1. **Что такое утилита/команда/фильтр**

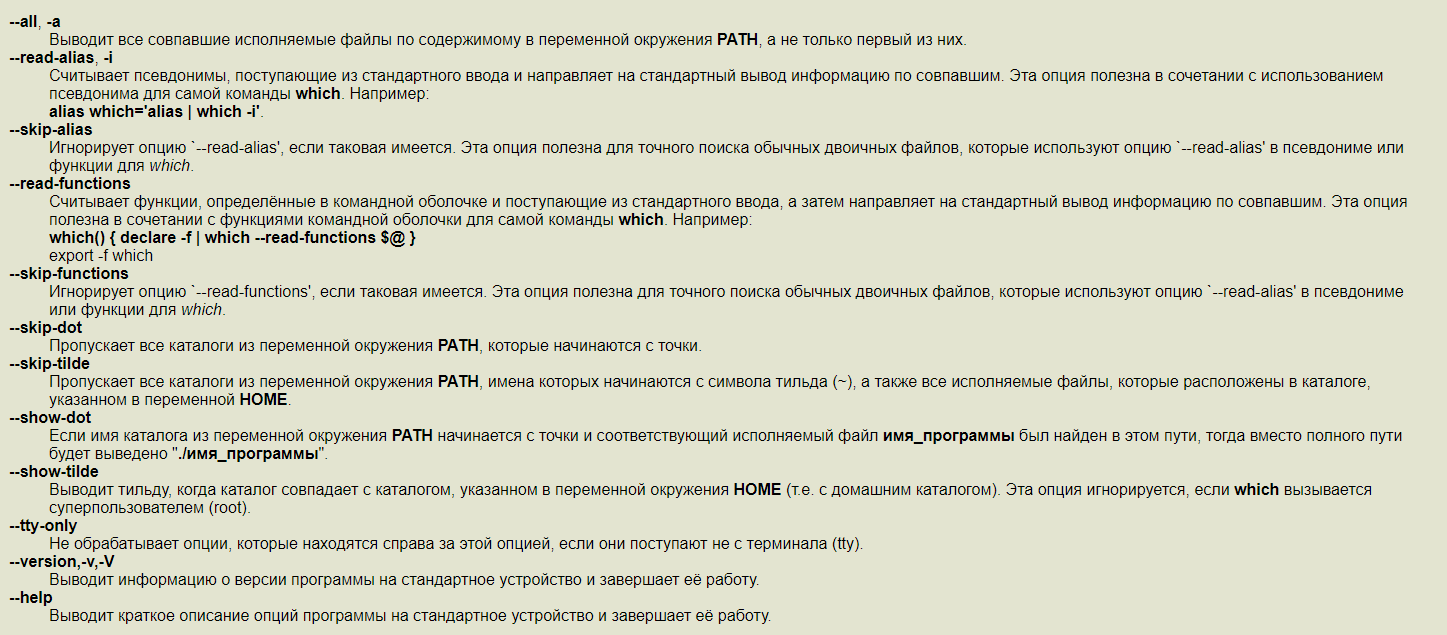
Утилита - программа для выполнения специализированных задач, связанных с работой ОС. Они предоставляют доступ к возможностям ОС. (Удаление/ создание файлов, файловые менеджеры, редактирование файлов и прочее) К ним в UNIX относят head, tail, more

**Кома́нда** — это указание [компьютерной программе](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0) действовать как некий [интерпретатор](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BF%D1%80%D0%B5%D1%82%D0%B0%D1%82%D0%BE%D1%80) для решения задачи(Интерпретатор — [программа](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0) выполняющая построчный анализ, обработка и выполнение исходного кода программы или запроса.)

Фильтры — это команды (или программы), которые воспринимают входной поток данных, производят над ним некоторые преобразования и выдают результат на стандартный вывод (откуда его можно перенаправить куда-то еще по желанию пользователя). К числу команд-фильтров относятся уже упоминавшиеся выше команды cat, more, less, wc, cmp, diff**,** а также следующие команды.

