# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

**Факультет Физико-Математических Наук Кафедра прикладной информатики и теории вероятностей**

# ОТЧЁТ

**ПО ЛАБОРАТОРНОЙ РАБОТЕ № 7**

*дисциплина: Операционные системы*

Студент: Манаева Варвара Евгеньевна Группа: НФИбд-01-20

Преподаватель: Кулябов Дмитрий Сергеевич

# МОСКВА

2021 г.

# Техническое оснащение:

Персональный компьютер с операционной системой Windows 7; Планшет для записи видеосопровождения и голосовых комментариев;

Виртуальная коробка VirtualBox, виртуальная машина с установленной на ней операционной системой CentOS; Microsoft Teams, использующийся для записи скринкаста лабораторной работы;

Приложение MarkPad 2 для редактирования файлов формата *md*; *pandoc* для конвертации файлов отчётов и презентаций.

**Объект и предмет исследования:** Инструмент поиска файлов, файловая система *Linux*.

**Цель:** ознакомиться с инструментами поиска файлов и фильтрации текстовых данных, приобрести практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

# Задачи:

1. Научиться искать файлы различными методами по различным признакам;
2. Научиться управлять процессами и заданиями;
3. Научиться проверять использование диска и обслуживание файловых систем.

**Теоретические вводные данные: Перенаправление ввода-вывода** 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 вывода 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 *>*, *>>*, *<*, *<<*. Рассмотрим пример.

# Перенаправление stdout (вывода) в файл. # Если файл отсутствовал, то он создаётся, # иначе -- перезаписывается.

# Создаёт файл, содержащий список дерева каталогов.

*ls -lR > dir-tree.list 1>filename*

# Перенаправление вывода (stdout) в файл "filename".

*1>>filename*

# Перенаправление вывода (stdout) в файл "filename", # файл открывается в режиме добавления.

*2>filename*

# Перенаправление stderr в файл "filename".

*2>>filename*

# Перенаправление stderr в файл "filename", # файл открывается в режиме добавления. *&>filename*

# Перенаправление stdout и stderr в файл "filename".

**Конвейер.** Конвейер (*pipe*) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей. Синтаксис следующий: *команда 1 | команда 2*

# означает, что вывод команды 1 передастся на ввод команде 2

Конвейеры можно группировать в цепочки и выводить с помощью перенаправления в файл, например:

*ls -la |sort > sortilglist*

*вывод команды ls -la передаётся команде сортировки sort\verb, которая пишет результат в файл sortinglist\verb*.

Чаще всего скрипты на *Bash* используются в качестве автоматизации каких-то рутинных операций в консоли, отсюда иногда возникает необходимость в обработке *stdout* одной команды и передача на *stdin* другой команде, при этом результат выполнения одной команды

должен быть неким образом обработан.

**Поиск файла.** Команда *find* используется для поиска и отображения имён файлов, соответствующих заданной строке символов. Формат команды:

*find путь [-опции]*

Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск.

Примеры:

1. Вывести на экран имена файлов из вашего домашнего каталога и его подкаталогов, начинающихся на *f*: *find ~ -name "f\*" -print* , где *~* — обозначение вашего домашнего каталога,

*-name* — после этой опции указывается имя файла, который нужно найти,

*"f\*"* — строка символов, определяющая имя файла,

-print — опция, задающая вывод результатов поиска на экран.

1. Вывести на экран имена файлов в каталоге */etc*, начинающихся с символа *p*:

*find /etc -name "p\*" -print*

1. Найти в вашем домашнем каталоге файлы, имена которых заканчиваются символом и удалить их:

*find ~ -name "\*~" -exec rm "{}" \;*

где опция *-exec rm "{}" \;* задаёт применение команды *rm* ко всем файлам, имена которых соответствуют указанной после опции *-name*

строке символов.

Для просмотра полного списка опций команды *find* воспользуйтесь командой *man*.

**Фильтрация текста.** Найти в текстовом файле указанную строку символов позволяет команда *grep*. Формат команды:

*grep строка имя\_файла*

Кроме того, команда *grep* способна обрабатывать стандартный вывод других команд (любой текст). Для этого следует использовать конвейер, связав вывод команды с вводом *grep*.

Примеры:

1. Показать строки во всех файлах в вашем домашнем каталоге с именами, начинающимися на *f*, в которых есть слово *begin*: *grep begin f\**
2. Найти в текущем каталоге все файлы, в имени которых есть буквосочетание «лаб»:

*ls -l | grep лаб*

**Проверка использования диска.** Команда *df* показывает размер каждого смонтированного раздела диска. Формат команды:

*df [-опции] [файловая\_система]*

Пример:

*df -vi*

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

*du [-опции] [имя\_файла...]*

Пример.

