# Командная оболочка ОС Linux, основные команды, скрипты.

Задание:

1. Для каждой из команд, приведенных в списке 1 выяснить её назначение и привести пример (скриншот) работы команды.
2. Создать скрипт и выполнить по вариантам задание:

Список 1. *cat, cd, clear, cp, date, df, du, echo, exit, grep, head, history, info, less, logout, ls, man, mc, mkdir, mv, nano, passwd, ps, pwd, rm, tail, touch, who*

Вариант 9. Создайте скрипт, печатающий сводку по указанному каталогу: количество текстовых файлов (\*.txt), количество исполняемых файлов (установлен атрибут х), количество каталогов, и количество всех остальных файлов.

1. cat

|  |
| --- |
| Описание: |
| cat - это сокращение от слова catenate (сцеплять, связывать). Читает данные из файла или стандартного ввода и выводит их на экран.  Синтаксис: $ cat опции файл1 файл2 ...  Можно передать утилите несколько файлов и тогда их содержимое будет выведено поочередно, без разделителей.  **Опции Описание**  -b нумеровать только непустые строки;  -E показывать символ $ в конце каждой строки;  -n нумеровать все строки;  -s удалять пустые повторяющиеся строки;  -T отображать табуляции в виде ^I;  -h отобразить справку;  -v версия утилиты. |
| Пример работы: |
| *Чтение данных из файла readme.txt, а затем hello.py*    *Чтение данных из файла с нумерацией строк*    *Отображение справки о команде при помощи опции -h* |

1. cd

|  |
| --- |
| Описание: |
| Cd (от англ. change directory — «изменение каталога») – позволяет менять текущий рабочий каталог.  Синтаксис: cd опции директория  Команда имеет множество удобных шорткодов для более быстрой навигации:  cd – перемещение в домашний каталог пользователя, аналог cd home или cd ~  cd .. – перемещение на один каталог вверх  cd - – возвращение в предыдущей каталог  cd / – перемещение в корневую директорию |
| Пример работы: |
| *Изменение текущего каталога на /.mozilla*    *Дополнительные примеры работы с демонстрацией изменения каталога* |

1. clear

|  |
| --- |
| Описание: |
| Команда clear удаляет весь видимый текст из окна терминала. При этом она не завершает сеанс и не удаляет историю команд.  Вместо ввода команды можно нажать сочетание клавиш Ctrl + L. |
| Пример работы: |
| *Очистка окна терминала (UI: X Window)*    *Очистка окна терминала (UI: Console)* |

1. cp

|  |
| --- |
| Описание: |
| Команда cp копирует файл или каталог, указанный в параметре Исходный\_файл или Исходный\_каталог, в файл или каталог, заданный в параметре Целевой\_файл или Целевой\_каталог.  Если Целевой\_файл существует, то его содержимое заменяется без предупреждения. При копировании нескольких Исходных\_файлов должен быть указан целевой каталог.  Если файл с указанным именем уже существует в целевом каталоге, то его содержимое будет заменено на содержимое исходного файла. В связи с этим рекомендуется указывать новое имя для копии файла, чтобы в целевом каталоге не было файла с тем же именем.  Синтаксис: cp опции /путь/к/файлу/источнику /путь/к/файлу/назначения  **Опции** **Описание**  --attributes-only - не копировать содержимое файла, а только флаги доступа и владельца;  -b, --backup - создать резервную копию файла назначения если он существует;  --copy-contents - копировать содержимое для специальных файлов (сокеты, файлы устройств);  -f, --force - удалить файл назначения перед попыткой записи в него если он существует;  -i, --interactive - спрашивать, нужно ли перезаписывать существующие файлы;  -n, --no-clobber - не перезаписывать существующие файлы;  -P, --no-dereference - копировать сами символические ссылки, а не то на что они указывают;  -L, --dereference - копировать не символические ссылки, а то, на что они указывают;  -l, --link - создавать жесткие ссылки вместо копирования;  --preserve - переносить указанные атрибуты с файла источника в файл назначения, возможные значения: mode, ownership, time‐stamps, context, links, xattr, all;  --no-preserve - не переносить указанные атрибуты;  --parents - сохранять путь, указанный в файле источнике, в папке назначения;  -r, --recursive - копировать папку Linux рекурсивно;  --reflink - использовать Copy on Write если это поддерживается файловой системой;  -s, --symbolic-link - не выполнять копирование файлов в Linux, а создавать символические ссылки;  -S, --suffix - указать суффикс для резервных копий файлов;  --sparse - настройка работы с разреженными файлами;  -t, --target-directory - считать файл-назначения директорией и копировать файл-источник или директорию-источник в эту директорию с оригинальным именем;  -T, --no-target-directory - считать директорию назначения файлом или директорией для записи данных. Если в качестве источника выбран файл, то он будет скопирован с новым именем. Если директория, то её содержимое будет скопировано в директорию назначения;  -u, --upgrade - скопировать файл, только если он был изменён;  -x, --one-file-system - рекурсивное копирование не должно выходить за пределы этой файловой системы;  -v, --verbose - максимально подробный вывод. |
| Пример работы: |
| *Копирование файла hello.txt* |

