

# **Лабораторная работа №8**

**Отчёт**

Александр Денисович Мосолов

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение</b>	<b>7</b>
<b>4</b>	<b>Контрольные вопросы</b>	<b>11</b>
<b>5</b>	<b>Выводы</b>	<b>14</b>
	<b>Список литературы</b>	<b>15</b>

## Список иллюстраций

3.1	Выведите имена всех файлов из file.txt, имеющих расширение .conf	7
3.2	Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа с . . . . .	7
3.3	Вариант нахождения файлов с помощью ls . . . . .	7
3.4	Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h . . . . .	8
3.5	Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы . . . . .	8
3.6	Удалите файл ~/logfile . . . . .	8
3.7	Запустите из консоли в фоновом режиме редактор gedit . . . . .	8
3.8	Определите идентификатор процесса gedit, используя jobs . . . . .	8
3.9	Определите идентификатор процесса gedit, используя команду ps	9
3.10	Определите идентификатор процесса gedit, используя команду pidof	9
3.11	используйте kill для завершения процесса gedit . . . . .	9
3.12	Выполните команды df . . . . .	9
3.13	Выполните команды du . . . . .	10
3.14	выведите имена всех директорий . . . . .	10

## Список таблиц

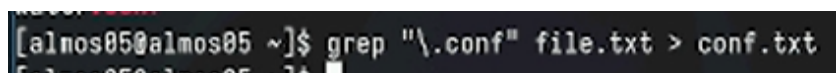
# 1 Цель работы

Ознакомление с инструментами поиска файлов и фильтрации текстовых данных. Приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.[1]

## **2 Задание**

## 3 Выполнение

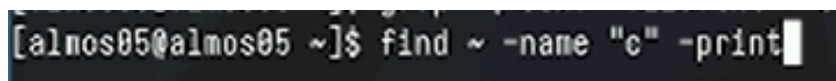
Выведите имена всех файлов из file.txt, имеющих расширение .conf, после чего запишите их в новый текстовый файл conf.txt. (рис. 3.1)



```
[alnos05@alnos05 ~]$ grep "\.conf" file.txt > conf.txt
```

Рис. 3.1: Выведите имена всех файлов из file.txt, имеющих расширение .conf

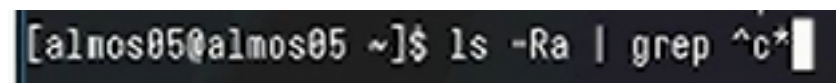
Определите, какие файлы в вашем домашнем каталоге имеют имена, начинающиеся с символа c? (рис. 3.2)



```
[alnos05@alnos05 ~]$ find ~ -name "c" -print
```

Рис. 3.2: Определите, какие файлы в вашем домашнем каталоге имеют имена, начинающиеся с символа c

Определите, какие файлы в вашем домашнем каталоге имеют имена, начинающиеся с символа c? (рис. 3.3)



```
[alnos05@alnos05 ~]$ ls -Ra | grep ^c*
```

Рис. 3.3: Вариант нахождения файлов с помощью ls

Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h (рис. 3.4)

```
..
[almos05@almos05 ~]$ find ~ -name "h*" -print | less
```

Рис. 3.4: Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h

Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log (рис. 3.5)

```
[almos05@almos05 ~]$ find ~ -name "log*" -print >> logfile &
[1] 2433
```

Рис. 3.5: Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы

Удалите файл ~/logfile (рис. 3.6)

```
[almos05@almos05 ~]$ rm logfile
```

Рис. 3.6: Удалите файл ~/logfile

Запустите из консоли в фоновом режиме редактор gedit (рис. 3.7)

```
[almos05@almos05 ~]$ gedit &
```

Рис. 3.7: Запустите из консоли в фоновом режиме редактор gedit

Определите идентификатор процесса gedit, используя jobs. (рис. 3.8)

```
[almos05@almos05 ~]$ jobs | grep gedit
[1]+  Завершён          gedit
[1]+  Завершён          gedit
```

Рис. 3.8: Определите идентификатор процесса gedit, используя jobs

Определите идентификатор процесса gedit, используя команду ps (рис. 3.9)



```
[almos05@almos05 ~]$ ps | grep gedit
2530 pts/0    00:00:00 gedit
```

Рис. 3.9: Определите идентификатор процесса gedit, используя команду ps

Определите идентификатор процесса gedit, используя команду pidof (рис. 3.10)

```
[almos05@almos05 ~]$ pidof gedit
2530
```