*du -a ~/*

На *afs* можно посмотреть использованное пространство командой

*fs quota*

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

*gedit &*

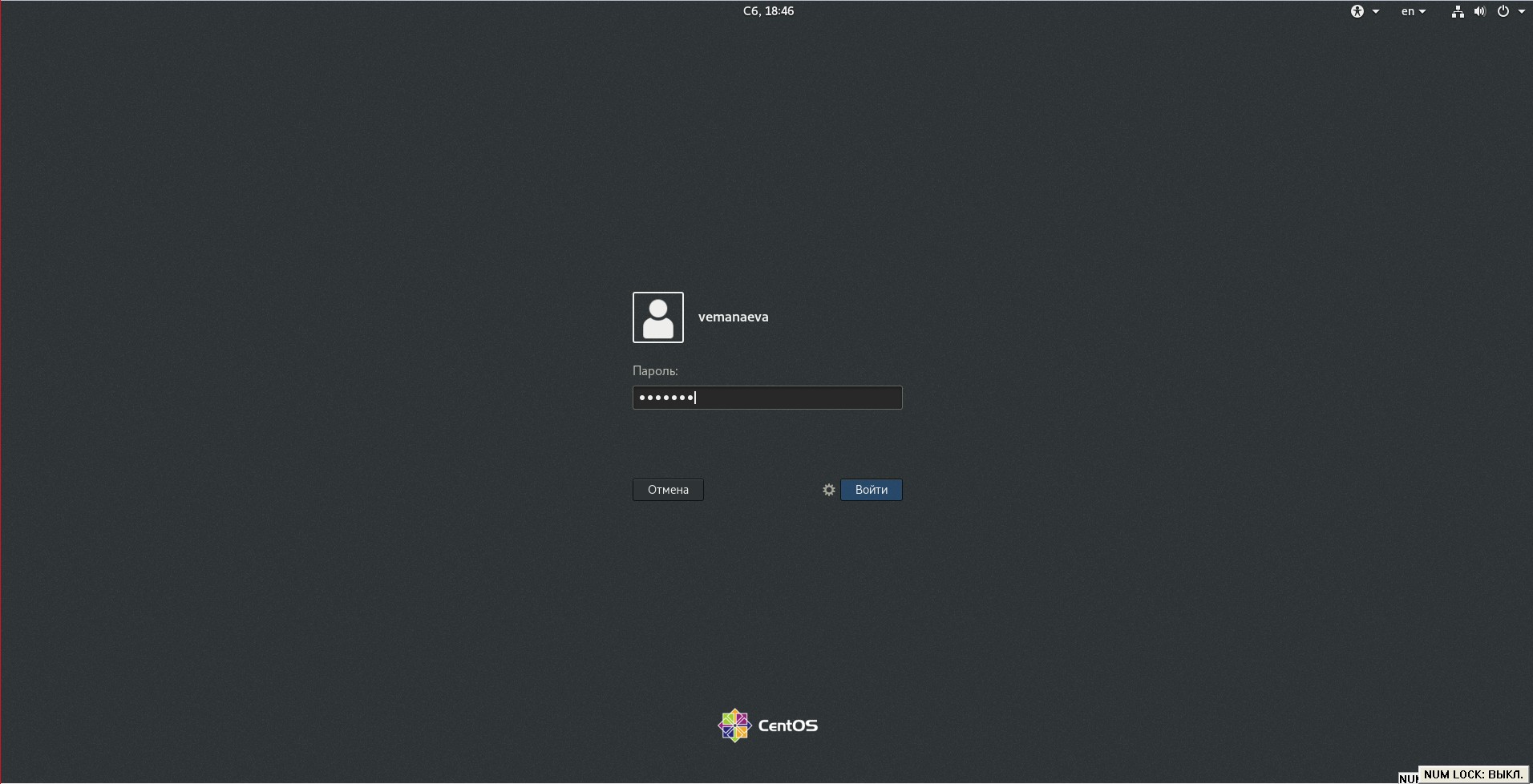
Будет запущен текстовой редактор *gedit* в фоновом режиме. Консоль при этом не будет заблокирована.

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 %номер задачи*

# Этапы работы:

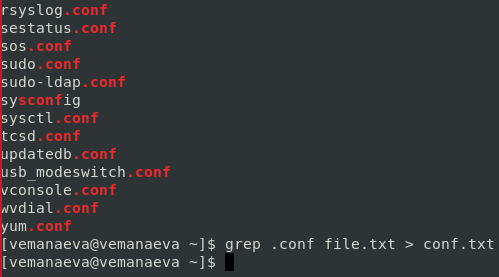
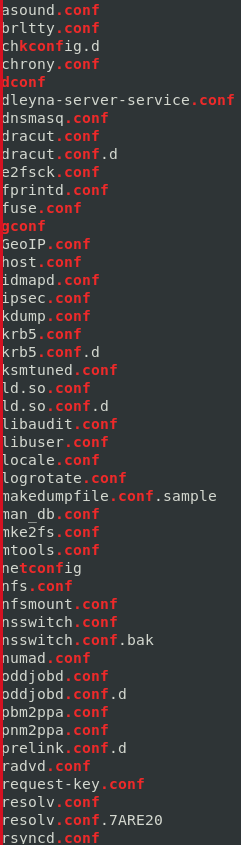
1. Осуществила вход в систему, используя соответствующее имя пользователя.



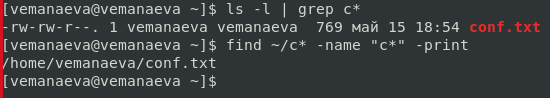
1. Записала в файл *file.txt* названия файлов, содержащихся в каталоге */etc*. Дописала в этот же файл названия файлов, содержащихся в моём домашнем каталоге.