1. date

|  |
| --- |
| Описание: |
| Позволяет извлекать дату и время в разнообразном формате, в том числе и рассчитывать прошлое и будущее время.  Синтаксис: date опции формат  **Формат** **Значение**  %% Знак процента  %a День недели текущей локали в короткой форме ("Чтв")  %A День недели текущей локали в длинной форме ("Четверг")  %b Месяц года т.л. в короткой форме в родительном падеже ("янв")  %B Месяц года т.л. в длинной форме в родительном падеже ("января")  %c Дата и время текущей локали без указания часового пояса  %С Первые две цифры текущего года  %d Числовой день месяца с ведущим нулём  %D Дата в формате %m/%d/%y  %e День месяца; аналог %\_d  %F Дата в формате %Y-%m-%d  %h Аналог %b  %H Часы (00..23)  %I Часы (01..12)  %j День года (001..366)  %m Месяц (01..12)  %M Минуты (00..59)  %n Новая строка  %q Квартал года  %S Секунды (00..59)  %t Знак табуляции  %T Время в формате %H:%M:%S  %u Числовой день недели; 1 — понедельник  %x Дата в локальном формате  %X Время в локальном формате  %Z Аббревиатура временной зоны |
| Пример работы: |
| *Отображение текущей даты и времени*  \  *Отображение времени последней модификации файла* |

1. df

|  |
| --- |
| Описание: |
| Позволяет выводить список подключённых устройств, информацию о занятом месте и точку монтирования.  Синтаксис: df опции устройство  **Опции Описание**  -a, --all Отобразить все файловые системы, в том числе виртуальные, псевдо и недоступные;  -B Изменить размер одного блока перед выводом данных, например, можно использовать BM, чтобы вывести все данные в мегабайтах;  -h Выводить размеры в читаемом виде, в мегабайтах или гигабайтах;  -H Выводить все размеры в гигабайтах;  -i Выводить информацию об inode;  -k Выводить размеры в килобайтах;  --output Использовать специальный формат вывода, если не задано, выводит все поля. Доступны такие варианты: 'source', 'fstype', 'itotal', 'iused', 'iavail', 'ipcent', 'size', 'used', 'avail', 'pcent', 'file' и 'target';  -P Использовать формат вывода POSIX;  --total Выводить всю инф-ию про использованное и доступное место;  -t, --type Выводить информацию только про указанные файловые системы;  -x Выводить инф-ю обо всех, кроме указанных файловых систем; |
| Пример работы: |
| *Вывод списка подключённых устройств,*  *информации о занятом месте и точке монтирования* |

