

Цель работы

Приобретение практических навыков взаимодействия пользователя с системой посредством командной строки.

Задание

1. Определите полное имя вашего домашнего каталога. Далее относительно этого каталога будут выполняться последующие упражнения.
2. Выполните следующие действия:
 - 2.1. Перейдите в каталог /tmp.
 - 2.2. Выведите на экран содержимое каталога /tmp. Для этого используйте команду ls с различными опциями. Поясните разницу в выводимой на экран информации.
 - 2.3. Определите, есть ли в каталоге /var/spool подкаталог с именем cron?
 - 2.4. Перейдите в Ваш домашний каталог и выведите на экран его содержимое. Определите, кто является владельцем файлов и подкаталогов?
3. Выполните следующие действия:
 - 3.1. В домашнем каталоге создайте новый каталог с именем newdir.
 - 3.2. В каталоге ~/newdir создайте новый каталог с именем morefun.
 - 3.3. В домашнем каталоге создайте одной командой три новых каталога с именами letters, memos, misk. Затем удалите эти каталоги одной командой.
 - 3.4. Попробуйте удалить ранее созданный каталог ~/newdir командой rm. Проверьте, был ли каталог удалён.
 - 3.5. Удалите каталог ~/newdir/morefun из домашнего каталога. Проверьте, был ли каталог удалён.
4. С помощью команды man определите, какую опцию команды ls нужно использовать для просмотра содержимого не только указанного каталога, но и подкаталогов, входящих в него.
5. С помощью команды man определите набор опций команды ls, позволяющий отсортировать по времени последнего изменения выводимый список содержимого каталога с развёрнутым описанием файлов.
6. Используйте команду man для просмотра описания следующих команд: cd, pwd, mkdir, rmdir, rm. Поясните основные опции этих команд.
7. Используя информацию, полученную при помощи команды history, выполните модификацию и исполнение нескольких команд из буфера команд.

Выполнение лабораторной работы

Определите полное имя вашего домашнего каталога. Далее относительно этого каталога будут выполняться последующие упражнения (рис. [-@fig:001])

```
[PROMofficialiPROMise@fedora ~]$ ls
abcl      cus      Documents feathers Music Pictures program.txt ski.plases Templates Videos
australia Desktop Downloads labs     my_os  play    Public   study_2021-2022_os-intro text.txt work
```

Перейдите в каталог /tmp. У меня его не было в домашней папке, поэтому пришлось создавать его заново.

Выведите на экран содержимое каталога /tmp. Для этого используйте команду ls с различными опциями (рис. [-@fig:002], [-@fig:003], [-@fig:004])

```
[PRoMofficialiPRoMise@fedora ~]$ ls
abc1      Desktop  feathers  my_os    program.txt  study_2021-2022_os-intro  tmp
australia Documents labs    Pictures  Public       Templates          Videos
cus       Downloads Music    play     ski.plases   text.txt           work
[PRoMofficialiPRoMise@fedora ~]$ ls tmp
[PRoMofficialiPRoMise@fedora ~]$ cd tmp
```

```
[PRoMofficialiPRoMise@fedora tmp]$ ls
[PRoMofficialiPRoMise@fedora tmp]$
```

```
[PRoMofficialiPRoMise@fedora tmp]$ ls
[PRoMofficialiPRoMise@fedora tmp]$ ls -a
. .
[PRoMofficialiPRoMise@fedora tmp]$ ls -l
итого 0
[PRoMofficialiPRoMise@fedora tmp]$ ls -F
[PRoMofficialiPRoMise@fedora tmp]$
```

Определите, есть ли в каталоге /var/spool подкаталог с именем cron? (Даже каталога var не было, поэтому я его создал) (рис. [-@fig:005])

```
[PRoMofficialiPRoMise@fedora ~]$ ls
abc1      Documents Music    program.txt  Templates  Videos
australia Downloads my_os     Public       text.txt   work
cus       feathers  Pictures  ski.plases   tmp
Desktop   labs     play     study_2021-2022_os-intro  var
[PRoMofficialiPRoMise@fedora ~]$ ls var/spool/cron
[PRoMofficialiPRoMise@fedora ~]$
```

В домашнем каталоге создайте новый каталог с именем newdir. (рис. [-@fig:006])

```
[PRoMofficialiPRoMise@fedora ~]$ mkdir newdir
[PRoMofficialiPRoMise@fedora ~]$ ls
abc1      Documents Music    play     study_2021-2022_os-intro  var
australia Downloads my_os     program.txt  Templates          Videos
cus       feathers  newdir    Public       text.txt           work
Desktop   labs     Pictures  ski.plases   tmp
```