1. Вывела имена всех файлов из *file.txt*, имеющих расширение *.conf*, после чего записала их в новый текстовой файл *conf.txt*.

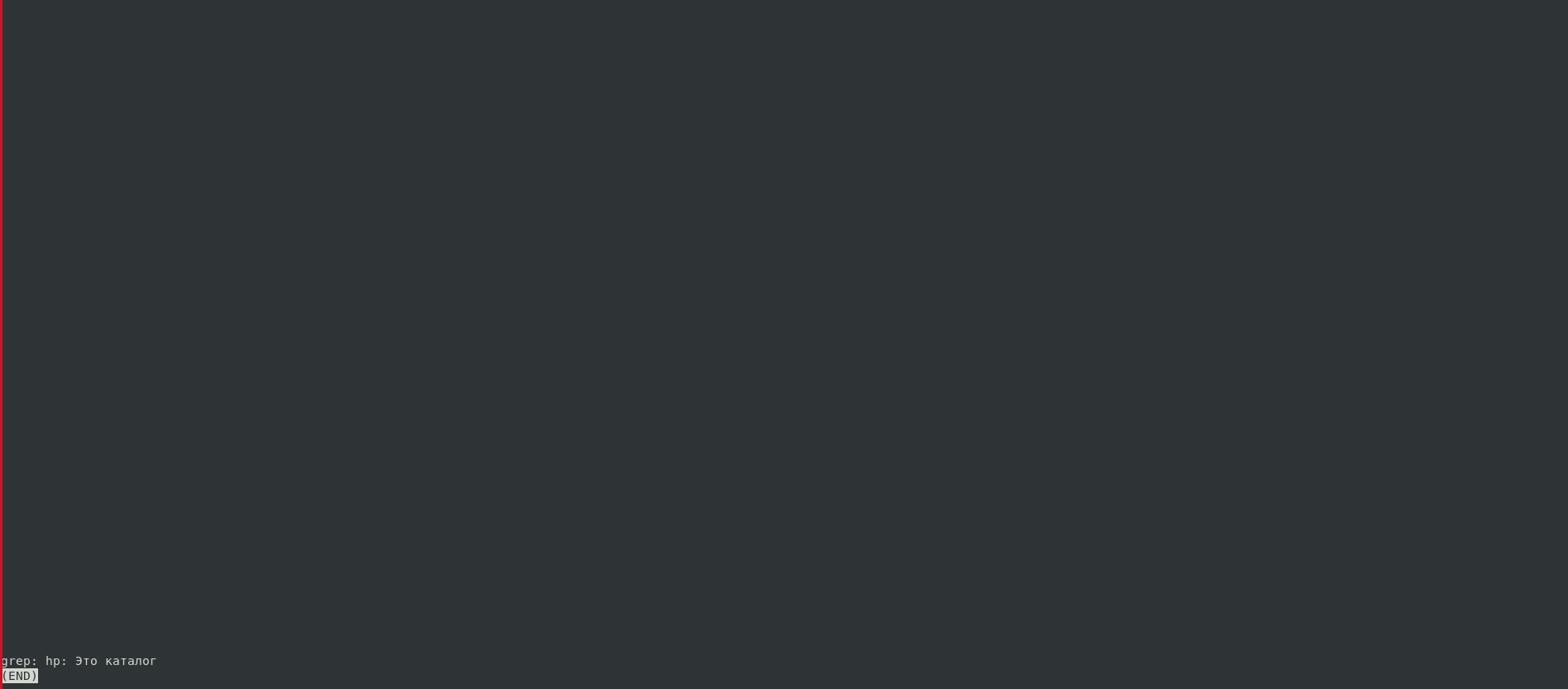


1. Определила, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа "c". Несколько вариантов, как это сделать:



1. Вывела на экран (постранично) имена файлов из каталога */etc*, начинающиеся с символа "h".





1. Запустила в фоновом режиме процесс, который будет записывать в файл *~/logfile* файлы, имена которых начинаются с "log".

