Лабораторная работа №8

Операционные системы

Тойчубекова Асель Нурлановна

Содержание

# 1 Цель работы

Целью данной лабораторной работы является ознакомление с инструментами поиска файлов и фильтрации текстовых данных. Также приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

# 2 Задание

Выполнить все указания к лабораторной работе:  
1. Осуществите вход в систему, используя соответствующее имя пользователя.  
2. Запишите в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.  
3. Выведите имена всех файлов из file.txt, имеющих расширение .conf, после чего запишите их в новый текстовой файл conf.txt.  
4. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа c? Предложите несколько вариантов, как это сделать.  
5. Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h.  
6. Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.  
7. Удалите файл ~/logfile.  
8. Запустите из консоли в фоновом режиме редактор gedit.  
9. Определите идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. Как ещё можно определить идентификатор процесса?  
10. Прочтите справку (man) команды kill, после чего используйте её для завершения процесса gedit.  
11. Выполните команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man.  
12. Воспользовавшись справкой команды find, выведите имена всех директорий, имеющихся в вашем домашнем каталоге.

# 3 Теоретическое введение

В системе по умолчанию открыто три специальных потока: – **stdin** — стандартный поток ввода (по умолчанию: клавиатура), файловый дескриптор 0;

– **stdout** — стандартный поток вывода (по умолчанию: консоль), файловый дескриптор 1;

– **stderr** — стандартный поток вывод сообщений об ошибках (по умолчанию: консоль), файловый дескриптор 2.

Большинство используемых в консоли команд и программ записывают результаты своей работы в стандартный поток вывода stdout. Например, команда ls выводит в стандартный поток вывода (консоль) список файлов в текущей директории. Потоки вывода и ввода можно перенаправлять на другие файлы или устройства. Проще всего это делается с помощью символов >, >>, <, <<.

**Конвейер** (pipe) служит для объединения простых команд или утилит в цепочки, в которых результат работы прудыдущей команды передается поседующей. Для этого используется символ “|”.

**Команда find** используется для поиска и отображения на экран имён файлов, соответствующих заданной строке символов. Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск.

Найти в текстовом файле указанную строку символов позволяет **команда grep**.

**Команда df** показывает размер каждого смонтированного раздела диска.

**Команда du** показывает число килобайт, используемое каждым файлом или каталогом.

Любую выполняющуюся в консоли команду или внешнюю программу можно запустить в фоновом режиме. Для этого следует в конце имени команды указать **знак амперсанда &**

Запущенные фоном программы называются задачами (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент задач. Для завершения задачи необходимо выполнить **команду kill**

Любой команде, выполняемой в системе, присваивается идентификатор процесса (process ID). Получить информацию о процессе и управлять им, пользуясь идентификатором процесса, можно из любого окна командного интерпретатора.

**Команда ps** используется для получения информации о процессах.

# 4 Выполнение лабораторной работы

Для начала осуществляю вход в систему, используя соответствующее имя пользователя.

Записыва в файл file.txt названия файлов, содержащихся в каталоге /etc. С помощью утилиты head, которая выводит первые 10 строчек содержания файла на экран проверяю, что все имена файлов были записаны. (рис. 1).

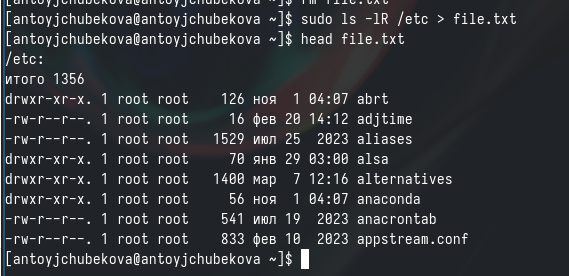


Рис. 1: Запись названий файлов

Затем добавляю в этот же файл названия файлов, содержащихся в моем домашнем каталоге (рис. 2). Используя команду cat вывожу на экран содержимое файла file.txt и вижу, что все названия файлов из дмащнего каталога были записаны в файл (рис. 3).