В каталоге ~/newdir создайте новый каталог с именем morefun. (рис. [-@fig:007])

```
[PRoMofficialiPRoMise@fedora ~]$ mkdir newdir/morefun
[PRoMofficialiPRoMise@fedora ~]$ ls newdir/
morefun
```

В домашнем каталоге создайте одной командой три новых каталога с именами letters, memos, misk. Затем удалите эти каталоги одной командой. (рис. [-@fig:008], [-@fig:009])

```
[PRoMofficialiPRoMise@fedora ~]$ mkdir letters memos misk
[PRoMofficialiPRoMise@fedora ~]$ ls
abc1      Downloads misk     play     Templates  work
australia feathers Music    program.txt  text.txt
cus       labs    my_os     Public       tmp
Desktop   letters newdir    ski.plases   var
Documents memos    Pictures  study_2021-2022_os-intro  Videos
[PRoMofficialiPRoMise@fedora ~]$ rm -r letters memos misk
[PRoMofficialiPRoMise@fedora ~]$ ls
abc1      Documents Music    play     study_2021-2022_os-intro  var
australia Downloads my_os     program.txt  Templates          Videos
cus       feathers  newdir    Public       text.txt           work
Desktop   labs     Pictures  ski.plases   tmp
```

Попробуйте удалить ранее созданный каталог ~/newdir командой rm. Проверьте, был ли каталог удалён. (рис. [-@fig:010])

```
[PRoMofficialiPRoMise@fedora ~]$ rm -r newdir
[PRoMofficialiPRoMise@fedora ~]$ ls
abc1      Documents Music      program.txt      Templates  Videos
australia Downloads my_os    Public          text.txt    work
cus       feathers  Pictures ski.plases      tmp
Desktop   labs     play    study_2021-2022_os-intro var
```

Удалите каталог ~/newdir/morefun из домашнего каталога. Проверьте, был ли каталог удалён. (рис. [-@fig:011])

```
[PRoMofficialiPRoMise@fedora ~]$ rm -r newdir/morefun
rm: невозможно удалить 'newdir/morefun': Нет такого файла или каталога
```

Вызов мануалов к командам (рис. [-@fig:013], [-@fig:014], [-@fig:015], [-@fig:016], [-@fig:017], [-@fig:018], [-@fig:019])