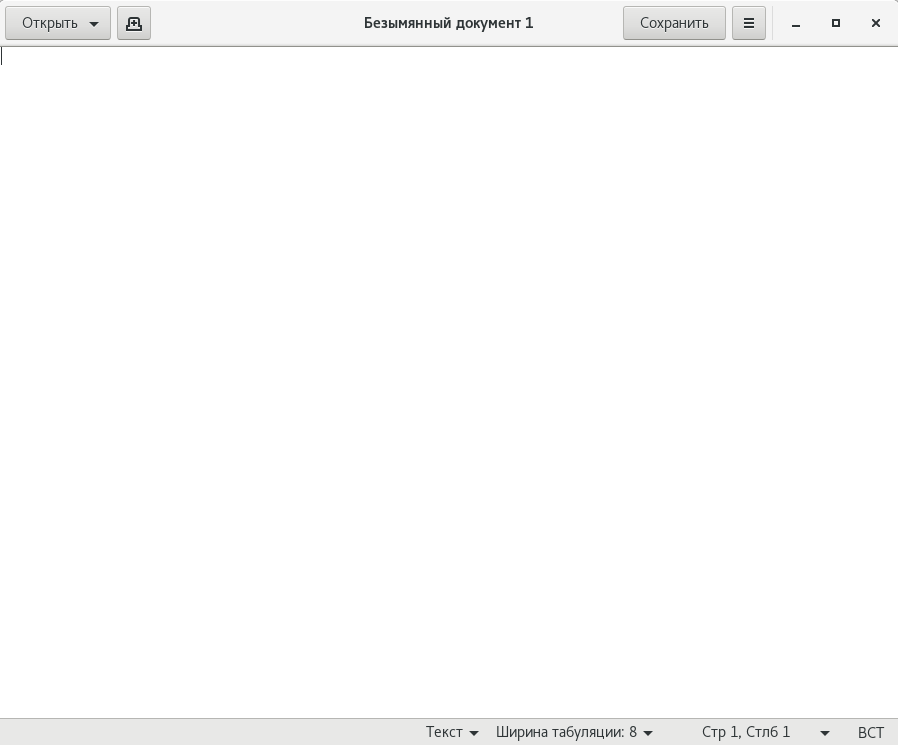


1. Удалила файл *~/logfile*.

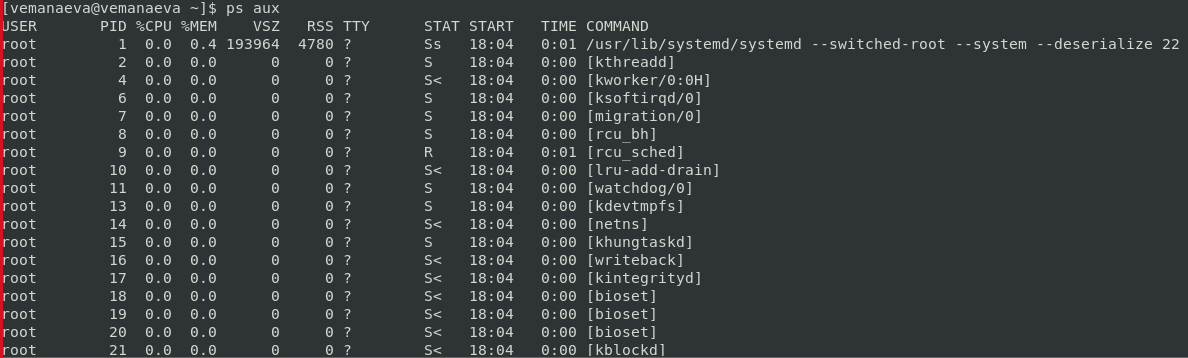


1. Запустила из консоли в фоновом режиме редактор *gedit*.