1. du

|  |
| --- |
| Описание: |
| Позволяет вывести размер всех файлов в определённой папке в байтах или в более удобном формате.  Синтаксис: du опции /путь/к/папке |
| Пример работы: |
| *Вывод размера всех файлов в папке .cache* |

1. echo

|  |
| --- |
| Описание: |
| Выводит текст или переменные на стандартный вывод. Широко используется в скриптах командной оболочки для вывода текстовых сообщений или переменных.  Синтаксис: echo опции строка  **Опции Описание**  -n не выводить перевод строки;  -e включить поддержку вывода Escape последовательностей;  -E отключить интерпретацию Escape последовательностей.  **Escape последовательности для вставки специальных символов:**  /c удалить перевод строки;  /t горизонтальная табуляция;  /v вертикальная табуляция;  /b удалить предыдущий символ;  /n перевод строки;  /r символ возврата каретки в начало строки. |
| Пример работы: |
| *Вывод текста “My name is” и переменной Author*    *Вывод текста “My name is…” с интерпретацией*  *Escape последовательностей* |

1. exit

|  |
| --- |
| Описание: |
| Позволяет выйти из оболочки, в которой она запущена.  Синтаксис: exit число  число - необязательный код выхода, числовое значение от 0 до 255. |
| Пример работы: |
| *Попытка выйти из оболочки с кодом выхода* |

1. grep

|  |
| --- |
| Описание: |
| Команда для поиска текста по шаблону с использованием регулярных выражений. Она выполняет поиск в одном или нескольких файлах, выводя все строки, которые соответствуют указанному шаблону.  grep позволяет искать в нескольких файлах и даже в каталоге, используя рекурсивный режим.  Синтаксис: grep опции шаблон /путь/к/файлу/или/папке, где:  Опции – доп. параметры – различные настройки поиска и вывода.  Шаблон – строка или регулярное выражение, по которому будет выполняться поиск. |
| Пример работы: |
| *Поиск текста «hello world» в файле «hello.c»*  *с выводом номера строки*    *Поиск текста «1594029461» в файле «.wine/user.reg»*  *с выводом номера строки* |

1. head

|  |
| --- |
| Описание: |
| Используется для отображения первых нескольких строк (или байтов) одного или нескольких текстовых файлов. По умолчанию она выводит первые 10 строк файла.  Синтаксис: head опции файл  **Опции Описание**  -c (--bytes) - позволяет задавать количество текста не в строках, а в байтах. При записи в виде --bytes=[-]NUM выводит на экран все содержимое файла, кроме NUM байт, расположенных в конце документа.  -n (--lines) - показывает заданное количество строк вместо 10, которые выводятся по умолчанию. Если записать эту опцию в виде --lines=[-]NUM, будет показан весь текст кроме последних NUM строк.  -q (--quiet, --silent) - выводит только текст, не добавляя название файла.  -v (--verbose) - перед текстом выводит название файла.  -z (--zero-terminated) - символы перехода на новую строку заменяет символами завершения строк. |
| Пример работы: |
| *Отображение первых двух строк файла «hello.c»*  *с выводом названия файла* |

1. history

|  |
| --- |
| Описание: |
| Отображает список ранее введённых команд в терминале. По умолчанию в нём отображаются последние 1000 команд.  Синтаксис: history опции файл  В качестве файла можно указать файл истории. По умолчанию история для текущего пользователя хранится в файле ~/.history (В нашем случае ~/.ash\_history).  **Опции Описание**  -c - очистить историю;  -d - удалить определенную строку из истории;  -a - добавить новую команду в историю;  -n - скопировать команды из файла истории в текущий список;  -w - перезаписать содержимое одного файла истории в другой, заменяя повторяющиеся вхождения. |
| Пример работы: |
| *Отображение списка введённых команд в терминале* |