Запись названий файлов из корневого каталога

Рис. 2: Запись названий файлов из корневого каталога

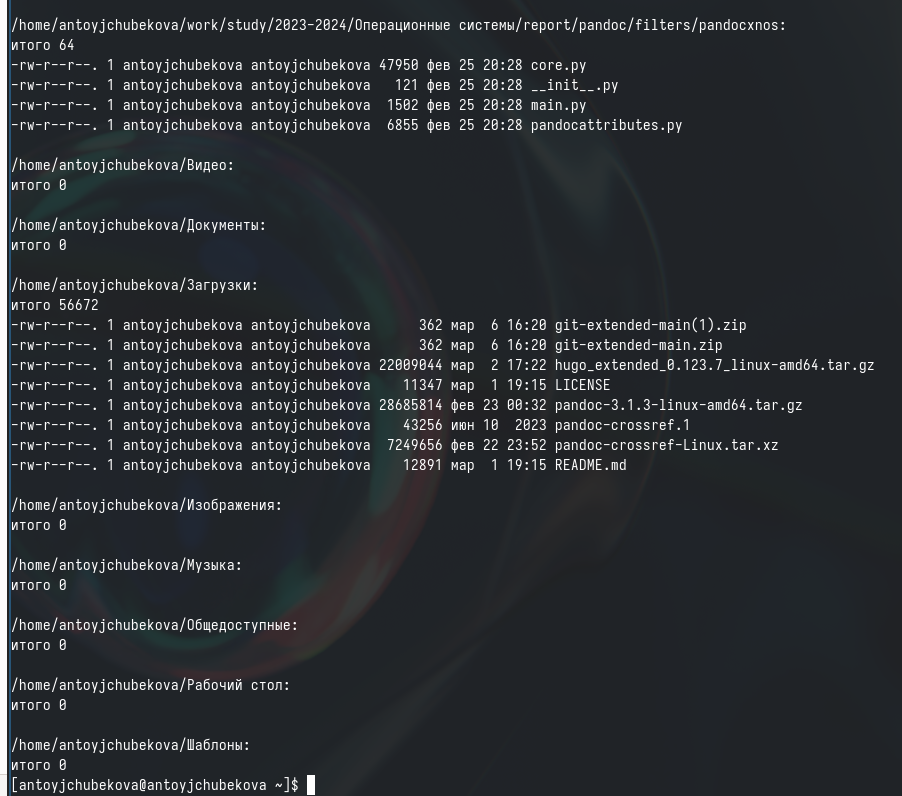


Рис. 3: Содержимое файла file.txt

Вывожу имена всех файлов из file.txt, имеющих расширение .conf, с помощью утилиты grep. (рис. 4).

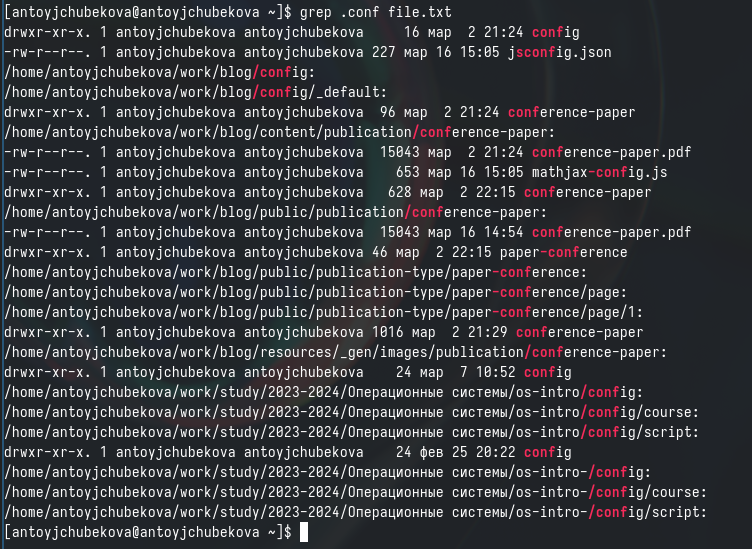


Рис. 4: Сортировка и вывод названий файлов с расширением .conf

Далее записываю их в новый текстовый файл conf.txt, используя символ “>”. С помощью команды head проверяю их наличия в файле. (рис. 5).