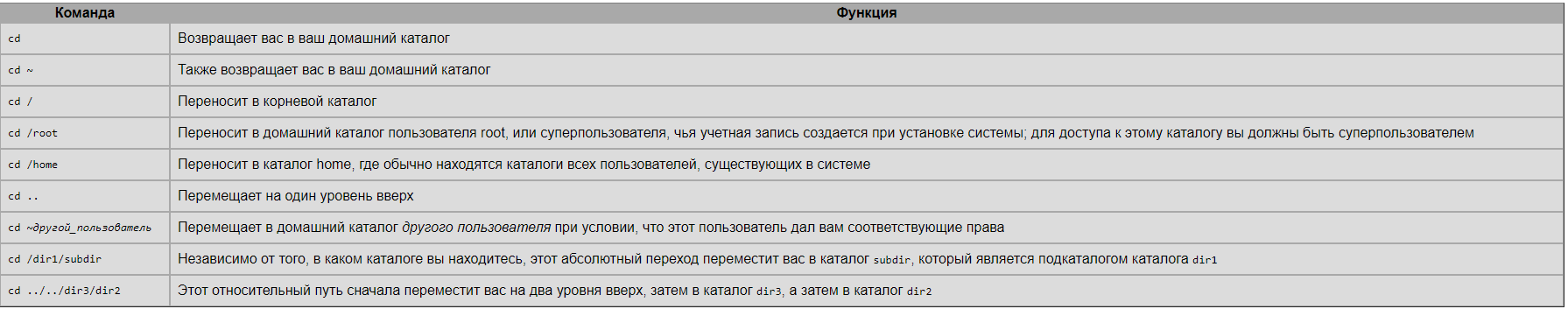
**2) команда which**

which - отображает полный путь к указанным командам или сценариям.

****

**3) команда cd, её аргументы**

**cd** также доступная как **chdir** — команда [командной строки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D0%BD%D0%BE%D0%B9_%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8) для изменения текущего [рабочего каталога](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D0%B1%D0%BE%D1%87%D0%B8%D0%B9_%D0%BA%D0%B0%D1%82%D0%B0%D0%BB%D0%BE%D0%B3) в [Unix](https://ru.wikipedia.org/wiki/Unix" \o "Unix), [DOS](https://ru.wikipedia.org/wiki/DOS) и других [операционных системах](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0)

