# ™Лабораторная работа 9. Специальные команды файловой системы

**Аннотация:** Цель работы: изучить специальные команды файловых систем, в том числе команды поиска, сжатия и архивирования файлов, а также команды монтирования в Linuxна примере операционной системы Ubuntu.

**Задание 9-1.** Изучите назначение и синтаксис команды *df*.

Ход выполнения:

1. Ознакомьтесь с описанием команды *df*.

Команда *df*выдает информацию о доступном и занятом дисковом пространстве на файловых системах.

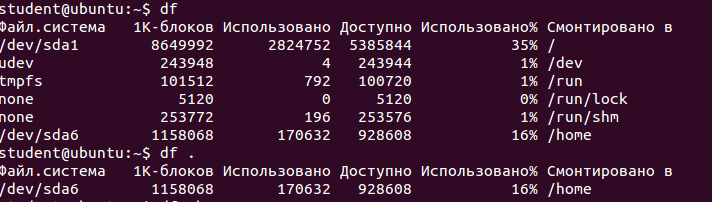
Вызов команды без аргумента выдает отчет для всех файловых систем, которые смонтированы в данный момент. По умолчанию все размеры выдаются в блоках по 1024 байта.

Для вывода на экран итоговых данных текущей файловой системы используется команда

*df .*

2. Выведите на экран информацию о свободном пространстве жесткого диска.

3.Выведите на экран информацию о свободном пространстве текущей файловой системы.



**Задание 9-2.** Изучите назначение и синтаксис команды *du*.

Ход выполнения:

1.Ознакомьтесь с описанием команды *du*.

Команда *du* предоставляет информацию о размере занимаемого каталогом или файлом дискового пространства. Общий вид команды:

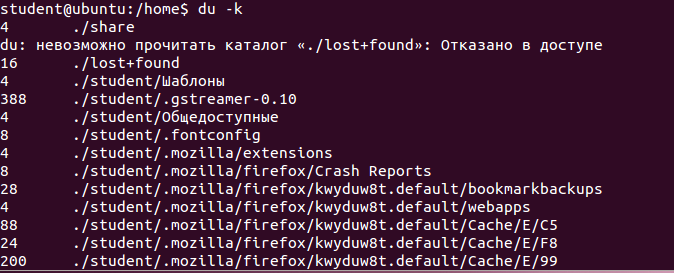
*du [-ask] <имена файлов>*

*du -a* – разделит суммарное значение и покажет размер каждого каталога и подкаталога.

*du -s* – позволит показать только итоговое значение.

*du -k* – приводит к выводу размеров всех файлов в килобайтах.

2. Отобразите размеры всех файлов рабочей папки на экране.



**Задание 9-3.** Изучите назначение и синтаксис команды *dd*.

Ход выполнения:

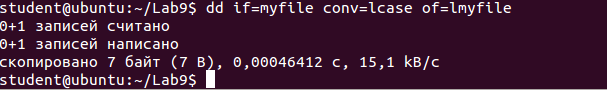
1.Ознакомьтесь с кратким описанием команды *dd*.

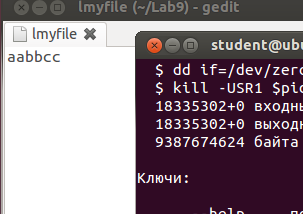
Команда *dd* используется для преобразования файлов из одного формата в другой. Синтаксис:

*dd if = <файл ввода> [conv = тип преобразования] of = <файл вывода> [obs = размер входного блока]*

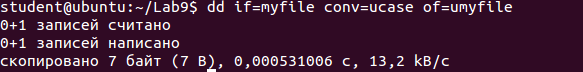
2. Создайте файл *myfile*, содержащий строку символов «*aAbBcC*».

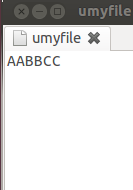
3. Командой *dd* создайте файл *lmyfile* путем следующего преобразования формата файла: все символы файла *myfile* преобразуйте в нижний регистр (тип преобразования *lcase*).