LS(1)	User Commands	LS(1)
NAME		
ls - list directory contents		
SYNOPSIS		
ls [OPTION]... [FILE]...		
DESCRIPTION		
List information about the FILES (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.		
Mandatory arguments to long options are mandatory for short options too.		
-a, --all do not ignore entries starting with .		
-A, --almost-all do not list implied . and ..		
--author with -l , print the author of each file		
-b, --escape print C-style escapes for nongraphic characters		
--block-size=SIZE with -l , scale sizes by SIZE when printing them; e.g., '--block-size=M' ; see SIZE format below		
-B, --ignore-backups do not list implied entries ending with ~		
-c with -lt : sort by, and show, ctime (time of last modification of file status information); with -l : show ctime and sort by name; otherwise: sort by ctime, newest first		
-C list entries by columns		
--color[=WHEN] colorize the output; WHEN can be 'always' (default if omitted), 'auto', or 'never'; more info below		
-d, --directory list directories themselves, not their contents		
-D, --dired generate output designed for Emacs' dired mode		
-f do not sort, enable -aU , disable -ls --color		

```
--time-style=TIME STYLE
    time/date format with -l; see TIME_STYLE below

-t      sort by time, newest first; see --time
```

```
PRoMofficialiPRoMise@fedora:~ — man cd
BASH_BUILTINS(1)                                General Commands Manual                                BASH_BUILTINS(1)

NAME
bash, :, ., [, alias, bg, bind, break, builtin, caller, cd, command, compgen, complete, compopt, continue,
declare, dirs, disown, echo, enable, eval, exec, exit, export, false, fc, fg, getopts, hash, help, history,
jobs, kill, let, local, logout, mapfile, popd, printf, pushd, pwd, read, readonly, return, set, shift,
shopt, source, suspend, test, times, trap, true, type, typeset, ulimit, umask, unalias, unset, wait - bash
built-in commands, see bash(1)

BASH BUILTIN COMMANDS
Unless otherwise noted, each builtin command documented in this section as accepting options preceded by -
accepts -- to signify the end of the options. The :, true, false, and test/[ builtins do not accept options
and do not treat -- specially. The exit, logout, return, break, continue, let, and shift builtins accept
and process arguments beginning with - without requiring --. Other builtins that accept arguments but are
not specified as accepting options interpret arguments beginning with - as invalid options and require -- to
prevent this interpretation.

: [arguments]
    No effect; the command does nothing beyond expanding arguments and performing any specified redirec-
    tions. The return status is zero.

. filename [arguments]
source filename [arguments]
    Read and execute commands from filename in the current shell environment and return the exit status
    of the last command executed from filename. If filename does not contain a slash, filenames in PATH
    are used to find the directory containing filename. The file searched for in PATH need not be exe-
    cutable. When bash is not in posix mode, the current directory is searched if no file is found in
    PATH. If the sourcepath option to the shopt builtin command is turned off, the PATH is not searched.
    If any arguments are supplied, they become the positional parameters when filename is executed. Oth-
    erwise the positional parameters are unchanged. If the -T option is enabled, source inherits any
    trap on DEBUG; if it is not, any DEBUG trap string is saved and restored around the call to source,
    and source unsets the DEBUG trap while it executes. If -T is not set, and the sourced file changes
    the DEBUG trap, the new value is retained when source completes. The return status is the status of
    the last command exited within the script (0 if no commands are executed), and false if filename is
    not found or cannot be read.

alias [-p] [name[=value] ...]
    Alias with no arguments or with the -p option prints the list of aliases in the form alias name=value
    on standard output. When arguments are supplied, an alias is defined for each name whose value is
    given. A trailing space in value causes the next word to be checked for alias substitution when the
    alias is expanded. For each name in the argument list for which no value is supplied, the name and
    value of the alias is printed. Alias returns true unless a name is given for which no alias has been
    defined.

bg [jobspec ...]
    Resume each suspended job jobspec in the background, as if it had been started with &. If jobspec is
    not present, the shell's notion of the current job is used. bg jobspec returns 0 unless run when job
    control is disabled or, when run with job control enabled, any specified jobspec was not found or was
    started without job control.

bind [-m keymap] [-lpsvPSVX]
bind [-m keymap] [-q function] [-u function] [-r keyseq]
bind [-m keymap] -f filename
```

```
PWD(1)                                User Commands                                PWD(1)

NAME
    pwd - print name of current/working directory

SYNOPSIS
    pwd [OPTION]...

DESCRIPTION
    Print the full filename of the current working directory.

    -L, --logical
        use PWD from environment, even if it contains symlinks

    -P, --physical
        avoid all symlinks

    --help
        display this help and exit

    --version
        output version information and exit

    If no option is specified, -P is assumed.

    NOTE: your shell may have its own version of pwd, which usually supersedes the version described here.
    Please refer to your shell's documentation for details about the options it supports.

AUTHOR
    Written by Jim Meyering.

REPORTING BUGS
    GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
    Report any translation bugs to <https://translationproject.org/team/>

COPYRIGHT
    Copyright © 2020 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later
    <https://gnu.org/licenses/gpl.html>.
    This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent per-
    mitted by law.

SEE ALSO
    getcwd(3)

    Full documentation <https://www.gnu.org/software/coreutils/pwd>
    or available locally via: info '(coreutils) pwd invocation'

GNU coreutils 8.32                                July 2021                                PWD(1)
Manual page pwd(1) line 1/17 (END) (press h for help or q to quit)
```

```
[PRoMofficialiPRoMise@fedora ~]$ man ls
[PRoMofficialiPRoMise@fedora ~]$ man cd
[PRoMofficialiPRoMise@fedora ~]$ man pwd
[PRoMofficialiPRoMise@fedora ~]$ man mkdir
[PRoMofficialiPRoMise@fedora ~]$ man rmdir
[PRoMofficialiPRoMise@fedora ~]$
[PRoMofficialiPRoMise@fedora ~]$ man rm
```

```
[PRoMofficialiPRoMise@fedora ~]$ history
 1  cd
 2  touch abc1
 3  cp abc1 april
 4  cp abc1 may
 5  ls
 6  ls april
 7  cd april; ls; cd ~/may; ls
 8  mkdir monthly
 9  cd april may monthly
10  cp april may monthly
11  ls
12  ls monthly
13  cp monthly/may monthly/june
14  ls monthly
15  mkdir monthly.00
16  cp -r monthly monthly.00
17  ls monthly.00
18  ls monthly
19  ls
20  cp -r monthly.00 /tmp
21  ls /tmp
22  cd
23  mv april july
24  ls
25  mv july monthly.00
26  ls monthly.00
27  mv monthly.00/july monthly.00/monthly
28  ls monthly.00/monthly
29  mv monthly.00 monthly.01
30  ls
```

```
306  ls
307  rm -r newdir/morefun
308  man ls
309  man cd
310  man pwd
311  man mkdir
312  man rmdir
313  man rm
314  history
[PRoMofficialiPRoMise@fedora ~]$ ls var/spool/
```

Выводы

Я приобрел практические навыки взаимодействия пользователя с системой посредством командной строки.