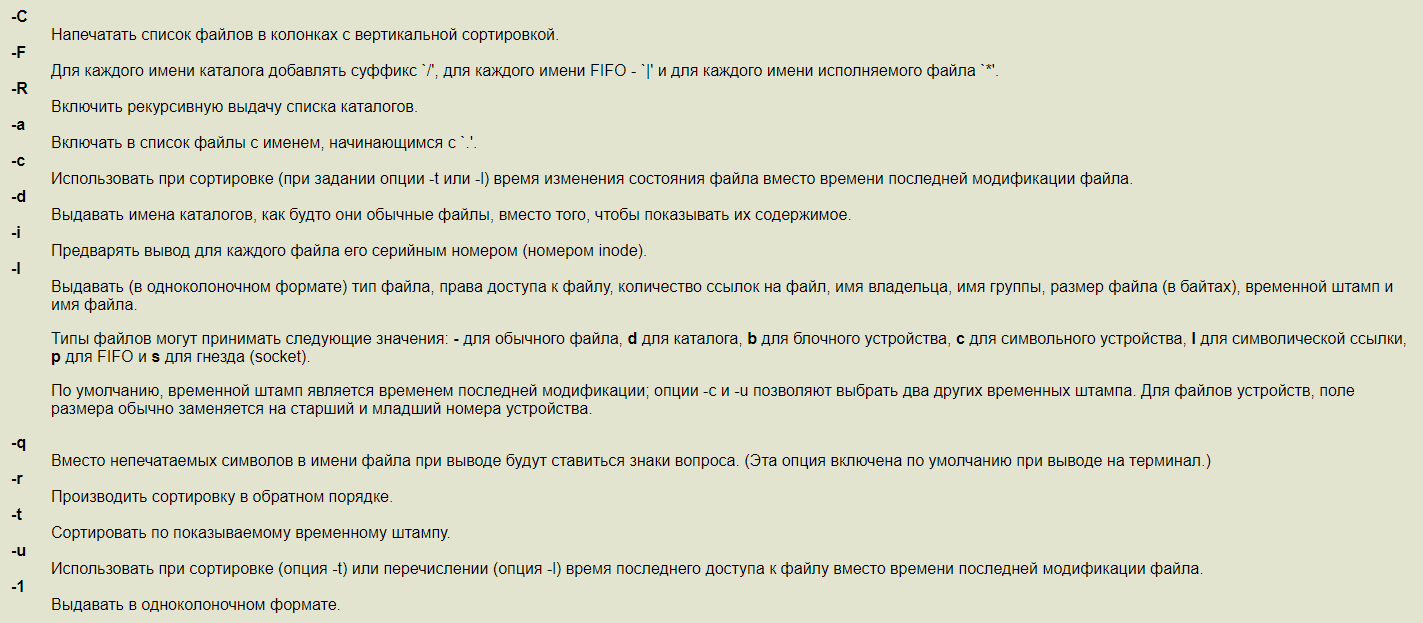
****

**4) определение пользователя системы, подстановка тильды**

SHELL заменяет тильду на абсолютный путь к домашней директории.

**5) аргументы команды ls**

ls, dir, vdir - выводит содержимое каталога



**6) что такое терминал**

[Компьютерный терминал](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D1%8B%D0%B9_%D1%82%D0%B5%D1%80%D0%BC%D0%B8%D0%BD%D0%B0%D0%BB) — устройство ввода-вывода, рабочее место на многопользовательских ЭВМ, монитор с клавиатурой. В графических оболочках UNIX-подобных операционных систем термином «терминал» называют [эмулятор терминала](https://ru.wikipedia.org/wiki/%D0%AD%D0%BC%D1%83%D0%BB%D1%8F%D1%82%D0%BE%D1%80_%D1%82%D0%B5%D1%80%D0%BC%D0%B8%D0%BD%D0%B0%D0%BB%D0%B0).

[**Эмулятор**](https://ru.wikipedia.org/wiki/%D0%AD%D0%BC%D1%83%D0%BB%D1%8F%D1%82%D0%BE%D1%80)**терминала** — это программа, которая эмулирует [терминал компьютера](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D1%80%D0%BC%D0%B8%D0%BD%D0%B0%D0%BB_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%B0) внутри некоторой другой архитектуры вывода данных на экран.

**7) блочное устройство**

**Блочное устройство** (block device) — вид файла устройств в UNIX/Linux-системах, обеспечивающий интерфейс к устройству, реальному или виртуальному, в виде файла в файловой системе .Пример : жесткий диск. **8) символьное устройство**

**Символьное устройство** — (character device) — вид файла устройства в UNIX/Linux-системах, обеспечивающий интерфейс к устройству, реальному или виртуальному, с возможностью посимвольного обмена информацией **9) что означает количество жестких ссылок в выводе ls –l**

**10) команда ln и её синтаксис**

ln - make links between files

**  
11) find**

find - search for files in a directory

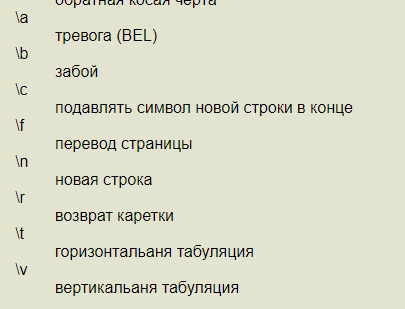
**12) символьные и жесткие ссылки, что такое inode**

Симлинк зависит от файла, на который он ссылается, и если удалить последний, симлинк перестанет работать. Жесткая ссылка (хардлинк) является копией идентификатора файла (i-node) и если удалить оригинальный файл, он останется в системе, пока не удалить все ссылки.

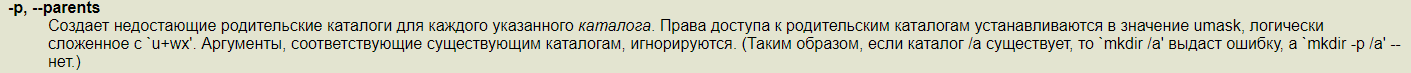
Inode означает индексный узел, в нем хранятся методанные (данные о данных). Если у вас закончились inode, вы не можете создавать новые файлы, даже если у вас есть свободное место на данном разделе.