1. info

|  |
| --- |
| Описание: |
| Отображает информацию в формате документа. Она похожа на команду man, но более надёжна для связывания страниц.  Информационные страницы создаются с помощью инструментов texinfo, могут связываться с другими страницами и создавать меню.  Синтаксис: info опции файл  **Опции Описание**  -a, --a - использовать все подходящие справочники  -k, --apropos= - поиск строки во всех указателях всех справочников  -d, --directory= - добавить каталог в INFOPATH  -f, --file= - задать открываемый Info-справочник  -h, --help - показать эту справку и выйти  --index-search= - перейти к ноде в пункте строка в указателе  -n, --node= - задать ноды в файле Info, посещаемом впервые  -o, --output= - запись выбранных нод в файл  --subnodes - рекурсивно выводить пункты меню  -v, --variable пер=знач - назначить значение переменной Info  --version - показать информацию о версии и выйти  -w, --where, --location - показать физическое расположение файла Info |
| Пример работы: |
| *Info* |

1. less

|  |
| --- |
| Описание: |
| Позволяет перематывать текст не только вперёд, но и назад, осуществлять поиск в обоих направлениях, переходить сразу в конец или в начало файла.  Особенность less: команда не считывает текст полностью, а загружает его небольшими фрагментами.  Синтаксис: less опции файл |
| Пример работы: |
|  |

1. logout

|  |
| --- |
| Описание: |
| - |
| Пример работы: |
|  |

1. ls

|  |
| --- |
| Описание: |
| - |
| Пример работы: |
|  |

1. man

|  |
| --- |
| Описание: |
| - |
| Пример работы: |
|  |

1. mc

|  |
| --- |
| Описание: |
| - |
| Пример работы: |
|  |

1. mkdir

|  |
| --- |
| Описание: |
| Создание новой директории (каталога, папки).  Синтаксис: mkdir опции директория  -p (-parents) – создаёт промежуточные каталоги, если они не существуют. |
| Пример работы: |
| *Создание новой папки «saifor», а затем каталога «saifor/maslov/vladislav/pi31» при помощи опции -p*    *Мы также можем переместиться в каталог*  *при помощи текстового редактора Qemacs* |

1. mv

|  |
| --- |
| Описание: |
| Используется для перемещения или переименования файлов и директорий.  Синтаксис: mv опции источник назначение  **Опции** **Описание**  -help Вывод на экран оф. документацию об утилите  -version Отображает версию mv  -b / -backup / -backup= Создает копию файлов  -f Не спрашивать разрешение у владельца файла  -i Спросить разрешение у владельца  -n Отключает перезапись уже существующих объектов  -strip-trailing-slashes Удаляет завершающий символ / у файла  -t Перемещает все файлы в указанную директорию  -u Перемещение только если исходный файл новее объекта назначения  -v Отображает сведения о каждом элементе во время обработки команды |
| Пример работы: |
| *Перемещение файла «hello.txt» в папку «saifor»* |

1. nano

|  |
| --- |
| Описание: |
| Консольный текстовый редактор для UNIX и Unix-подобных операционных систем.  Синтаксис: nano опции /адрес/файла/файл  **Опции** **Описание**  +номер Указать строку в файле, с которой нужно начинать редактирование;  -B Создавать резервную копию для файла при сохранении;  -С Папка для резервной копии;  -D Выводить текст жирным;  -E Конвертировать табуляции в пробелы;  -F Включить поддержку редактирования нескольких файлов;  -H Сохранять историю поиска и замены;  -I Не читать конфигурацию из файлов nanorc;  -P Запоминать и восстанавливать позицию курсора во время последнего редактирования;  -T Размер табуляции в пробелах;  -m Включить поддержку мыши;  -v Режим только для чтения, вы не можете сохранить файл. |
| Пример работы: |
| *Файл «hello.c» в текстовом редакторе GNU NANO*    *Файл «hello.txt» в текстовом редакторе GNU NANO* |