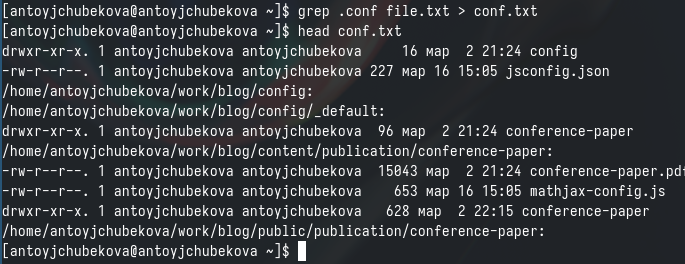


Рис. 5: Редактирование файла conf.txt

Определяю, какие файлы в домашнем каталоге начинаются с символа “c” с помощью утилиты find, прописываю ей в аргументы домашнюю директорию , такде опции -name и -print для того, чтобы команда искала файлы по имени и выводила их на экран. (рис. 6 и рис. 7)

Поиск файлов, начинающихся с определенного элемента

Рис. 6: Поиск файлов, начинающихся с определенного элемента

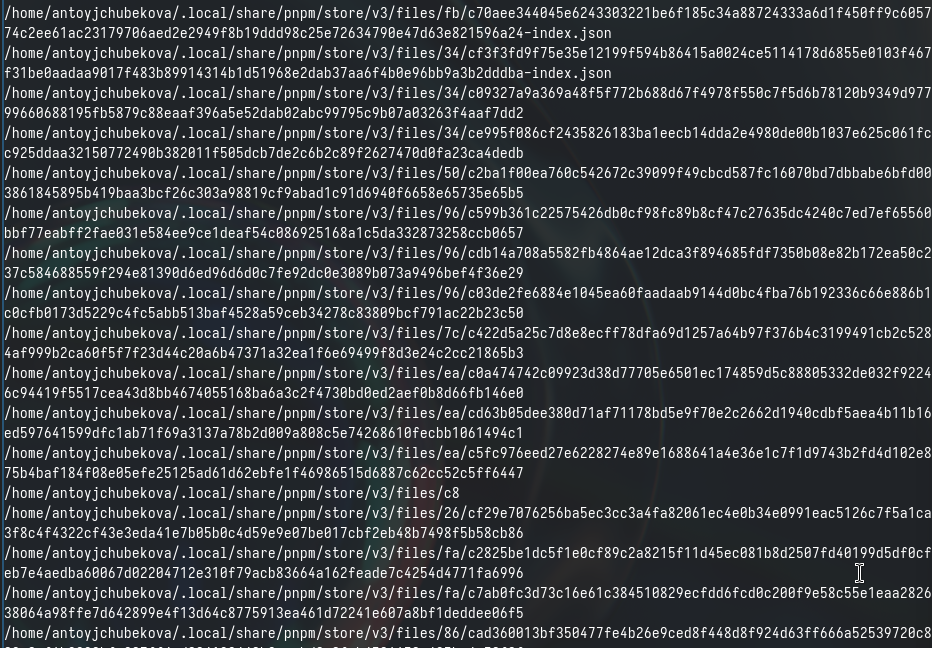


Рис. 7: Поиск файлов, начинающихся с определенного элемента

Второй способ заключается в использовании утилиты ls -lR и использовать grep, чтобы найти элементы с первым символом с. Однако этот способ не работает для поиска файлов из подкаталогов каталога. (рис. 8).

Поиск файлов, начинающихся с определенного элемента

Рис. 8: Поиск файлов, начинающихся с определенного элемента

С помощью утилиты find вывожу на экран имена файлов из каталога etc, начинающиеся с символом h. (рис. 9).

