение текущей ра**б**очей позиции в файле**. После ввода пользователем в командном режиме СТRL-G в нижней части экрана появится статусная информация в соответствии с положением курсора в текcте, включающая: имя файла; сведения о проведенной ранее модификации; номер текущей строки; общее число строк; расстояние курсора от начала файла (в процентах).

Для окончания работы с редактором введите в командном режиме :wq (запись текста из рабочей области в файл и окончание редактирования) и нажмите клавишу *Enter.* На экране появится сообщение о том, что Вы вышли из редактора и находитесь в Shell:

:wq <*Enter*>

## **3 Задание для самостоятельной работы**

Порядок выполнения работы:

1. Ознакомиться с теоретической частью к лабораторной работе.

2. Определить день недели, в который Вы родились.

3. Вывести календарь на следующий месяц на экран.

4. Вывести календарь на следующий месяц в файл. Добавить в конец этого файла Ваше имя и фамилию, а также текущую дату.

5. Получить подробную информацию обо всех активных процессах.

6. Используя редактор VI (см. приложение), создать два текстовых файла (с расширением TXT) и командой САТ просмотреть их на экране.

7. Получить информацию о работающих пользователях, подсчитать их количество и запомнить в файле.

8. Объединить текстовые файлы в единый файл и посмотреть его на экране.

9. Создайте текстовый файл с использованием редактора nano.