**13) echo -e зачем ключ и как работает**

****

****

**14) mkdir, ключ –p**

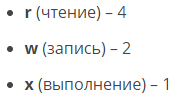
****

15) отличие каталога от файла

Ничем. Папка=Каталог=Директория. А папка тоже файл, в котором хранятся сведения о файлах в неё входящих.

**16) . И .. – ссылки**

**17) задание прав командой chmod, разные виды записи**

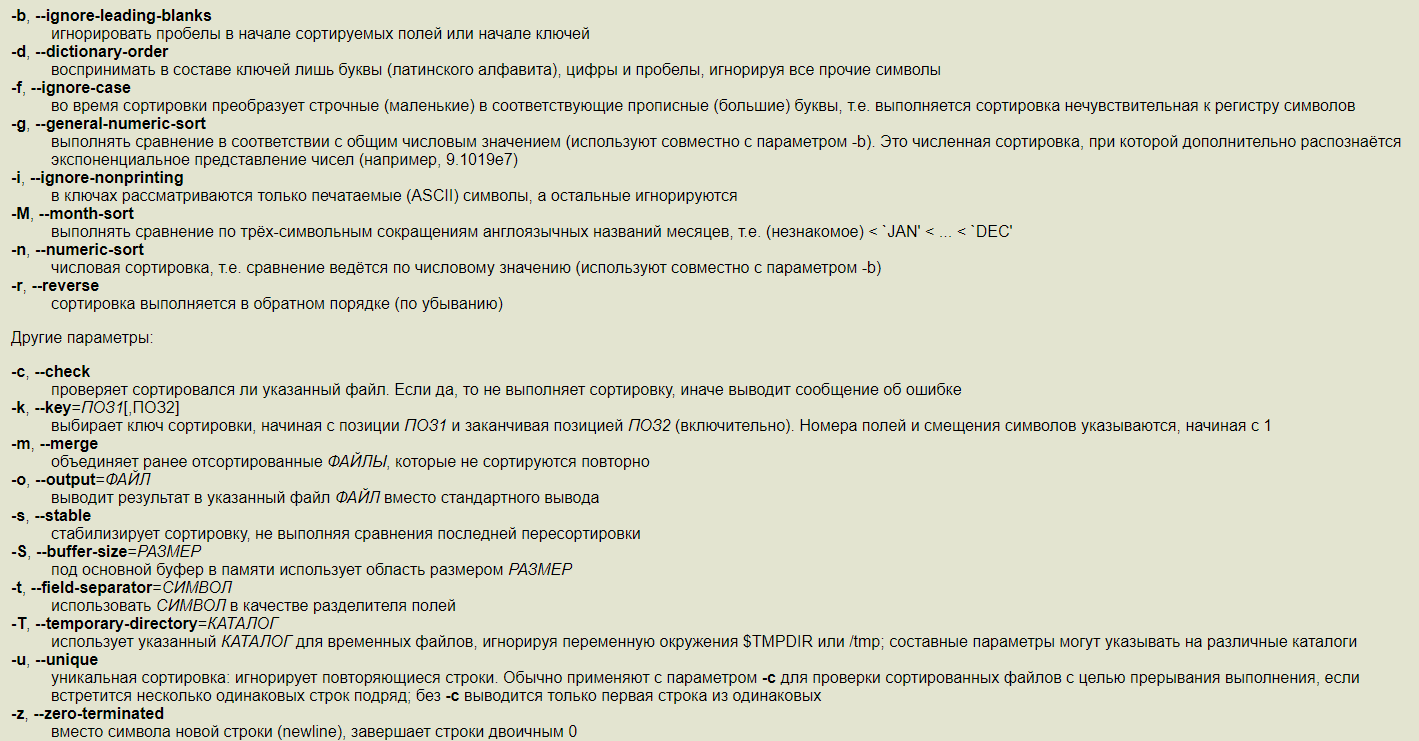
****

****

1. **Chmod u=…,g=…,o=… file1.txt**

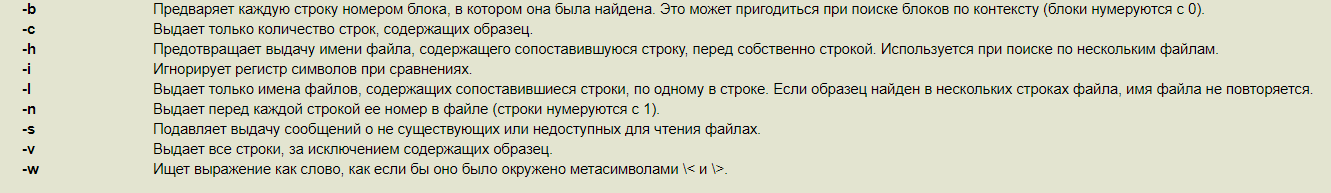
**18) команда sort и её ключи**

sort - сортирует строки текстовых файлов. **sort** [*ПАРАМЕТР*]... [*ФАЙЛ(Ы)*]...