Контрольные вопросы

1. Что такое командная строка?
2. При помощи какой команды можно определить абсолютный путь текущего каталога? Приведите пример.
3. При помощи какой команды и каких опций можно определить только тип файлов и их имена в текущем каталоге? Приведите примеры.

4. Каким образом отобразить информацию о скрытых файлах? Приведите примеры.
5. При помощи каких команд можно удалить файл и каталог? Можно ли это сделать одной и той же командой? Приведите примеры.
6. Каким образом можно вывести информацию о последних выполненных пользователем командах? работы?
7. Как воспользоваться историей команд для их модифицированного выполнения? Приведите примеры.
8. Приведите примеры запуска нескольких команд в одной строке.
9. Дайте определение и приведите примера символов экранирования.
10. Охарактеризуйте вывод информации на экран после выполнения команды `ls` с опцией `l`.
11. Что такое относительный путь к файлу? Приведите примеры использования относительного и абсолютного пути при выполнении какой-либо команды.
12. Как получить информацию об интересующей вас команде?
13. Какая клавиша или комбинация клавиш служит для автоматического дополнения вводимых команд?

Ответы на контрольные вопросы:

1. Интерфейс командной строки - управление программами с помощью команд. Команды состоят из букв, цифр, символов, набираются построчно, выполняются после нажатия клавиши Enter. Основным инструментом здесь является клавиатура. Данный интерфейс встроен в ядро системы, он будет доступен, даже если графический интерфейс не запустится. Добраться до командной строки можно двумя способами: через консоль или терминал.

2. При помощи команды `realpath` можно определить абсолютный путь текущего каталога. Например, если ввести `realpath /var` на экран выведется `/home//var`.

3. При помощи команды `ls -F` можно определить только тип файлов и их имена в текущем каталоге.
4. Файл (или директория) считается скрытым, если его название начинается с символа точка «.». Например, «.myfile». Обычно такие файлы используются приложениями для хранения настроек, конфигураций и другой информации, которую нужно скрыть от пользователя. Зачастую пользователю требуется отредактировать соответствующий конфигурационный скрытый файл, чтобы настроить какую-нибудь программу, и пользователи сталкиваются с тем, что не знают, как их вообще просмотреть. По умолчанию файловые менеджеры обычно не отображают такие файлы.
Для просмотра списка файлов в командной строке используется команда `ls`. Чтобы по команде `ls` также выводились скрытые файлы, существует опция `-a`.
5. При помощи команд `rm` и `rmdir` можно удалить файл и каталог. Это нельзя сделать одной и той же командой. `rmdir` используется, чтобы удалить файлы, которые должны быть пустые. `rm` используется, чтобы удалить непустые файлы или целые деревья каталогов.
6. Определить какие команды выполнил пользователь в сеансе работы можно с помощью команды `history`.
7. Исправить и запустить на выполнение команду, которую пользователь уже использовал в сеансе работы, можно с помощью команды: `!`

`<номеркоманды>:s/<чтоменяем>/<начтоменяем>` Например,
`history`

.

3 `ls -a`

```
.  
.
!3:s/a/F
ls -F
```

8. В одной строке можно записать несколько команд. Если требуется выполнить последовательно несколько команд, записанный в одной строке, то для этого используется символ точка с запятой. Пример: `cd; ls`.
9. Экранирование — это способ заключения в кавычки одиночного символа. Экранирующий символ (`\`) сообщает интерпретатору, что следующий за ним символ должен восприниматься как обычный символ. Пример:
`echo "Привет" # Привет`
`echo "Он сказал: "Привет"." # Он сказал: "Привет".`
9. Если используется опция `l` в команде `ls`, то на экран выводится подробный список, в котором будет отображаться владелец, группа, дата создания, размер и другая информация о файлах и каталогах.
10. Относительный путь – это путь к файлу относительно текущей папки. При использовании команды `pwd` на экран выведется относительный путь текущей директории, а при использовании команды `realpath` на экран выведется абсолютный путь текущей директории.
11. Получить информацию об интересующей вас команде можно с помощью команды `man`. Например, команда `man ls` выведет все опции команды `ls`.
12. Сочетание клавиш `Ctrl+C` прерывает текущий процесс, запущенный в терминале.

Список литературы{.unnumbered}

```
::: {#refs}  
:::
```

1. [Командная строка](#)