4. Командой *dd* создайте файл *umyfile* путем следующего преобразования формата файла: все символы файла *myfile* преобразуйте в верхний регистр (тип преобразования *ucase*).





**Задание 9-4.** Изучите команды сжатия и архивирования файлов в Linux.

Ход выполнения:

1.Ознакомьтесьс основными командами сжатия и архивирования в Linux.

*compress*– команда сжатия файла. Файл заменяется файлом с расширением *.Z*. Если команда с параметром *-v*, то будет выводиться информация о сокращении размера сжатого файла в процентах и имя нового файла.

*uncompress* – восстановление файла с расширением *.Z*.

*gzip* – команда сжатия файла. Файл заменяется файлом с расширением *.gz*.

Если команда с параметром *-9*, то сжатие будет улучшено за счет скорости. Параметр *-v* обеспечивает подробный вывод.

*gunzip*– восстановление файлов в их первоначальном виде.

*zip* – архивация файлов, сжатые файлы получают расширение *.zip*. Эта команда может создавать самораспаковывающиеся файлы, добавлять комментарии в файлы ZIP, удалять файлы из архива и защищать архив с помощью пароля.

*unzip* – восстановление файла с расширением *.zip*.

*tar*– создание архива множества файлов и каталогов в одном файле с расширением *.tar* и извлечение файлов и каталогов из такого архивного файла.

Синтаксис:

*tar [c] [x] [v] [z] [f имя создаваемого файла] <имена файлов и каталогов>:*

параметр *c*– создание tar-файла;

параметр *v* – позволяет увидеть, какие будут архивироваться файлы;

параметр *z* – создаваемый файл тоже сжимается, ему должно быть присвоено расширение *.gz* (например, tar czf backup.tar.gz important\_dir, расширение .tar.gz часто сокращают до .tgz);

параметр *x* – извлечение архивного файла (например, tar xf backup.tar или tar xzp backup.tgz, если tar-файл сжатый).

2. Создайте папку с названием Lab\_4.

В папку скопируйте файлы:

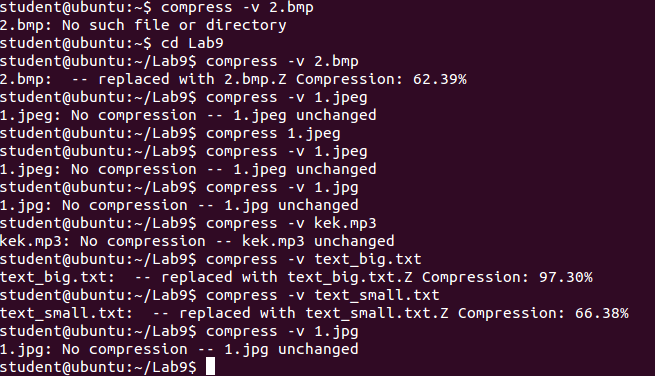
* графические файлы bmp и jpg; (найти в операционной системе или создать с помощью графического редактора изображение и сохранить его в разных форматах)
* текстовый файл объемом более 700 Кб;
* текстовый файл маленького размера;
* любые другие файлы.

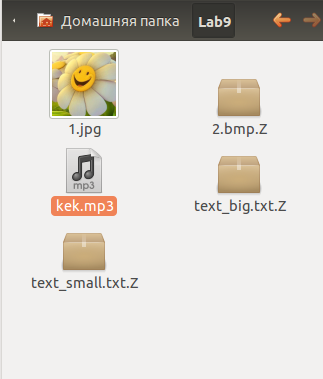
Для составления отчета используйте таблицу вида:

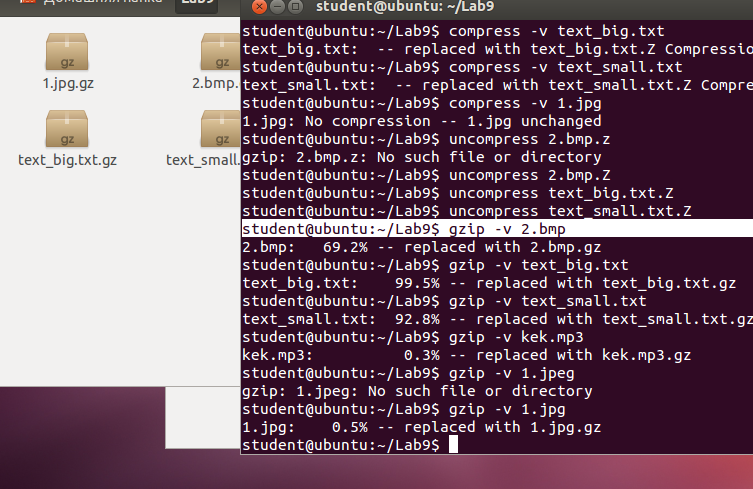
3. Заархивируйте файлы с помощью команды *compress*. Выведите информацию о размере полученного файла. Восстановите файлы. Отразите результат в таблице 9.1. Обратите внимание на размер небольшого текстового файла до сжатия и после.

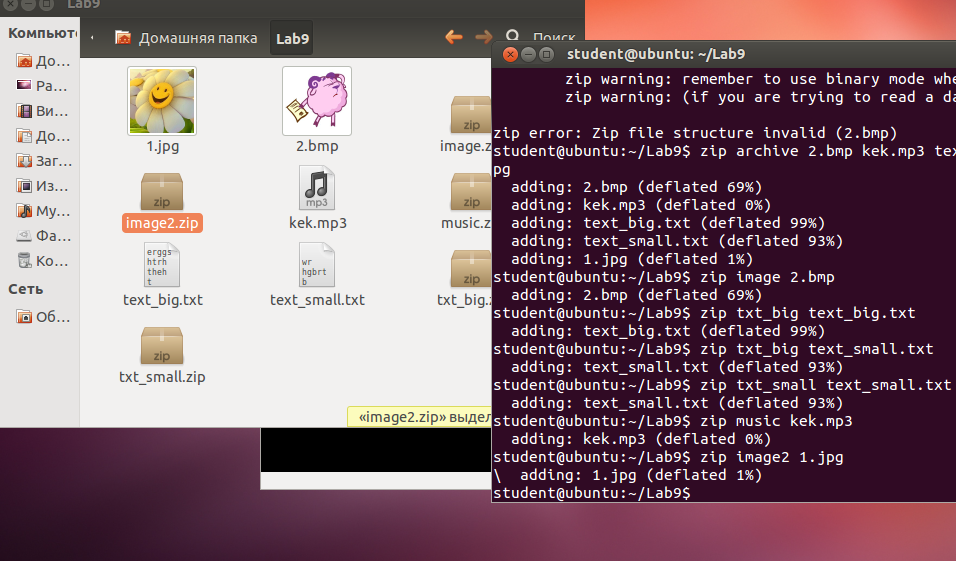
Таблица 9.1. Сравнение результатов сжатия файлов разными командами.

