

РУДН. Операционные системы

Отчёт по лабораторной работе №6

Косинов Никита Андреевич, НПИМбв-02-20

Содержание

1	Цель работы	5
2	Ход работы	6
3	Перенаправление ввода-вывода	7
4	Фильтрация текста	10
5	Поиск файла	12
6	Обработка информации о процессах	14
7	Выводы	20

Список иллюстраций

3.1	Синхронизация рабочего пространства с git	7
3.2	Перенаправление вывода	7
3.3	Просмотр содержимого документа	7
3.4	Просмотр содержимого документа	8
3.5	Дозапись с перенаправлением	8
3.6	Просмотр содержимого документа	9
4.1	Поиск файлов определённого расширения	10
4.2	Открытие файла	10
4.3	Содержимое файла	11
5.1	Поиск файла в каталоге по маске	12
5.2	Результат поиска	12
5.3	Поиск файла в созданном списке	13
5.4	Поиск файла по маске конвейером	13
5.5	Поиск файла по маске конвейером	13
6.1	Запуск фонового процесса	14
6.2	Просмотр содержимого logfile	15
6.3	Проверка запущенных процессов командой jobs	15
6.4	Удаление файла	15
6.5	Запуск фонового процесса	16
6.6	Вывод идентификаторов процессов	16
6.7	Вывод идентификаторов процессов	16
6.8	Поиск идентификатора конкретного процесса	17
6.9	Завершение процесса	17
6.10	Использование команды df	17
6.11	Использование команды df	17
6.12	Использование команды du	18
6.13	Использование команды du	18
6.14	Использование команды du	19
6.15	Чтение документпции команды	19
6.16	Вызов find с модификатором	19
6.17	Результат: каталоги домашней директории	19

Список таблиц

1 Цель работы

В прошлой лабораторной работе мы познакомились со структурой файловой системы. Кроме открытия, чтения и работы с правами доступа необходимо быстро находить необходимые документы, и знать, как их записывать, какие команды и как можно применять к дереву каталогов.

Цель данной работы - приобретение теоретических и практических навыков по работе с файловой системой из командной строки на примере ОС **Linux**.

2 Ход работы

Лабораторная работа выполнена в терминале **ОС Linux** и хостинге хранения проектов **Github**. Действия по лабораторной работе представлены в следующем порядке:

1. перенаправление ввода-вывода;
2. фильтрация текста;
3. поиск файла;
4. обработка информации о процессах.

По завершении отчёта, вся рабочая папка отправляется на репозиторий на *github*.

3 Перенаправление ввода-вывода

Перед началом работы, по обычаю, получим обновления из **git** командой *git pull*, и создадим рабочую папку **lab02-06** командой *mkdir*.

```
nakosinov@dk8n81 ~ $ cd work/study/2022-2023/Операционные\ системы/os-intro/  
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro $ git pull  
Уже актуально.
```

Рис. 3.1: Синхронизация рабочего пространства с git

Выполняем примеры из лабораторной работы по перенаправлению вывода.

1. Записываем все файлы, содержащиеся в каталоге **/etc**, используя стандартную команду *ls* вывода содержимого с применением перенаправления записи *>* в файл **file.txt**.

```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro $ cd labs/lab06  
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ ls  
/etc > file.txt
```

Рис. 3.2: Перенаправление вывода

2. Просмотрим полученный результат командой *cat*.

```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ cat  
file.txt
```

Рис. 3.3: Просмотр содержимого документа

```
udmcpd.conf
udisks2
ufw
unbound
unixODBC
updatedb.conf
UPower
vbox
vconsole.conf
vde2
vdpau_wrapper.cfg
vhosts
vim
vpl
vulkan
wgetrc
whois.conf
wireless-regdb
wireplumber
wpa_supplicant
X11
xattr.conf
xdg
xinetd.conf
xinetd.d
x12tpd
xml
xrootd
zsh
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $
```

Рис. 3.4: Просмотр содержимого документа

3. Дозапишем в тот же файл все файлы из домашней директории. Для дозаписи используем команду »

```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ ls
~ >> file.txt
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $
```

Рис. 3.5: Дозапись с перенаправлением

4. Просмотрим содержимое, чтобы убедиться, что команда выполнена корректно.