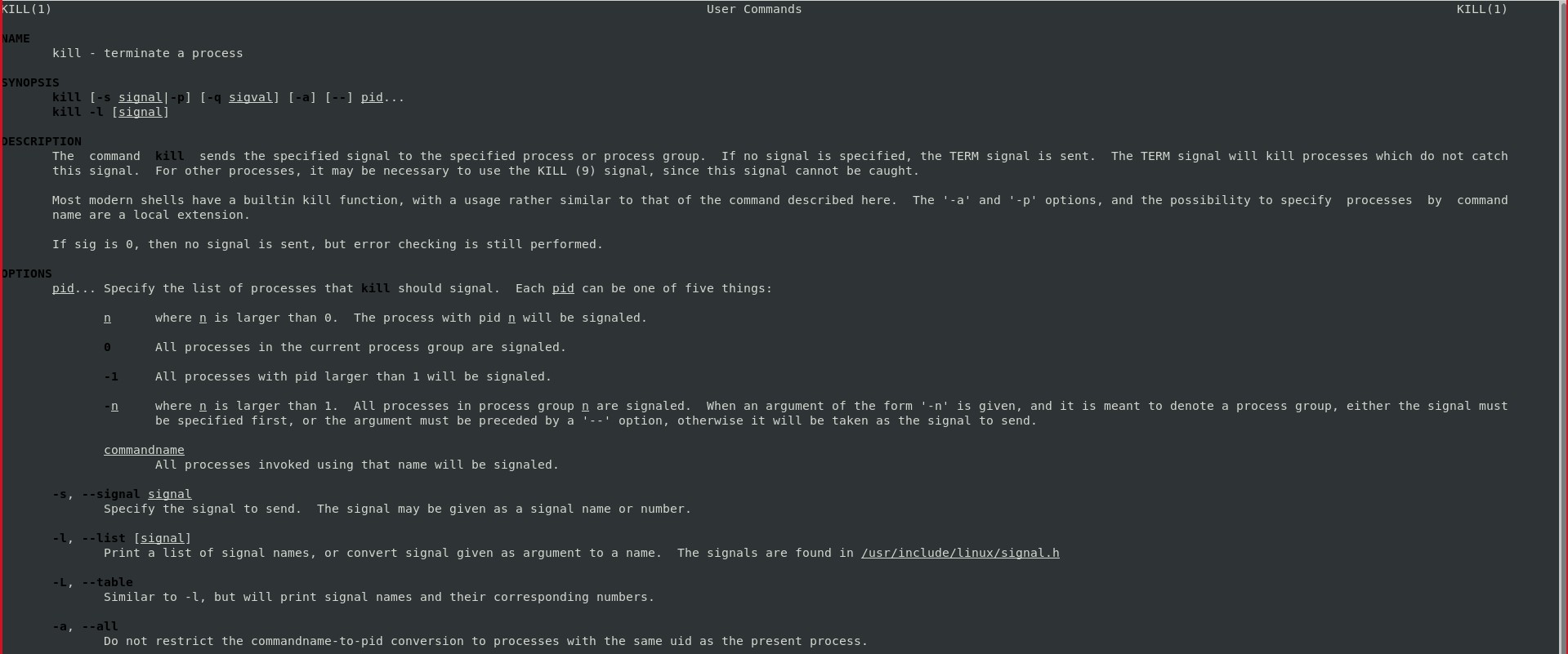




1. Определила идентификатор процесса *gedit*, используя команду *ps*, конвейер и фильтр *grep*. Более простым способом определить этот идентификатор получилось: просто введя команду ps aux и глазками отыскав её в списке. Насколько этот способ проще судить не берусь.

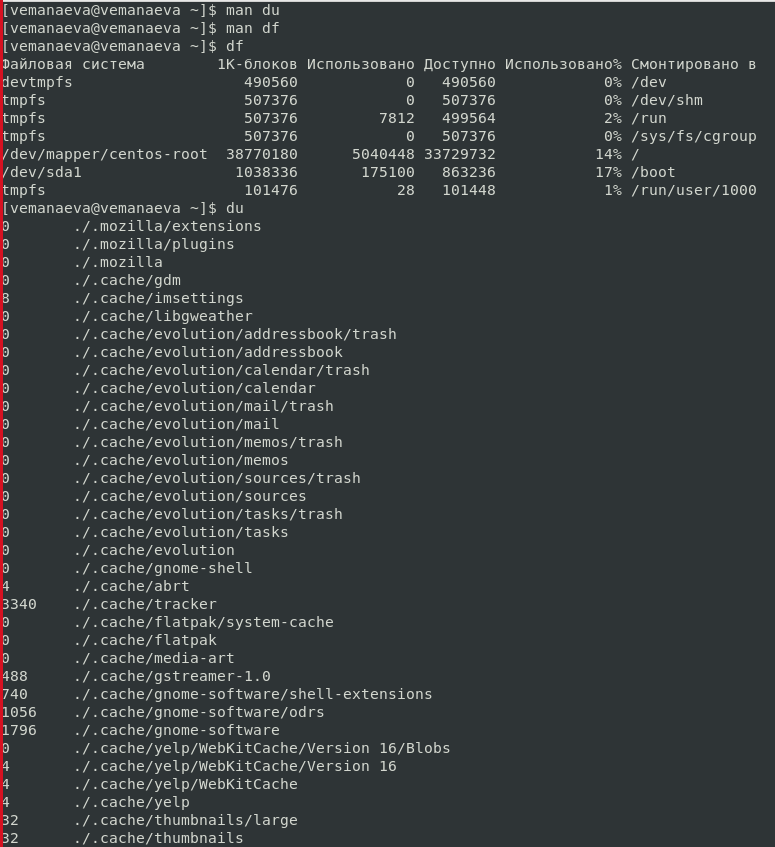
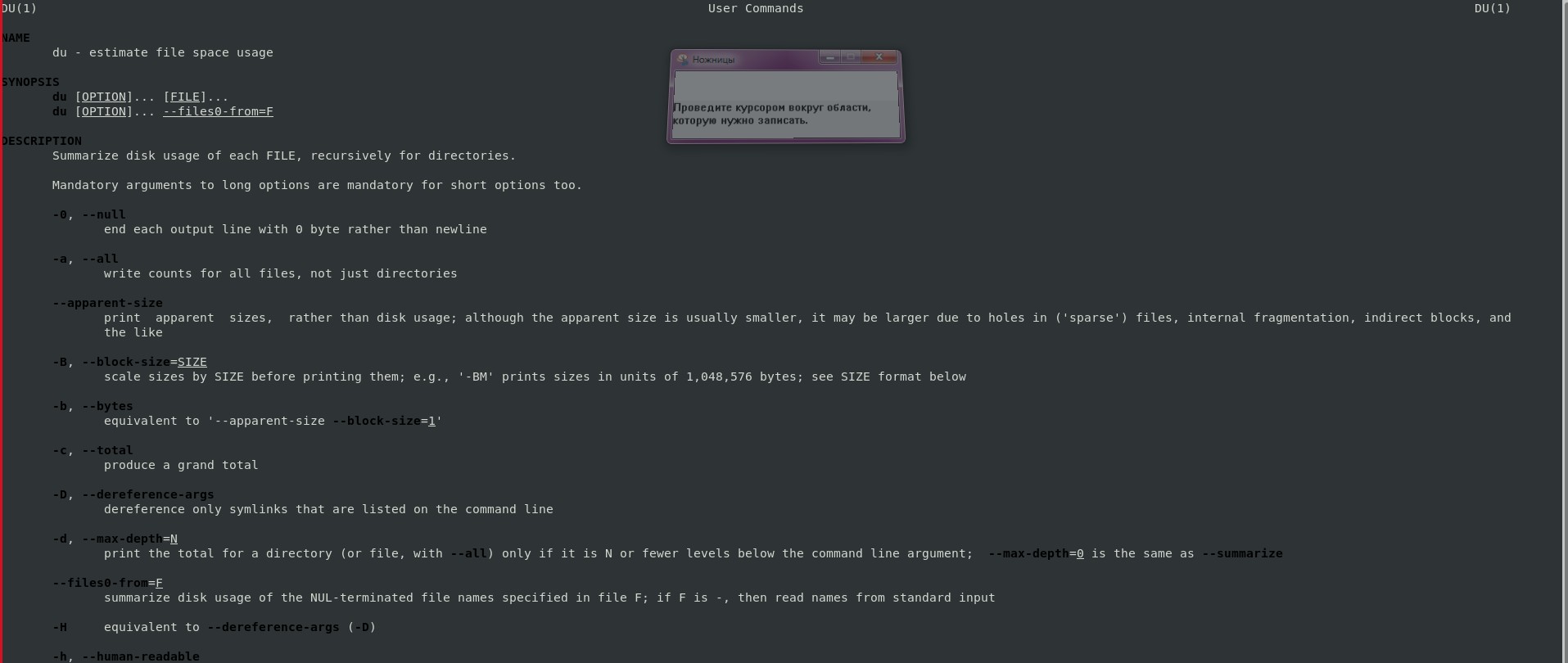