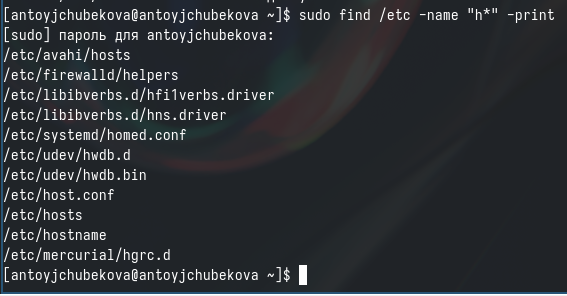


Рис. 9: Поиск файла по заданному символу

Запускаю в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log, с помощью утилиты find, амперсанта и символа перенаправления. (рис. 10).

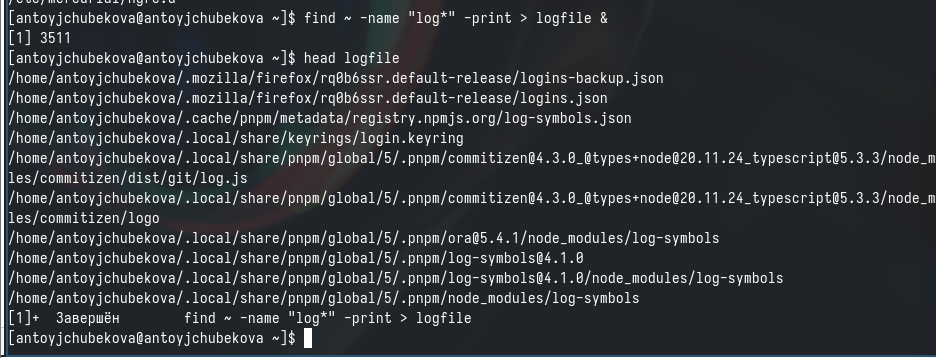


Рис. 10: Запуск процесса в фоновом режиме

Удаляю файл ~/logfile. С помощью команды ls, мы видим, что все было удалено. (рис. 11).

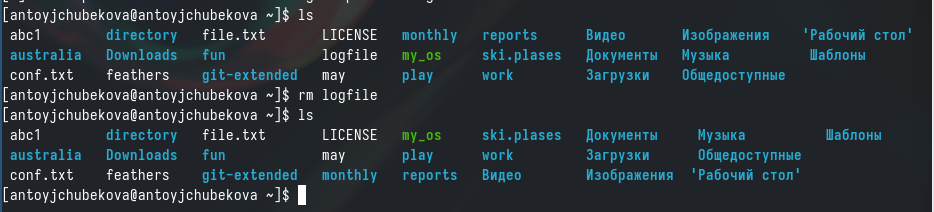


Рис. 11: Удаление файла

Запускаю из консоли в фоновом режиме редактор gedit, используя амперсант в конце. Мы видим, что редактор запустился, так что можно продолжать работу как в консоли так и в редакторе. (рис. 12 и рис. 13)

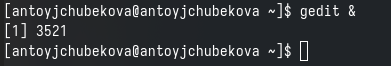


Рис. 12: Заруск процесса в фоновом режиме

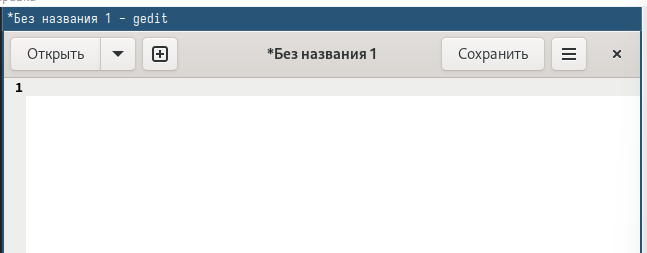


Рис. 13: Запуск процесса в фоновом режиме

Используя команду ps, также конвейер и фильтр grep определяю иденификатор процесса gedit. Мы видим, что идентификатор равен 3572. (рис. 14).

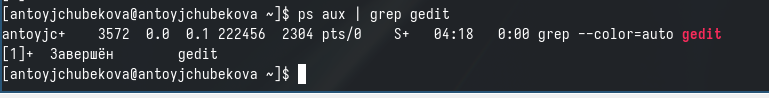


Рис. 14: Идентификатор процесса gedit

Также идентификатор процусса можно определить с помощью команды pgrep. (рис. 15).