Рис. 3.10: Определите идентификатор процесса gedit, используя команду pidof

Прочтите справку (man) команды kill, после чего используйте её для завершения процесса gedit (рис. 3.11)

```
[almos05@almos05 ~]$ kill 2530
[almos05@almos05 ~]$
```

Рис. 3.11: используйте kill для завершения процесса gedit

Выполните команды df и du, предварительно получив более подробную информацию об этих командах (рис. 3.12)

```
[almos05@almos05 ~]$ df
Файловая система 1K-блоков  Использовано  Доступно  Использовано%
Смонтировано в
/dev/sda3          82834432    13840188  68041812          17%
```

Рис. 3.12: Выполните команды df

Выполните команды df и du, предварительно получив более подробную информацию об этих командах (рис. 3.13)

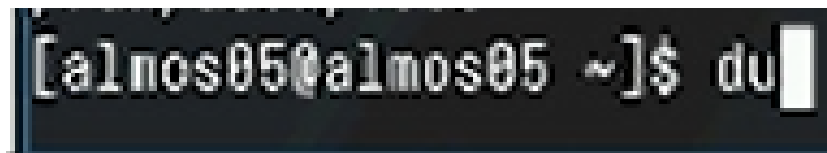


Рис. 3.13: Выполните команды du

Воспользовавшись справкой команды find, выведите имена всех директорий, имеющихся в вашем домашнем каталоге (рис. 3.14)

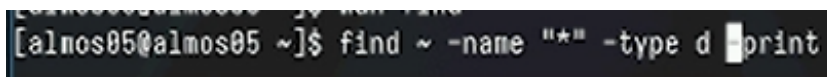


Рис. 3.14: выведите имена всех директорий

## 4 Контрольные вопросы

### 1. Потоки ввода/вывода

Стандартный ввод (stdin): Получает данные, вводимые пользователем или из другого потока ввода. Обычно связан с клавиатурой. Стандартный вывод (stdout): Выводит данные на экран или в другой поток вывода. Обычно связан с терминалом. Стандартная ошибка (stderr): Выводит сообщения об ошибках. Обычно связан с терминалом, но может быть перенаправлен.

### 2. Операции > и »

(перенаправление вывода): Перенаправляет вывод команды в файл. Заменяет содержимое файла. > (дополнение вывода): Перенаправляет вывод команды в файл. Дополняет содержимое файла.

### 3. Конвейер

Конвейер соединяет несколько команд таким образом, что вывод первой команды становится входом для второй команды и так далее. В Linux оболочке конвейеры создаются с помощью символа вертикальной черты (|).

### 4. Процесс и программа

Программа: Набор инструкций, которые должны быть выполнены. Процесс: Выполняющийся экземпляр программы. Каждый процесс имеет свой PID (идентификатор процесса).

## 5. PID и GID

PID (идентификатор процесса): Уникальный номер, присваиваемый каждому работающему процессу. GID (идентификатор группы): Уникальный номер, присваиваемый группе пользователей.

## 6. Задачи и команда управления

Задача: Объект, представляющий состояние и работу процесса. Команда управления задачами: `ps`

## 7. утилиты `top` и `htop`

`top`: Интерактивная утилита мониторинга системы, показывающая информацию о процессах, использовании памяти и процессора. `htop`: Расширенная альтернатива `top` с более удобным пользовательским интерфейсом и дополнительными функциями.

## 8. Команда поиска файлов

`find`: Позволяет рекурсивно искать файлы в указанной файловой системе на основе заданных критериев. Примеры: `find . -name "file.txt"`: Найти файл с именем "file.txt" в текущем каталоге и всех его подкаталогах. `find . -type f -size +100M`: Найти все обычные файлы размером более 100 МБ в текущем каталоге и его подкаталогах.

## 9. Поиск файлов по контексту

Да, можно искать файлы по контексту с помощью команды `grep`.

Синтаксис: `grep "шаблон" имя_файла`

Пример: `grep "слово" file.txt`

## 10. Определение объема свободной памяти на жестком диске

`df -h`: Выводит информацию о доступном дисковом пространстве во всех смонтированных файловых системах.

11. Определение объема домашнего каталога

`du -sh ~`: Выводит размер домашнего каталога и всех его подкаталогов.

12. Удаление зависшего процесса

`kill -9 PID`: Принудительно завершает процесс с указанным PID. Не используйте это для критических системных процессов.

## 5 Выводы

В данной работе мы ознакомились с инструментами поиска файлов и фильтрации текстовых данных. Приобрели практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

## Список литературы

1. Kulyabov. Архитектура компьютеров и операционные системы. Раздел "Операционные системы" (09.03.03, НПИбд).