****

**19) команда grep и её ключи**

**grep** - поиск образца в файле

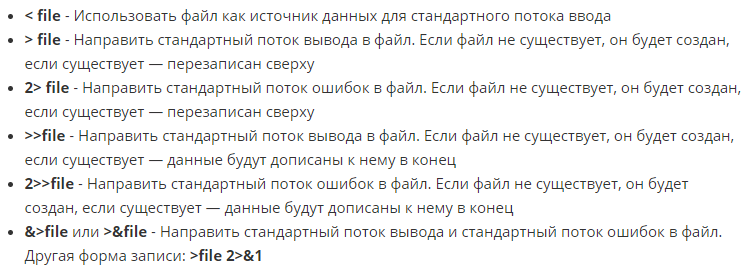
****

**20) регулярные выражения**

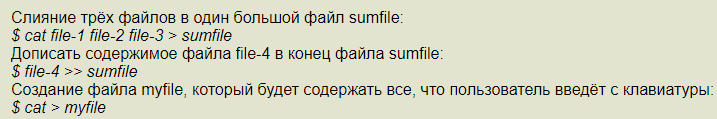
**21) создание сразу нескольких каталогов одной командой**

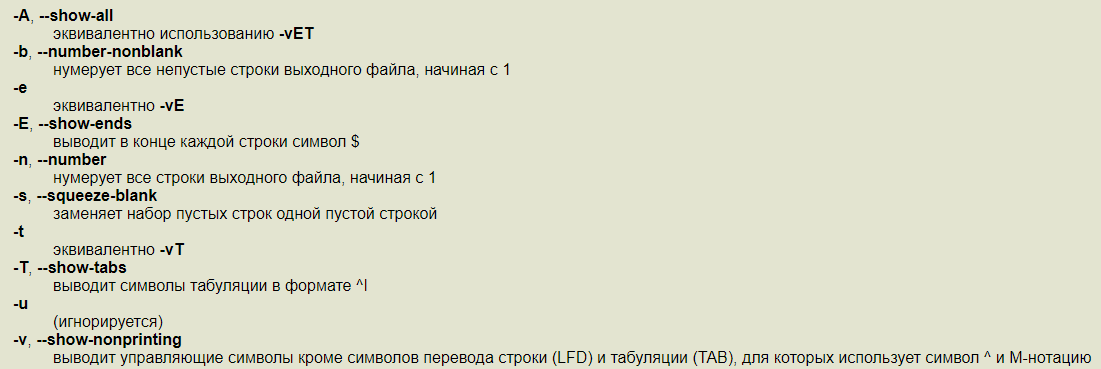
**Mkdir catalog1 catalog2 catalog3**

**22) перенаправление и его варианты**

**  
23) cat**

cat - объединяет файлы и направляет их на стандартный вывод 

****

**  
24) использование tail для отслеживания обновления файлов**

**25) сигналы ^C и ^D**

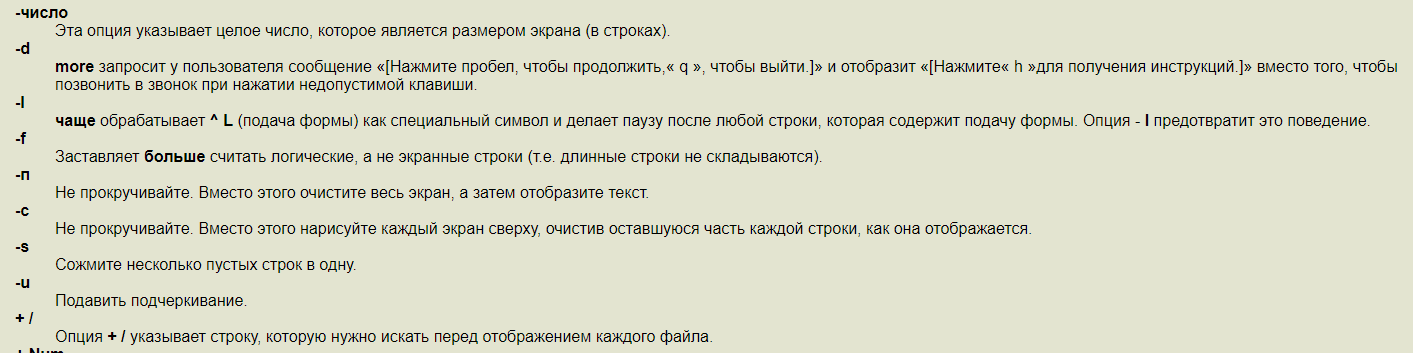
Нажатие **Ctrl + C** заставляет терминал послать сигнал SIGINT процессу, который на данный момент его контролирует. Когда foreground-программа получает сигнал SIGINT, она обязана прервать свою работу.