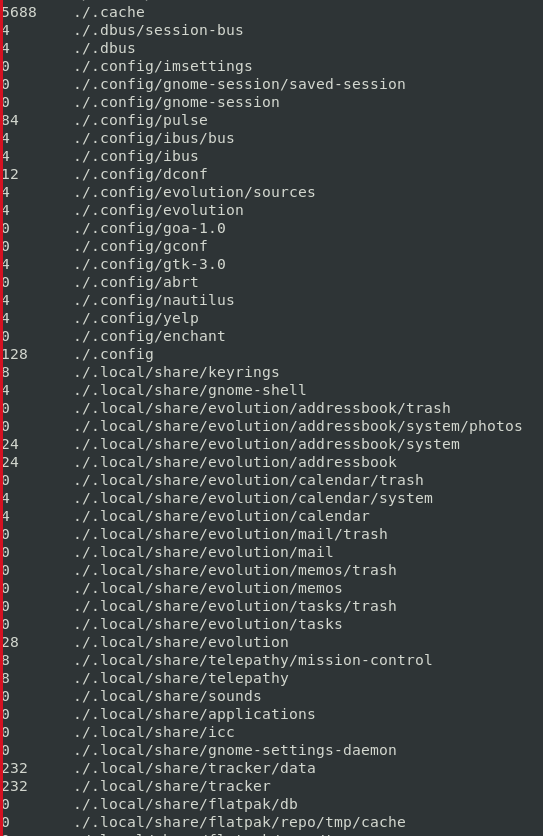
1. Прочитала справку (*man*) команды *kill*, после чего использовала её для завершения процесса *gedit*.

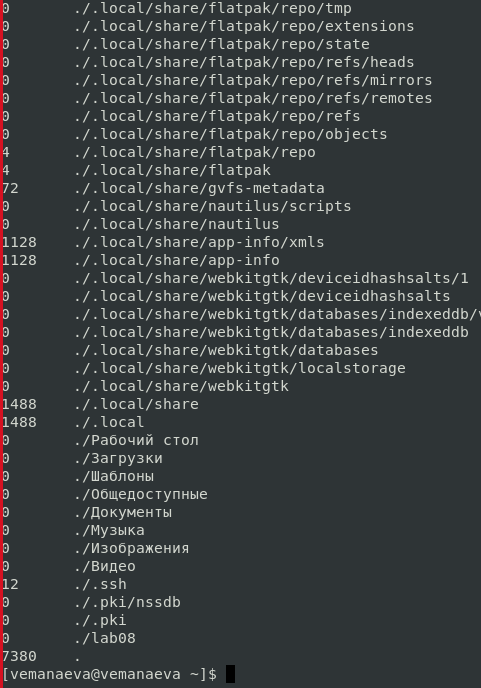


1. Выполнила команды *df* и *du*, предварительно получив более подробную информацию об этих командах, с помощью команды *man*.