1. passwd

|  |
| --- |
| Описание: |
| Позволяет управлять паролями пользователей. Список пользователей в Linux хранится в файле /etc/passwd, пароли же выделены в отдельный файл - /etc/shadow.  Синтаксис: passwd опции пользователь  **Опции** **Описание**  -d удалить пароль пользователя, после этого он не сможет войти  -e сделать пароль устаревшим  -i через сколько дней после того, как пароль устарел, отключить аккаунт, если пользователь не сменил пароль  -l запретить пользователю входить в систему  -n минимальное количество дней между сменами пароля  -S отобразить информацию об аккаунте  -u отменяет действие параметра -l  -x максимальное количество дней, пока пароль можно использовать.  -w количество дней, после которых нужно предупреждать пользователя о том, что надо сменить пароль. |
| Пример работы: |
| *Попытка изменить пароль текущего пользователя (root)*    *Список пользователей в файле «/etc/passwd»*    *Список паролей в файле «/etc/shadow»* |

1. ps

|  |
| --- |
| Описание: |
| Позволяет отслеживать активные процессы. С её помощью можно получить информацию о процессах, включая их ID, приоритет, использование памяти и процессора.  Синтаксис: ps опции  **Опции** **Описание**  -A, -e выбрать все процессы  -a выбрать все процессы, кроме фоновых  -d выбрать все процессы, даже фоновые, кроме процессов сессий  -N выбрать все процессы кроме указанных  -С выбирать процессы по имени команды  -G выбрать процессы по ID группы  -p выбрать процессы PID  --ppid выбрать процессы по PID родительского процесса  -s выбрать процессы по ID сессии  -t выбрать процессы по tty  -u выбрать процессы пользователя  **Опции форматирования**  -с отображать информацию планировщика  -f вывести максимум доступных данных (н-р, количество потоков)  -F аналогично -f, только выводит ещё больше данных  -l длинный формат вывода  -j вывести процессы в стиле Jobs, минимум информации  -M добавить информацию о безопасности  -o позволяет определить свой формат вывода  --sort выполнять сортировку по указанной колонке  -L отображать потоки процессов в колонках LWP и NLWP  -m вывести потоки после процесса  -V вывести информацию о версии  -H отображать дерево процессов |
| Пример работы: |
| *Отображение активных процессов* |

1. pwd

|  |
| --- |
| Описание: |
| Позволяет вывести в терминал путь к текущей папке.  Синтаксис: pwd опции  **Опции** **Описание**  -L, --logical Брать директорию из переменной окружения, даже если она содержит символические ссылки;  -P Отбрасывать все символические ссылки;  --help Отобразить справку по утилите;  --version Отобразить версию утилиты. |
| Пример работы: |
| *Вывод в терминала пути к текущей папке* |

1. rm

|  |
| --- |
| Описание: |
| Используется для удаления файлов и каталогов из файловой системы.  Синтаксис: rm опции источник назначение  **Опции Описание**  -f, --force - игнорировать несуществующие файлы и аргументы.  -i - запрашивать подтверждение перед каждым удалением.  -I - запрашивать подтверждение один раз перед удалением более трёх файлов или перед рекурсивным удалением; не так назойливо, как с -i, но всё же защищает от большинства ошибок.  --interactive[=КОГДА] - запрашивать подтверждение в соответствии с КОГДА: never, once (-I) или always (-i); если КОГДА не задано – запрашивать всегда.  --one-file-system - при рекурсивном удалении дерева, пропускать все каталоги, находящиеся не на той же файловой системе, что и соответствующий аргумент командной строки.  --no-preserve-root - не обрабатывать / особым образом.  --preserve-root[=all] - не удалять / (по умолчанию); при указании all отвергать любой аргумент командной строки на отдельном устройстве от его родителя.  -r, -R, --recursive - рекурсивно удалять каталоги и их содержимое.  -d, --dir - удалять пустые каталоги.  -v, --verbose - пояснять производимые действия. |
| Пример работы: |
| *Удаление файла «fileforRM»*    *Удаление файла «root» с запросом подтверждения перед удалением* |