Нажатие **Ctrl + D** говорит терминалу, что надо зарегистрировать так называемый EOF (end of file – конец файла), то есть поток ввода окончен. Bash интерпретирует это как желание выйти из программы.

Комбинация клавиш **Ctrl + Z** посылает процессу сигнал, который приказывает ему остановиться. Это значит, что процесс остается в системе, но как бы замораживается. Само собой разумеется он уходит в бэкграунд (background) – в фоновый режим. С помощью команды bg его можно снова запустить, оставив при этом в фоновом режиме. Команда fg не только возобновляет ранее приостановленный процесс, но и выводит его из фона на передний план.

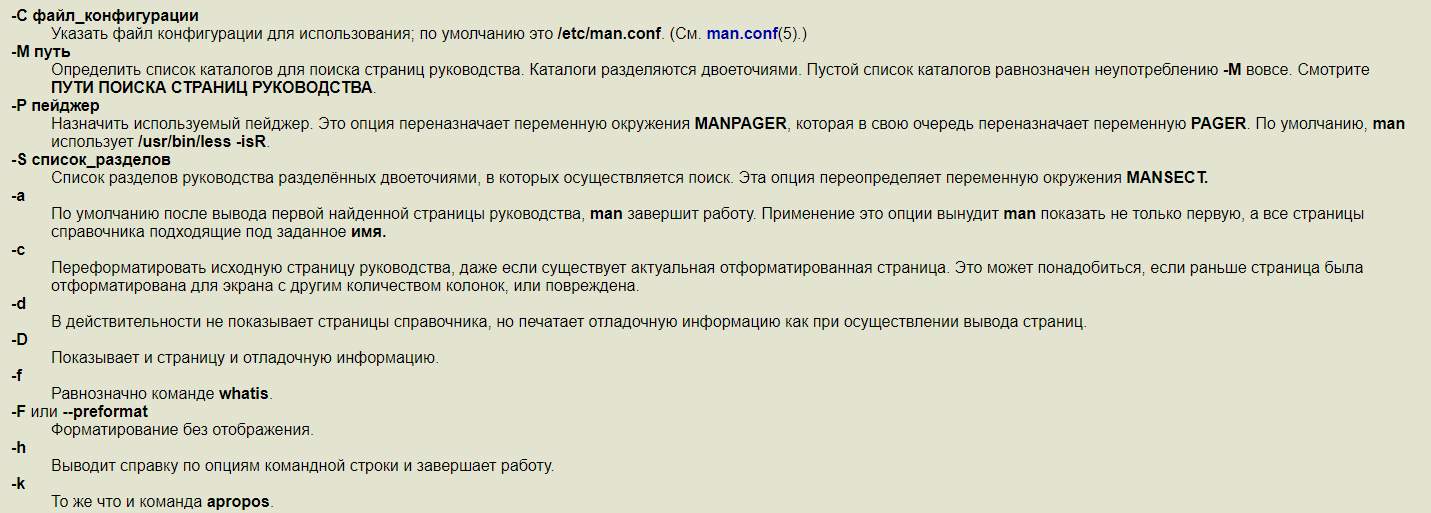
26)More

****

****

**27)Man**

****

****