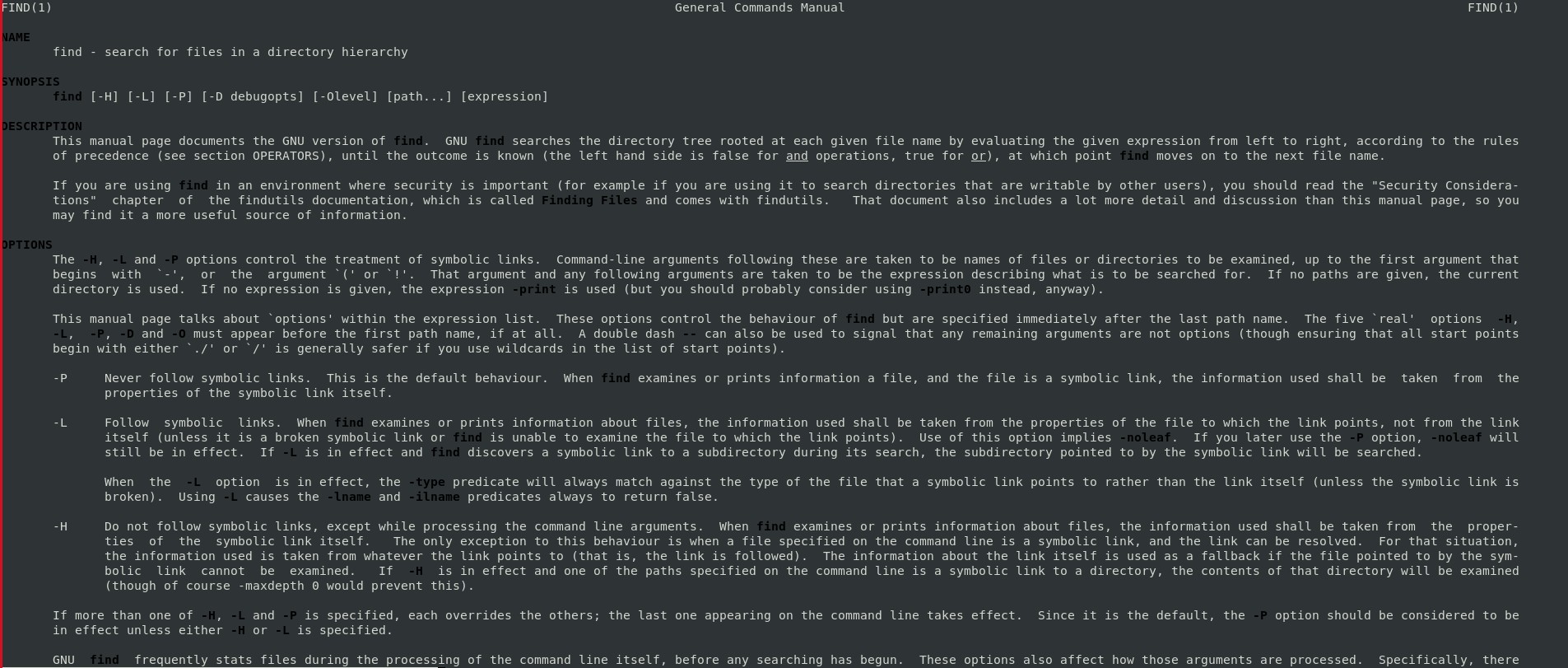


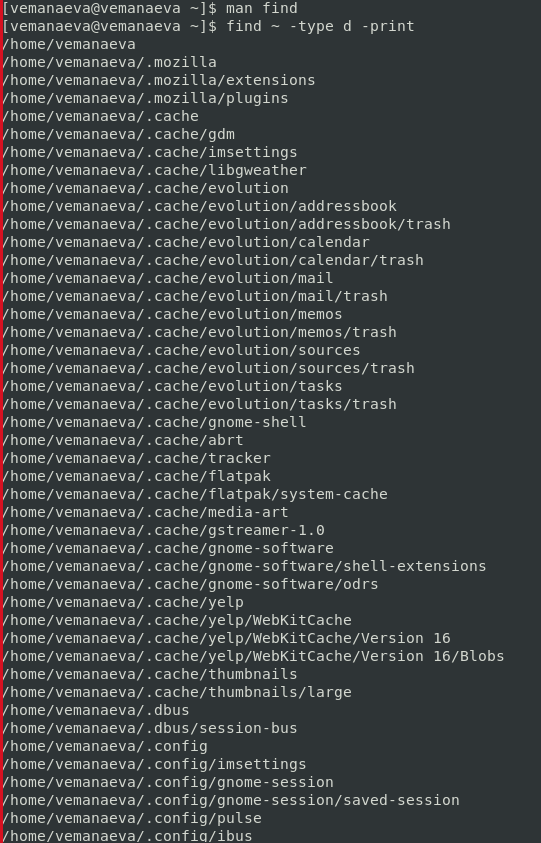


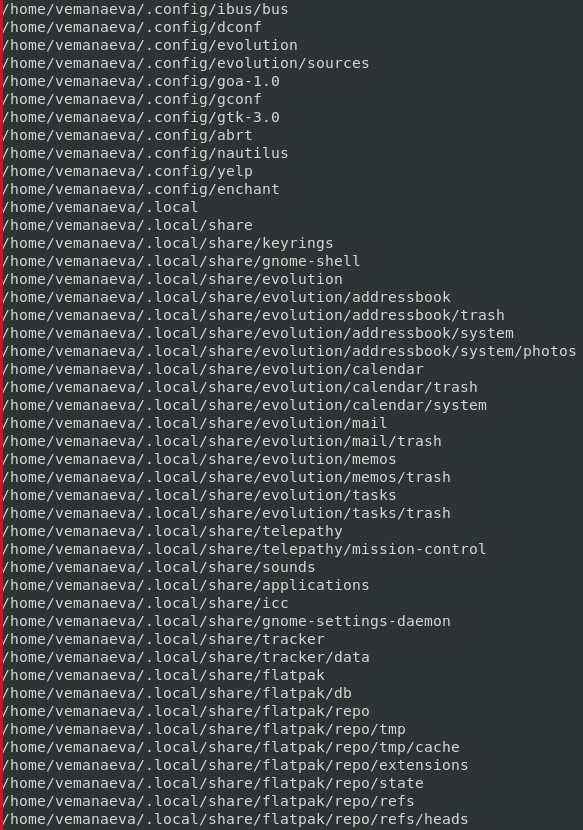


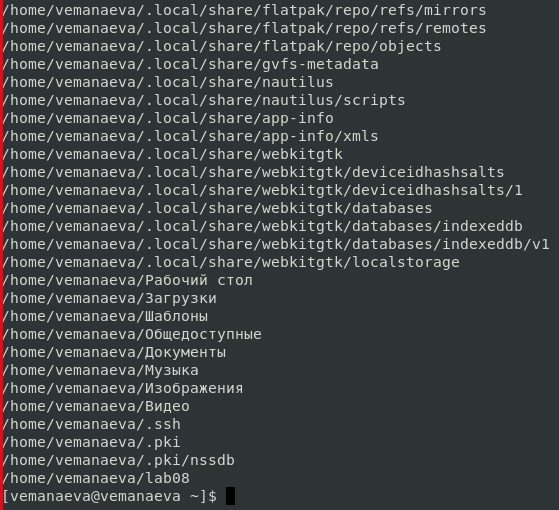


1. Воспользовавшись справкой команды *find*, вывела имена всех директорий, имеющихся в моём домашнем каталоге.









**Выводы:** я ознакомилась с инструментами поиска файлов и фильтрацией текстовых данных, приобрела практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

# Контрольные вопросы:

1. *Какие потоки ввода вывода вы знаете?*

Ответ: Существует несколько потоков ввода вывода:

stdin — стандартный поток ввода (по умолчанию: клавиатура), файловый дескриптор 0;

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

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

1. *Объясните разницу между операцией > и >>.*

Ответ: *>* - перенаправление stdout (вывода) в файл. Если файл отсутствовал, то он создаётся, иначе - перезаписывается.

*>>* - перенаправление stdout (вывода) в файл. Если файл отсутствовал, то он создаётся, иначе - добавляется.

1. *Что такое конвейер?*

Ответ: Конвейер (*pipe*) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.

1. *Что такое процесс? Чем это понятие отличается от программы?*

Ответ: Процесс - это совокупность программного кода и данных, загруженных в память ЭВМ. 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. Процессом называют выполняющуюся программу и все её элементы: адресное пространство, глобальные переменные, регистры, стек, открытые файлы и так далее.

1. *Что такое PID и GID?*

Ответ: *PID* — уникальный номер (идентификатор) процесса в многозадачной ОС.

*GID* – идентификатор группы.

1. *Что такое задачи и какая команда позволяет ими управлять?*

Ответ: 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 %номер задачи*.

1. *Найдите информацию об утилитах top и htop. Каковы их функции?*

Ответ: Команда *top* в *Linux* системах позволяет вывести в виде таблицы перечень запущенных процессов и оценить, какой объем ресурсов они потребляют, т.е., какую нагрузку создают на сервер и дисковую подсистему.