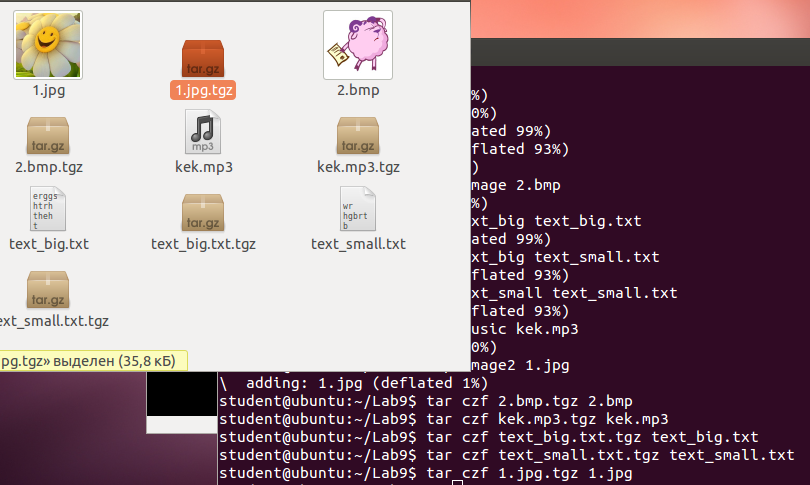
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Файл | начальный размер | compress | gzip | Zip | tar |
| 2.bmp | *196,7 кБ* | *74,0 кБ* | *60,6 кБ* | *60,7 кБ* | *60,7 кБ* |
| text\_big.txt (размер >700кб) | *701,4 кБ* | *20,2 кБ* | *3,9 кБ* | *4,0 кБ* | *4,0 кБ* |
| text\_small.txt (размер >700кб) | *961 байт* | *323 байта* | *102 байта* | *247 байт* | *190 байт* |
| kek.mp3 | *9,8 Мб* | *-* | *9,8 МБ* | *9,8 МБ* | *9,8 МБ* |
| 1.jpg | *35,8 кБ* | *-* | *35,6 кБ* | *35,8 кБ* | *35,8 кБ* |





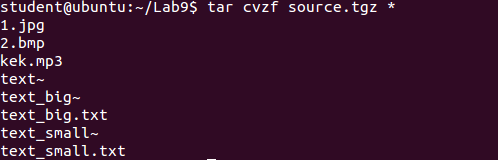


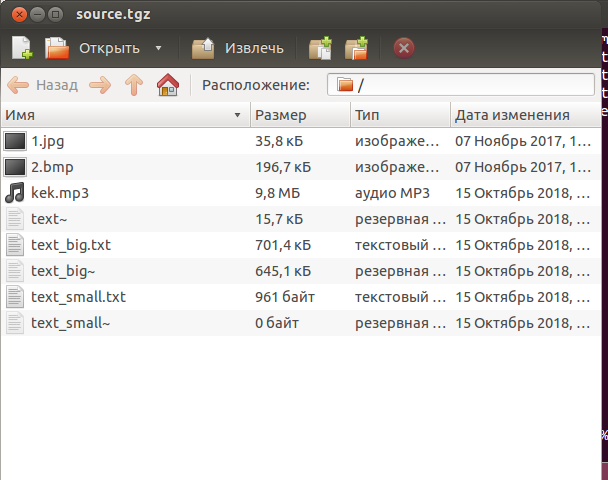


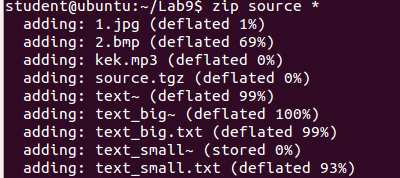


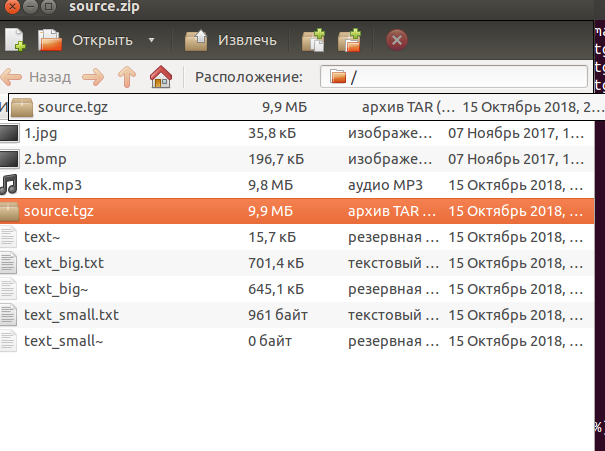
4. Выполните архивирование файлов разного типа с помощью команды *gzip*, *zip*, *tar*. Обратите внимание на разницу синтаксиса команд.

5. Создайте архив файлов с именем source, который будет содержать все файлы папки.