1. tail

|  |
| --- |
| Описание: |
| Выводит несколько (по умолчанию 10) последних строк из файла.  Синтаксис: mv опции файл  **Опции Описание**  -c - выводить указанное количество байт с конца файла;  -f - обновлять информацию по мере появления новых строк в файле;  -n - выводить указанное количество строк из конца файла;  --pid - используется с -f, позволяет завершить работу утилиты, когда завершится указанный процесс;  -q - не выводить имена файлов;  --retry - повторять попытки открыть файл, если он недоступен;  -v - выводить подробную информацию о файле; |
| Пример работы: |
| *Вывод 10 последних строк из файла «Xorg.0.log»* |

1. touch

|  |
| --- |
| Описание: |
| Используется для создания пустых файлов и обновления временных меток существующих файлов.  Синтаксис: touch опции файл  **Опции Описание**  -a - отвечает за изменение времени доступа к указанному файлу  -m - изменяет время модификации  -c - определяет, что объект с указанным именем создан не будет  -r - позволит использовать время доступа и модификации файла  -t - предназначена для изменения даты и времени путем ручного ввода  -d - использует дату и время, заданные в виде строки  --help - даст возможность прочесть официальную документацию  --version - отобразит текущую версию утилиты |
| Пример работы: |
| *Создание пустого файла «TOUCHFILE»*      *Создание пустого файла «touchf.txt» с изменённой датой и временем* |

1. who

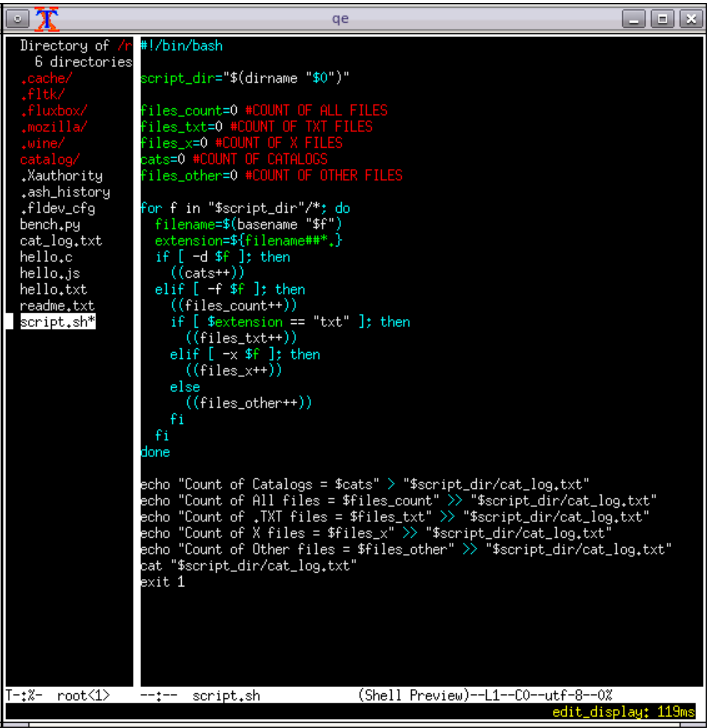
|  |
| --- |
| Описание: |
| Выводит список всех залогиненных пользователей вместе с информацией о их сеансах входа.  Синтаксис: who опции имя\_файла аргументы  -a (--all) - включает в себя все основные опции.  - b (--boot) - показывает время загрузки операционной системы.  -d (--dead) - выводит перечень зомби-процессов.  - H (--heading) - добавляет колонкам заголовки  -m - показать пользователя, который сейчас работает в терминале.  -r - вывести текущий уровень запуска (runinit);  -t - показать последнее изменение системных часов;  -s - вывести только имя, терминальную сессию и время.  -q - вывести количество авторизованных пользователей.  -T - данные о терминальной сессии.  -u - показать активных пользователей.  --ips - вместо названия хостов показывает ips.  --lookup - используется в сочетании с --ips, выводит данные, которые основываются на сохраненном IP, если он доступен, а не на названии хоста. |
| Пример работы: |
|  |

Задание 2: Создать скрипт и выполнить по вариантам задание.