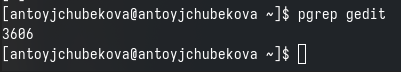


Рис. 15: Идентификатор процесса gedit

С помощью команды man просматриваю справку о команде kill, которая отвечает за прерывание процесса. (рис. 16).

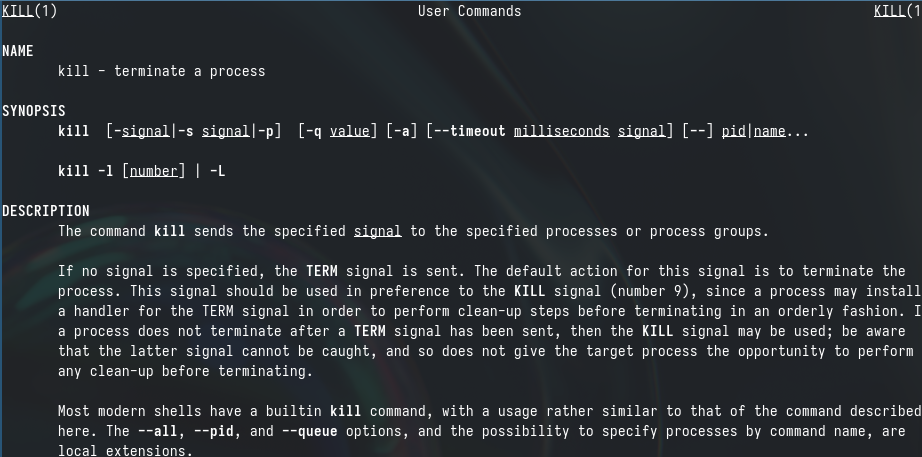


Рис. 16: Справка команды kill

Используя команду kill завершаю процесс gedit, указывая его идентификатор процесса. (рис. 17).

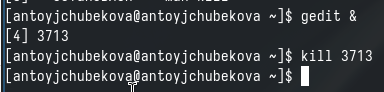


Рис. 17: Завершение процесса gedit

C помощью команды man получаю более подробную иныормацию про команды df (рис. 18) и du (рис. 19).