**Задание 9-5.** Создайте файловую систему в файле, расположенном в существующей файловой системе, чтобы продемонстрировать возможности уровня файловой системы Linux (и монтирования).

Ход выполнения:

1. Для выполнения данного задания необходимо иметь права пользователя. Для этого нужно либо зарегистрироваться в системе под учетной записью *root*:

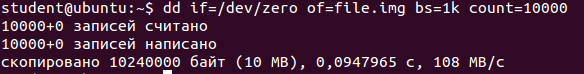
*suroot*

а затем ввести пароль пользователя *root*, который был задан при установке операционной системы.

Либо перед командами, на которые недостаточно прав обычного пользователя вводить команду *sudo*, а затем уже пароль пользователя.

Выполняя команды с правами администратора, будьте очень внимательны, так как ошибки или необдуманные действия могут привести к неожиданным результатам и даже к выходу из строя операционной системы.

2. Создайте файл заданного размера с помощью команды *dd* путем копирования файла из источника */dev/zero*– другими словами, инициализируя файл нулями, как показано на рисунке 9.1.



*Рис. 9.1*. Создание инициализированного файла

В результате выполнения команды создан файл file.img размером 10 МБ.

3. Свяжем с файлом блочное устройство-заглушку (loop) с помощью команды *losetup* (чтобы он выглядел как блочное устройство, а не как обычный файл файловой системы):

*losetup /dev/loop0 file.img*

Если вы попытаетесь выполнить данную команду с правами обычного пользователя, то увидите сообщение о том, что вам отказано в доступе. Для решения этой проблемы используйте один из вариантов, перечисленных в п.1 данного задания. Например, чтобы выполнить команду с правами администратора, не регистрируясь при этом под пользователем *root*, введите:

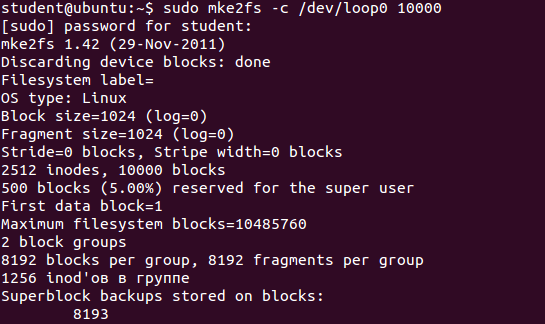
*sudo losetup /dev/loop0 file.img*

На экране появится приглашение ввести пароль пользователя (рис. 9.2):



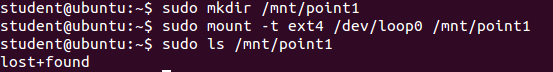
*Рис. 9.2.*Установка связи между файлом и блочным устройством-заглушкой

4. Создайте на блочном устройстве файловую систему определенного размера с помощью команды *mke2fs* (рис. 9.3).



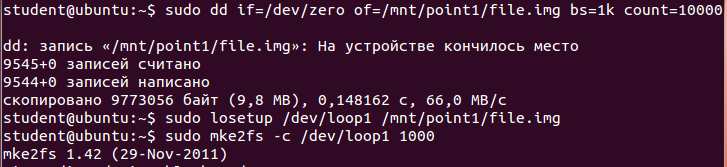
*Рис. 9.3*. Создание файловой системы на устройстве loop

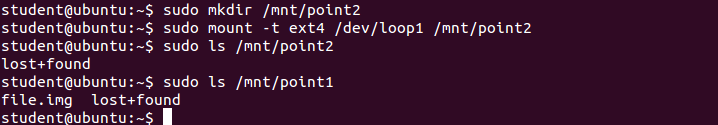
5. Примонтируем файл *file.img*, представленный блочным устройством (*/dev/loop0*) к точке */mnt/point1* с помощью команды *mount*. Обратите внимание, что указанный тип файловой системы – ext4. После монтирования вы можете обращаться к точке монтирования как к новой файловой системе с помощью команды *ls*, как видно из рисунка 9.4.