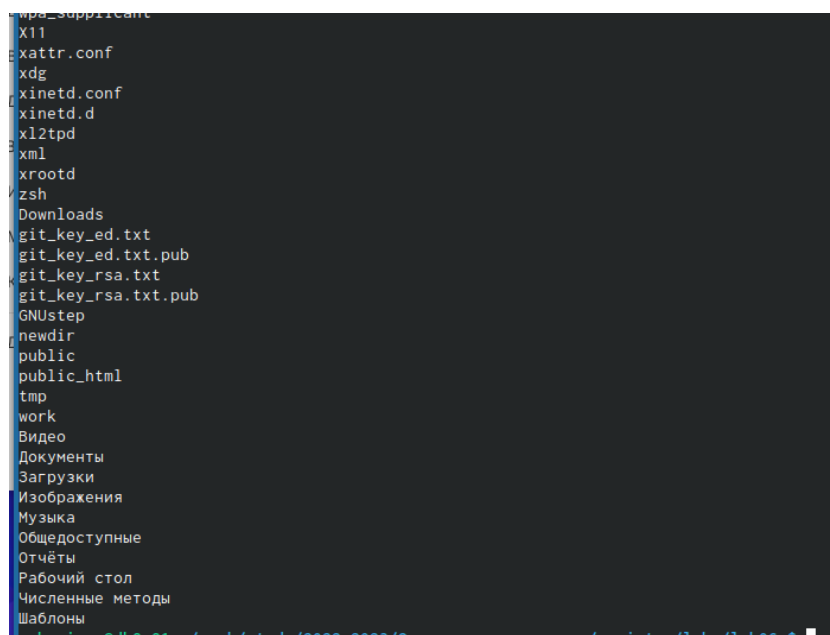
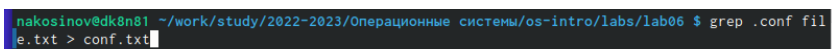


Рис. 3.6: Просмотр содержимого документа

4 Фильтрация текста

Данная часть работы посвящена поиску файлов с определёнными текстовыми маркерами.

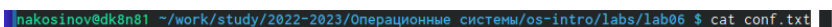
1. Командой *grep* ищем в недавно созданном **file.txt** все файлы с расширением **.conf**: это сочетание символов должно быть в названии. Запишем результаты поиска в новый файл **conf.txt**.



```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ grep .conf file.txt > conf.txt
```

Рис. 4.1: Поиск файлов определённого расширения

2. Открываем **conf.txt**, чтобы убедиться в корректности выполнения команды.



```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ cat conf.txt
```

Рис. 4.2: Открытие файла

```
pear.conf
pe-format2.conf
pmount.conf
prelink.conf.d
pump.conf
pwdb.conf
rc.conf
request-key.conf
resolv.conf
rofi-pass.conf
rsyncd.conf
rsyslog.conf
sandbox.conf
sddm.conf
sddm.conf.d
sensors3.conf
signond.conf
smartd.conf
strongswan.conf
sudo.conf
sudo_logsrvd.conf
swtpm-localca.conf
swtpm_setup.conf
sysconfig
systemconfig
udhcpd.conf
updatedb.conf
vconsole.conf
whois.conf
xattr.conf
xinetd.conf
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $
```

Рис. 4.3: Содержимое файла

5 Поиск файла

Чаще при работе с файловой структурой требуется натйти определённый файл, нежели найти символы в списке. Поэтому выполним несколько заданий на поиск.