Команда *htop* — продвинутый монитор процессов, показывает динамический список системных процессов, список обычно выравнивается по использованию ЦПУ. В отличие от *top*, *htop* показывает все процессы в системе. Также показывает время непрерывной работы, использование процессоров и памяти. *htop* часто применяется в тех случаях, когда информации, даваемой утилитой *top* недостаточно, например, при поиске утечек памяти в процессах.

1. *Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды.*

Ответ: Команда *find* используется для поиска и отображения имён файлов, соответствующих заданной строке символов. Формат команды:

*find путь [-опции]*

Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск. Примеры:

вывести на экран имена файлов из вашего домашнего каталога и его подкаталогов, начинающихся на f: *find ~ -name "f\*" -print*

вывести на экран имена файлов в каталоге */etc*, начинающихся с символа p: *find /etc -name "p\*" -print*

найти в вашем домашнем каталоге файлы, имена которых заканчиваются символом и удалить их: *find ~ -name "\*~" -exec rm "{}" \;*

1. *Можно ли по контексту (содержанию) найти файл? Если да, то как?*

Ответ: Найти файл по контексту (содержанию) позволяет команда *grep*. Формат команды: *grep строка имя\_файла*

Примеры:

показать строки во всех файлах в вашем домашнем каталоге с именами, начинающимися на f, в которых есть слово begin:

*grep begin f\**

найти в текущем каталоге все файлы, в имени которых есть буквосочетание «лаб»:

*ls -l | grep лаб*

1. *Как определить объём свободной памяти на жёстком диске?*

Ответ: Определить объём свободной памяти на жёстком диске позволяет команда *df*.

1. *Как определить объём вашего домашнего каталога?*

Ответ: Определить объём домашнего каталога позволяет команда *df /home/*.

1. *Как удалить зависший процесс?*

Ответ: Удалить зависший процесс можно командой *kill %номер задачи*.