*Рис. 9.4.* Создание точки монтирования и монтирование файловой системы

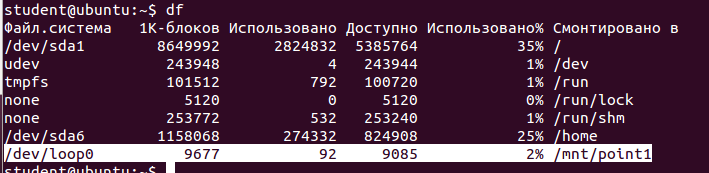
6. Как показано на рисунке 9.5, этот процесс можно продолжить, создавая новый файл в новой файловой системе, связывая его с устройством *loop* и создавая в нем еще одну файловую систему.





*Рис. 9.5*. Создание новой файловой системы *loop* в уже существующей

Из рассмотренного примера легко понять, насколько большие возможности предоставляет файловая система и устройство *loop* в Linux. Аналогичным образом с помощью устройства *loop* можно создавать в файле файловые системы с шифрованием. Это может быть полезно для защиты ваших данных. В случае необходимости такой файл можно быстро смонтировать с помощью устройства *loop*.



**Задание 9-6.** Изучите команду создания синонима каталога в дереве файловой системыLinux.

Ход выполнения:

1. Ознакомьтесь с назначением и использованием ключа *--bind* команды *mount*.

Команда *mount* с ключом *--bind* применяется в системах на ядре Linux (начиная с 2.4) для создания синонима каталога в дереве файловой системы.

Кпримеру, команда:

*mount --bind /mnt/cdrom/Files /var/ftp/cdrom*

позволит обращаться к файлам из */mnt/cdrom/Files* через путь */var/ftp/cdrom*, где */var/ftp/cdrom* – некий уже существующий (возможно, пустой) каталог (его настоящее содержимое будет недоступно до момента размонтирования).

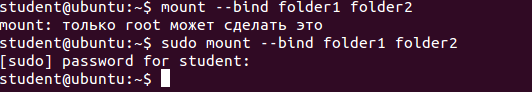
Преимуществом данного способа создания ссылок на каталоги над символьными ссылками является возможность обходить ограничения доступа к файловой системе, возникающие перед процессами, запущенными в среде *chroot* или серверами, применяющими принцип *chroot*. *chroot* — операция изменения корневого каталога в Unix-подобных операционных системах. Программа, запущенная с изменённым корневым каталогом, будет иметь доступ только к файлам, содержащимся в данном каталоге.

Например, FTP-сервер *proftpd* делает недоступными символьные ссылки, указывающие на файлы и каталоги вне определённого каталога.

2. Создайте каталог *folder1* в домашнем каталоге. Создайте в каталоге *folder1* файл с именем *file1*.

3. Создайте каталог *folder2* в домашнем каталоге. Создайте в каталоге *folder2* файл с именем *file2*.

4. Примонтируйте каталог *folder2* к каталогу *folder1*. Проверьте каталогов. Изменилось ли оно, почему?





5. Отмонтируйте каталог *folder2* от каталога *folder1*. Обратите внимание на содержимое каталогов. Сделайте выводы.