1. Первое задание: найти файлы в домашнем каталоге, начинающиеся на символ **с**. Первая мысль - сделать это командой *find*, применённой к домашней директории с фильтром ****с*** и вывести на экран модификатором *-print*.

```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ find ~ -name "с*" -print
```

Рис. 5.1: Поиск файла в каталоге по маске

```
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/tem
plate/report/report/pandoc/csl
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/tem
plate/report/report/pandoc/filters/pandocxnos/core.py
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab02-03/report/report/bib/cite.bib
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab02-03/report/report/pandoc/csl
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab02-03/report/report/pandoc/filters/pandocxnos/core.py
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab06/report/report/bib/cite.bib
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab06/report/report/pandoc/csl
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab06/report/report/pandoc/filters/pandocxnos/core.py
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab06/conf.txt
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab04/report/report/bib/cite.bib
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab04/report/report/pandoc/csl
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab04/report/report/pandoc/filters/pandocxnos/core.py
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab05/report/report/bib/cite.bib
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab05/report/report/pandoc/csl
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2022-2023/Операционные системы/os-intro/lab
s/lab05/report/report/pandoc/filters/pandocxnos/core.py
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $
```

Рис. 5.2: Результат поиска

2. Как видно выше, команда *find* ищет вообще все файлы внутри каталога, обладающие нужным свойством, не только в корне этого каталога. Поэтому попробуем сделать по другому - командой *grep* в предварительно созданном файле *home.txt*

```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ ls ~ > home.txt
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ grep ^c home.txt
code.txt
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $
```

Рис. 5.3: Поиск файла в созданном списке

3. Однако, создавать список в отдельный файл нецелесообразно. Здесь на помощь приходит конвейер, позволяющий передать в *grep* список файлов напрямую, без промежуточной записи.

```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ ls ~ | grep ^c
code.txt
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $
```

Рис. 5.4: Поиск файла по маске конвейером

4. Аналогичным образом, с помощью *grep* и конвейера выведем файлы, начинающиеся на **h** в каталоге */etc*.

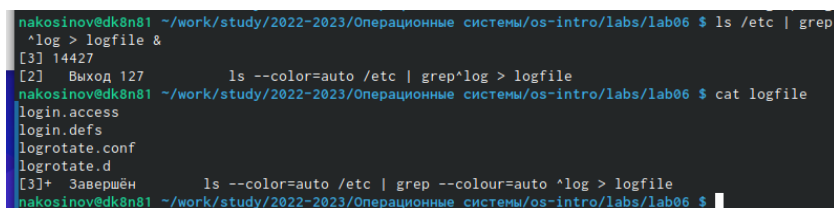
```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ ls /etc | grep ^h
^h
hal
harbour
harbour.cfg
highlight
host.conf
hostname
hosts
hosts.allow
hotplug
hotplug.d
hsqldb
htdig
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $
```

Рис. 5.5: Поиск файла по маске конвейером

6 Обработка информации о процессах

При работе с ЭВМ зачастую требуется управлять процессами: например, чтобы развести их по потокам для параллельной работы. Эффект хорошо виден, когда, например, на мониторе открыто несколько окон. Данная часть работы посвящена управлению и поиску процессов

1. Запустим процесс, записывающий в файл **logfile** все файлы каталога **/etc**, начинающиеся с **log**. Для этого поставим в конец команды **&**.



```
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ ls /etc | grep ^log > logfile &
[3] 14427
[2] Выход 127      ls --color=auto /etc | grep ^log > logfile
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ cat logfile
login.access
login.defs
logrotate.conf
logrotate.d
[3]+  Завершён      ls --color=auto /etc | grep --colour=auto ^log > logfile
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $
```

Рис. 6.1: Запуск фонового процесса

2. Откроем файл для просмотра. На предыдущем скриншоте хорошо видно, что процесс завершился автоматически, как только мы открыли файл на чтение. Действительно, нельзя одновременно и записывать, и читать документ.