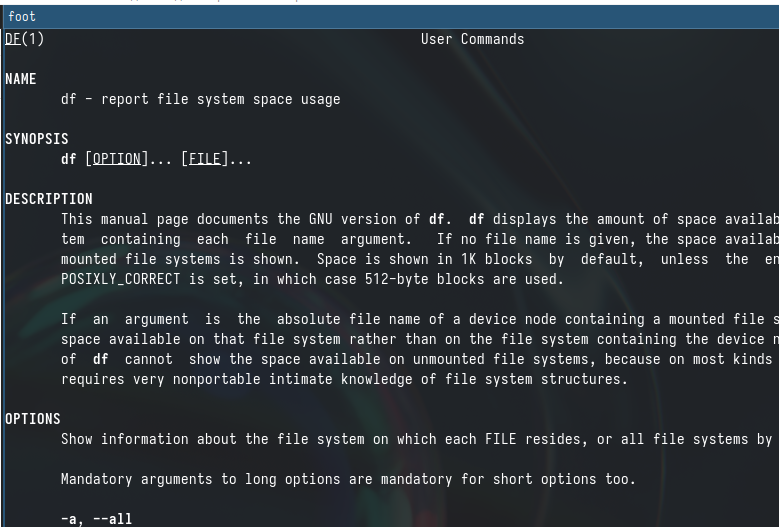


Рис. 18: Справка команды df

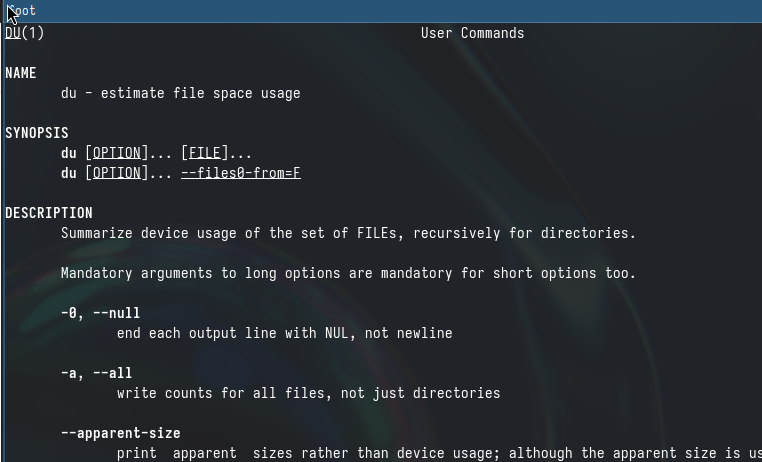


Рис. 19: Справка команды du

Выполняю команду df, которая показывает размер каждого смонтированного раздела диска. (рис. 20).

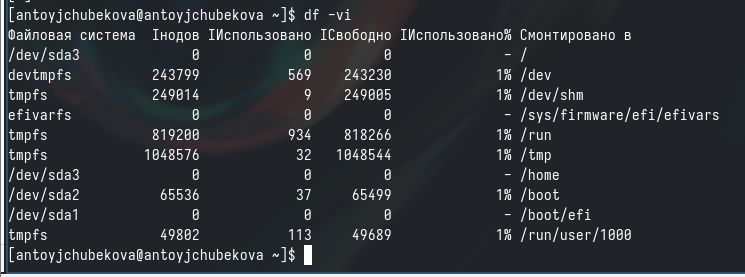


Рис. 20: Выполнение команды df

Выполняю команду du, которая показывает число килобайт, используемое каждым файлом или каталогом. (рис. 21 и рис. 22)

Выполнение команды du

Рис. 21: Выполнение команды du

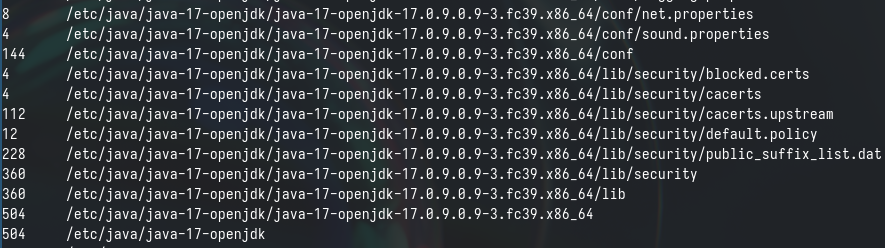


Рис. 22: Выполнение команды du

C помощью утилиты man нахожу опцию команды find, которая выведет имена всех директорий в домашнем каталоге. (рис. 23).

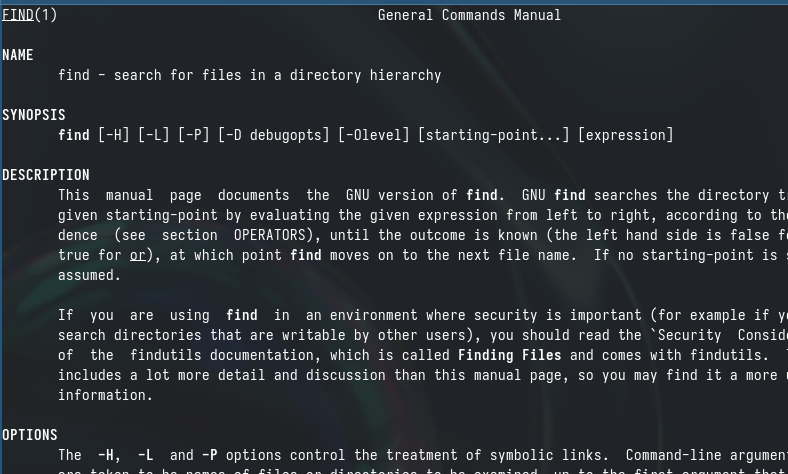


Рис. 23: Справка команды find

Вывожу на экран имена всех директорий в домашнем каталоге, используя команду find и опцию -type d. (рис. 24) и du (рис. 25).