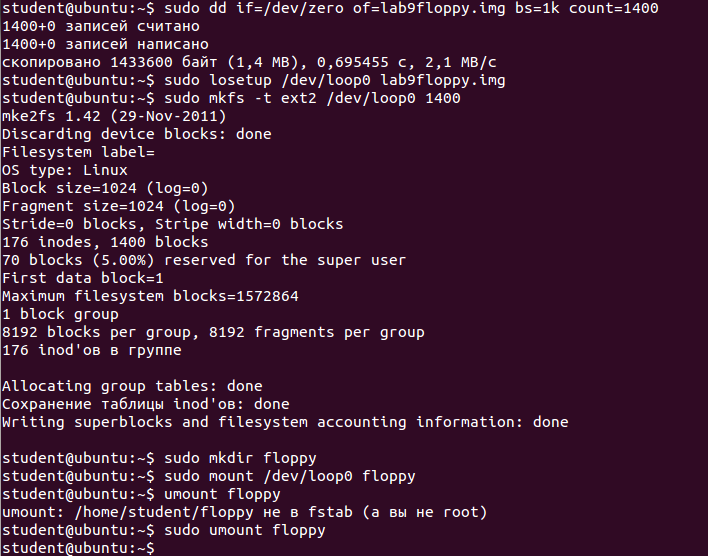


**Задание 9-7.**Изучите монтирование устройств к дереву файловой системыLinux.

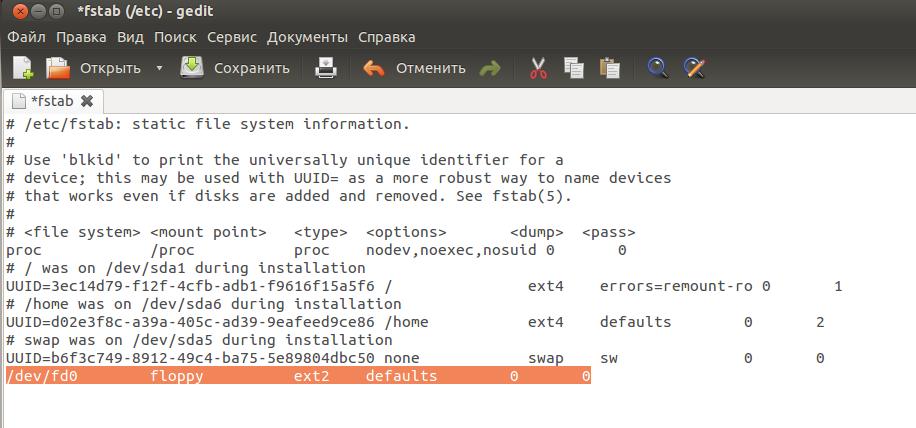
Ход выполнения:

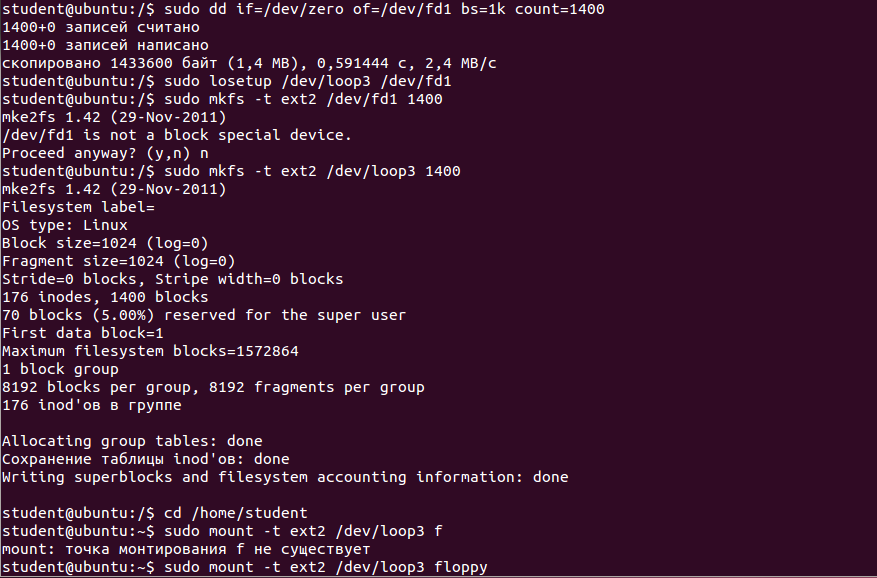
1.Создайте в домашнем каталоге папку *floppy*. Примонтируйте в нее дисковод гибких дисков с помощью команды *mount*.

2. Отмонтируйте дисковод гибких дисков.



3. Сделайте так, чтобы дисковод гибких дисков автоматически монтировался при загрузке операционной системы (используйте для этого конфигурационный файл */etc/fstab*).





1,44 Мб