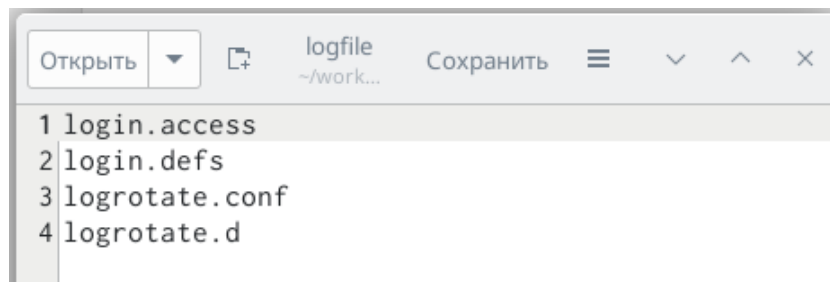


Рис. 6.2: Просмотр содержимого logfile

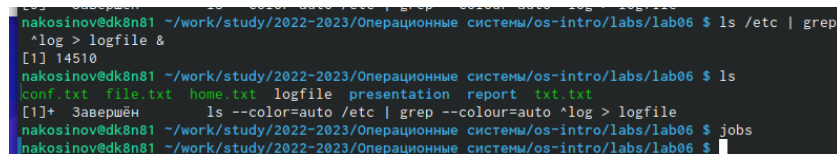


Рис. 6.3: Проверка запущенных процессов командой jobs

3. Удалим файл **logfile**.

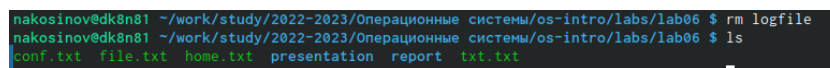


Рис. 6.4: Удаление файла

4. Запустим **gedit** в фоновом режиме. Окно редактора откроется поверх консоли, а процессу присвоится идентификатор, который видно внизу скриншота - 14877.

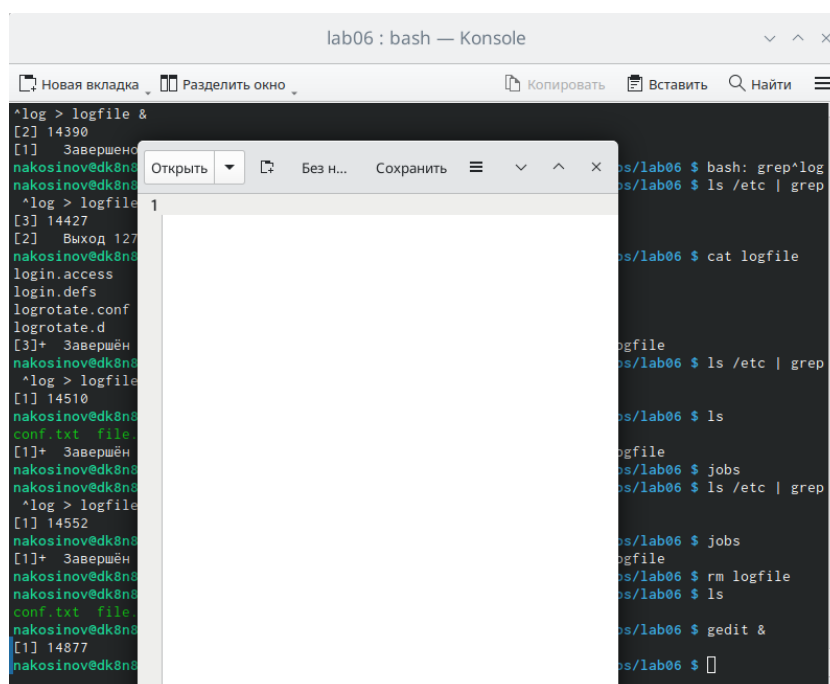


Рис. 6.5: Запуск фонового процесса

- Можно получить идентификатор недавно запущенного процесса командой *ps aux*, и промотав в конец списка.

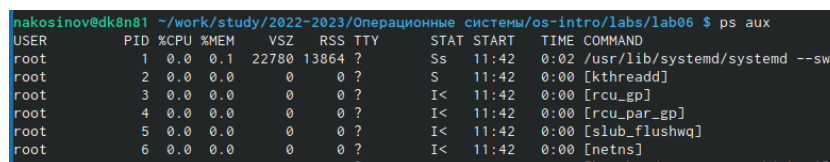


Рис. 6.6: Вывод идентификаторов процессов

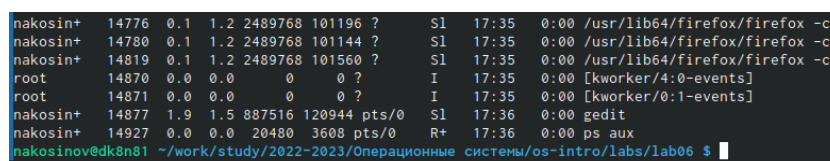


Рис. 6.7: Вывод идентификаторов процессов

- Также, можно получить идентификатор командой *grep*, соединённой конвейером с *ps aux*: так можно определить идентификаторы всех процессов, включающих в название **gedit**.


```

nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ ps aux | grep
gedit
nakosin+ 14877  0.6  1.5 887688 121384 pts/0    Sl   17:36   0:00 gedit
nakosin+ 14970  0.0  0.0   6800   2320 pts/0    S+   17:38   0:00 grep --colour=auto gedit
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $

```

Рис. 6.8: Поиск идентификатора конкретного процесса

7. Проверим командой *jobs* запущенные процессы. Видим, что нашему при-
своен номер **1** - пока это единственный наш фоновый процесс. Завершим
его командой *kill*. Снова проверим список наших процессов. Видим, что
команда выполнена успешно.

```

nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ jobs
[1]+  Запущен      gedit &
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ kill %1
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ jobs
[1]+  Завершено      gedit
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $

```

Рис. 6.9: Завершение процесса

8. Просмотрим, какие файловые системы занимают больше всего места. Для
этого воспользуемся командой *df*. Модификатор *-vi* покажет инод с игно-
рируемыми системами, а модификатор *-l* - предел локальной системы.

```

nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ df -vi
Файловая система  Индов  Использовано  ISвободно  IИспользовано%  Смонтировано в
/dev/sda8          33447936  1779298      31668638      6% /
devtmpfs           986729   1623         985106        1% /dev
tmpfs              999894   1          999893        1% /dev/shm
tmpfs              819200   1888        817312        1% /run
tmpfs              1048576  4455        1044121       1% /tmp
AFS                2147483647 0 2147483647  0% /afs
tmpfs              199978   194         199784        1% /run/user/5164
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $

```

Рис. 6.10: Использование команды *df*

```

nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ df -l
Файловая система  1К-блоков  Использовано  Доступно  Использовано%  Смонтировано в
/dev/sda8          525626596  105652348    393200456    22% /
devtmpfs           4096       0            4096         0% /dev
tmpfs              3999576    0            3999576      0% /dev/shm
tmpfs              1599832    24184       1575648      2% /run
tmpfs              3999580    327992      3671588      9% /tmp
tmpfs              799912     1440        798472       1% /run/user/5164
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $

```

Рис. 6.11: Использование команды *df*

9. Командой *du* можно увидеть отчёт об использовании пространства кон-
кретными файлами. Модификатор *-a* покажет все файлы, а не только ди-
ректории.

```

84 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image/31.png
10 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image/32.png
12 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image/30.png
79 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image/21.png
40 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image/25.png
36 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image/26.png
38 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image/27.png
7 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image/28.png
1370 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image
1233 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/image.zip
17 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/report.md
824 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/Л01_КосиновН_02-20_отчёт.pdf
1344 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/Л02_КосиновН_02-20_отчёт.pdf
196 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/004-lab_shell.pdf
196 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/004-lab_shell-1.pdf
2 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/Makefile
1251 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/report.docx
1286 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки/report.pdf
7721 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/Загрузки
1 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/git_key_rsa.txt.pub
1 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/git_key_ed.txt.pub
2 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/tmp
1 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/GNUstep/Library/Services/.GNUstepServices
1 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/GNUstep/Library/Services/.GNUstepAppList
4 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/GNUstep/Library/Services
6 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/GNUstep/Library
8 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/GNUstep
2 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/newdir/news
4 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/newdir
0 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/code.txt
272933 /afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ du -a ~/

```

Рис. 6.12: Использование команды du

```

59 ./report/report/pandoc/filters/pandocxnos
20 ./report/report/pandoc/filters/pandoc_eqnos.py
25 ./report/report/pandoc/filters/pandoc_fignos.py
13 ./report/report/pandoc/filters/pandoc_secnos.py
27 ./report/report/pandoc/filters/pandoc_tablenos.py
146 ./report/report/pandoc/filters
163 ./report/report/pandoc
13 ./report/report/report.md
2 ./report/report/Makefile
956 ./report/report/report.docx
970 ./report/report/report.pdf
940 ./report/report/image.zip
3055 ./report/report
1 ./report/scripts/image-report
1 ./report/scripts/mpv-shot
4 ./report/scripts
1 ./report/.git
2 ./report/.gitattributes
5 ./report/.gitignore
3 ./report/CHANGELOG.md
19 ./report/LICENSE
6 ./report/README.git-flow.md
1 ./report/README.md
1 ./report/README.ru.md
1 ./report/package.json
3100 ./report
5 ./file.txt
2 ./conf.txt
0 ./txt.txt
1 ./home.txt
3210 .
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ du -a

```

Рис. 6.13: Использование команды du

```

nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ man du
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ du -bc
47359  ./presentation/presentation/image
55403  ./presentation/presentation
88730  ./presentation
4866   ./report/report/bib
0      ./report/report/image
12853  ./report/report/pandoc/csl
56428  ./report/report/pandoc/filters/pandocxnos
141560 ./report/report/pandoc/filters
154413 ./report/report/pandoc
3106327 ./report/report
953    ./report/scripts
3140997 ./report
3235727 .
3235727  итого
nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $

```

Рис. 6.14: Использование команды `du`

10. Прочтя мануал команды `find`, и применив модификатор `type -d`, выведем все каталоги, лежащие в домашней директории.

```

nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ man find

```

Рис. 6.15: Чтение документации команды

```

nakosinov@dk8n81 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab06 $ find ~ -type -d
-print

```

Рис. 6.16: Вызов `find` с модификатором

```

it/objects/a3/2506c72f9ba5358e23c8584683c95f14ae09e6
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/a3
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/cd/f85b5955f1336ba3c5b0fde9dffe421267be5
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/cd/96abc8545181bb86143e2badbde87b146da27
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/cd
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/12/8ee966d7565babec06eedc83d2707cddfbbad7
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/12/c8081faafda165787681983681c1a8827f1a73
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/12/0b97390f5a7d64b9ed97c04e4d2b39c86a5550
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/12
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/c6/eb02bf698136aae59b5aa239d3fda97f7c71e0
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/c6/04905a4fb1f2c9793eb5350e81a8f7824e48ea
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/c6
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/c5/ec71896c4cf0582034c4ccbe36d86e5abc05f7
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/c5/1380551e0c05b6c8e450a1f817a3354300b13a
/afs/.dk.sci.pfu.edu.ru/home/n/a/nakosinov/work/study/2023-2024/Архитектура компьютера/arch-pc/.g
it/objects/c5
^C

```

Рис. 6.17: Результат: каталоги домашней директории

7 Выводы

Терминал предоставляет пользователю возможность получение информации о состоянии файловой системы: поиск файлов, перенаправление вывода полученной информации, управление процессами и т.п.