Вывод имен директорий

Рис. 24: Вывод имен директорий

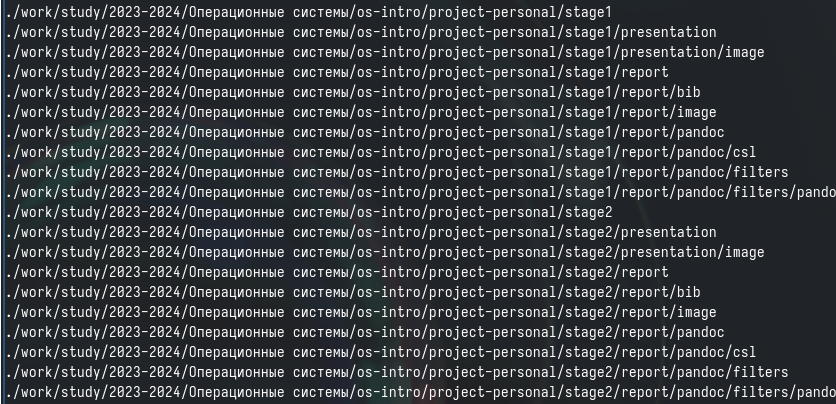


Рис. 25: Вывод имен директорий

## 4.1 Ответы на вопросы

1. Потоки ввода-вывода (I/O streams) включают стандартный ввод (stdin), стандартный вывод (stdout) и стандартный вывод ошибок (stderr).
2. Операция “>” используется для перенаправления вывода команды в файл, при этом файл будет перезаписан, если уже существует. Операция “>>” также используется для перенаправления вывода команды в файл, но в этом случае вывод будет добавлен в конец файла, не перезаписывая его содержимое.
3. Конвейер (pipeline) - это механизм в Unix-подобных операционных системах, который позволяет передавать вывод одной команды как ввод следующей команде, образуя цепочку команд.
4. Процесс - это экземпляр программы, который выполняется на компьютере. Отличие от программы заключается в том, что программа - это набор инструкций и данных, а процесс - это исполняемая копия этой программы, которая имеет свой собственный уникальный идентификатор (PID), область памяти и состояние выполнения.
5. PID (Process ID) - уникальный идентификатор процесса. GID (Group ID) - идентификатор группы, к которой принадлежит процесс.
6. Задачи (jobs) - это команды, которые выполняются в фоновом режиме в командной оболочке Unix. Команда для управления задачами - это “jobs”.
7. Утилиты “top” и “htop” предоставляют информацию о текущих процессах на компьютере, такую как использование CPU, память и другие ресурсы. “htop” предоставляет более детальную информацию и имеет более удобный интерфейс по сравнению с “top”.
8. Команда поиска файлов - “find”. Она позволяет найти файлы и каталоги по различным критериям, таким как имя, тип, размер и т. д. Примеры использования:
   * “find /home/user -name”\*.txt”” - поиск всех файлов с расширением .txt в домашнем каталоге пользователя.
   * “find /var/log” - поиск всех файлов в каталоге /var/log.
9. Да, можно. Для этого можно использовать команду “grep”, чтобы выполнить поиск по содержимому файлов. Например, чтобы найти файлы, содержащие определенную строку, можно использовать следующую команду: “grep -r”строка” /путь/к/каталогу”.
10. Объем свободной памяти на жестком диске можно определить, используя команду “df”. Например: “df -h” покажет размеры файловых систем, включая информацию о доступном и используемом месте.
11. Объем вашего домашнего каталога можно определить, используя команду “du” (от “disk usage”). Например: “du -sh ~” покажет общий объем использованного места в вашем домашнем каталоге.
12. Для удаления зависшего процесса можно использовать команду “kill”. Сначала нужно определить PID зависшего процесса, а затем отправить сигнал завершения процесса. Например: “kill PID”.

# 5 Выводы

В ходе выполнения лабораторной работы №8 я ознакомилась с инструментами поиска файлов и фильтрации текстовых данных. Также приобрела практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.