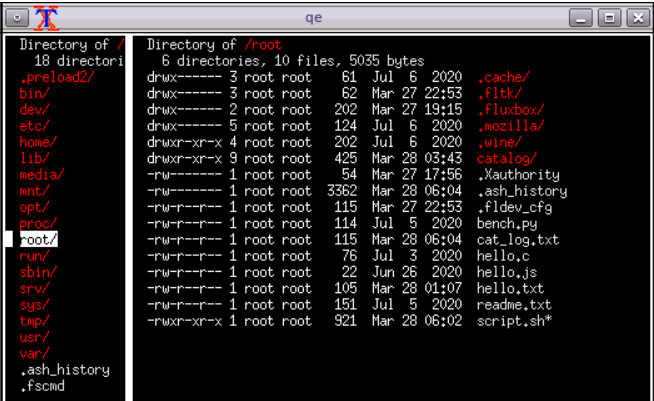
Вариант 9. Создайте скрипт, печатающий сводку по указанному каталогу: количество текстовых файлов (\*.txt), количество исполняемых файлов (установлен атрибут х), количество каталогов, и количество всех остальных файлов.

Скрипт:

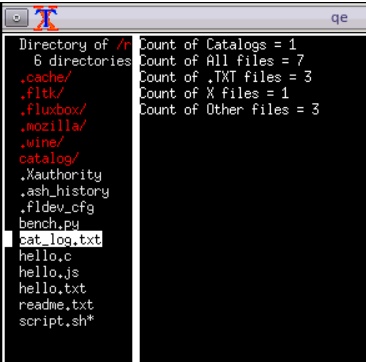
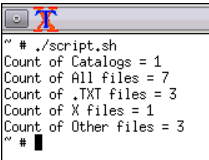
1. #!/bin/bash
2. files\_count=0
3. files\_txt=0
4. files\_x=0
5. cats=0
6. files\_other=0
7. for f in $HOME/catalog/\*; do
8. filename=$(basename "$f")
9. extension=${filename##\*.}
10. if [ -d $f ]; then
11. ((cats++))
12. elif [ -f $f ]; then
13. ((files\_count++))
14. if [ $extension == "txt" ]; then
15. ((files\_txt++))
16. elif [ -x $f ]; then
17. ((files\_x++))
18. else
19. ((files\_other++))
20. fi
21. fi
22. done
23. echo "Count of Catalogs = $cats" > cat\_log.txt
24. echo "Count of All files = $files\_count" >> cat\_log.txt
25. echo "Count of .TXT files = $files\_txt" >> cat\_log.txt
26. echo "Count of X files = $files\_x" >> cat\_log.txt
27. echo "Count of Other files = $files\_other" >> cat\_log.txt
28. cat cat\_log.txt
29. exit 1



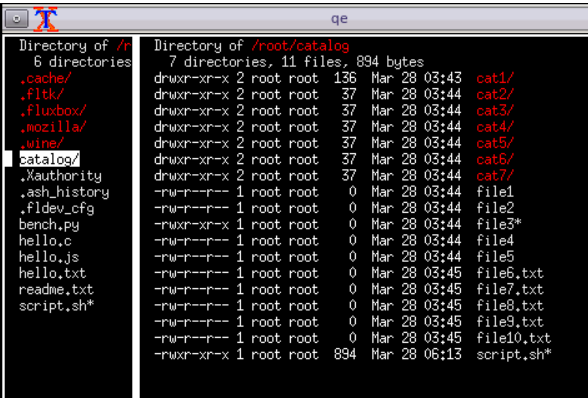
Директория /root:



Результат выполнения файл-скрипта:



Директория /root/catalog:



Результат выполнения файл-скрипта:

