grep(1)

НАЗВАНИЕ

grep - поиск образца в файле

СИНТАКСИС

```
/usr/bin/grep [ -bchilnsvw ] ограниченное_регулярное_выражение
[ имя_файла ... ]
/usr/xpg4/bin/grep [ -E | -F ] [ -c | -l | -q ] [ -bhinsvwx ]
-e список_образцов ... [ -f файл_образцов ] ...
[ имя_файла ... ]
/usr/xpg4/bin/grep [ -E | -F ] [ -c | -l | -q ] [ -bhinsvwx ]
[ -e список_образцов ... ] -f файл_образцов ...
[ имя_файла ... ]
/usr/xpg4/bin/grep [ -E | -F ] [ -c | -l | -q ] [ -bhinsvwx ]
образец [ имя_файла ... ]
```

ОПИСАНИЕ

Утилита **grep** выполняет поиск *образца* в текстовых файлах и выдает все строки, содержащие этот образец. Она использует компактный недетерминированный алгоритм сопоставления.

Будьте внимательны при использовании в **списке_образцов** символов \$, *, [, ^, |, (,) и \, поскольку они являются метасимволами командного интерпретатора. Лучше брать весь **список_образцов** в одиночные кавычки '... '.

Если **имя_файла** не указано, **grep** предполагает поиск в стандартном входном потоке. Обычно каждая найденная строка копируется в стандартный выходной поток. Если поиск осуществлялся в нескольких файлах, перед каждой найденной строкой выдается имя файла.

/usr/bin/grep

Утилита **/usr/bin/grep** использует для задания образцов *ограниченные регулярные выражения*, описанные на странице справочного руководства **regex(5)**.

/usr/xpg4/bin/grep

Опции **-E** и **-F** влияют на способ интерпретации списка_образцов программой **/usr/xpg4/bin/grep**. Если указана опция **-E**, программа **/usr/xpg4/bin/grep** интерпретирует образцы в списке как полные регулярные выражения (см. **описание опции -E**). Если же указана опция **-F**, **grep** интерпретирует **список_образцов** как фиксированные строки. Если ни одна из этих опций не указана, **grep** интерпретирует элементы **списка_образцов** как простые регулярные выражения, описанные на странице справочного руководства **regex(5)**.

ОПЦИИ

Следующие опции поддерживаются обеими программами, `/usr/bin/grep` и `/usr/xpg4/bin/grep`:

- b** Предваряет каждую строку номером блока, в котором она была найдена. Это может пригодиться при поиске блоков по контексту (блоки нумеруются с 0).
- c** Выдает только количество строк, содержащих образец.
- h** Предотвращает выдачу имени файла, содержащего сопоставившуюся строку, перед собственно строкой. Используется при поиске по нескольким файлам.
- i** Игнорирует регистр символов при сравнениях.
- l** Выдает только имена файлов, содержащих сопоставившиеся строки, по одному в строке. Если образец найден в нескольких строках файла, имя файла не повторяется.
- n** Выдает перед каждой строкой ее номер в файле (строки нумеруются с 1).
- s** Подавляет выдачу сообщений о не существующих или недоступных для чтения файлах.
- v** Выдает все строки, за исключением содержащих образец.
- w** Ищет выражение как слово, как если бы оно было окружено метасимволами `<` и `>`.

`/usr/xpg4/bin/grep`

Следующие опции поддерживаются только утилитой `/usr/xpg4/bin/grep`:

- e список_образцов** Задает один или несколько образцов для поиска. Образцы в **списке_образцов** должны разделяться символами новой строки. Пустой образец можно задать, введя два символа новой строки подряд. Если одновременно с этой опцией не указана опция **-E** или **-F**, каждый образец будет рассматриваться как простое регулярное выражение. Утилита **grep** воспринимает несколько опций **-e** и **-f**. При поиске строк, соответствующих образцу, используются все заданные образцы, но порядок сопоставления не определен.

-E Сопоставлять с полными регулярными выражениями. Рассматривать каждый заданный образец как *полное регулярное выражение*. Если любое из полных регулярных выражений-образцов сопоставляется с входной строкой, строка считается соответствующей. Пустое полное регулярное выражение соответствует любой строке. Каждый образец будет интерпретироваться как полное регулярное выражение в соответствии с описанием на странице справочного руководства **regex (5)**, за исключением метасимволов `\(` и `\)`, причем:

1. Полное регулярное выражение, за которым идет `+`, соответствует одному или более вхождениям полного регулярного выражения.
2. Полное регулярное выражение, за которым идет `?`, соответствует 0 или одному вхождению полного регулярного выражения.
3. Полным регулярным выражениям, разделенным символами `|` или символами новой строки, соответствуют строки, сопоставляющиеся с любым из указанных выражений.
4. Полные регулярные выражения можно брать в круглые скобки `()` для группировки.

-f файл_образцов Максимальный приоритет имеют операторы `[]`, затем `*?+`, конкатенация, и, наконец, оператор `|` и символ новой строки. Читает один или несколько образцов из файла с указанным полным именем **файл_образцов**. Образцы в **файле_образцов** завершаются символом новой строки. Пустой образец можно задать с помощью пустой строки в **файле_образцов**. Если только вместе с этой опцией не указана опция **-E** или **-F**, каждый образец считается простым регулярным выражением.

-F Задаёт сопоставление с фиксированными строками. Каждый образец ищется как строка, а не как регулярное выражение. Если входная строка содержит любой из образцов в качестве подряд идущих байтов, такая строка считается соответствующей образцу. Пустая строка-образец соответствует любой строке. Подробнее см. на странице **fgrep (1)**.

-q Немногословный режим. В стандартный выходной поток не выдается ничего, кроме сопоставившихся строк. Если одна из входных строк соответствует образцу, возвращается статус выхода 0.

-x Считает сопоставившимися только строки, все символы которых использованы при сопоставлении с фиксированной строкой или регулярным выражением.

ОПЕРАНДЫ

Поддерживаются следующие операнды:

имя_файла Имя файла, в котором должен выполняться поиск по образцу. Если файлы не указаны, поиск ведется в стандартном входном потоке.

/usr/bin/grep

образец Задаёт образец для поиска во входных строках.

/usr/xpg4/bin/grep

образец Задаёт один или несколько образцов для поиска во входных строках. Этот операнд используется так же, как если бы он был задан в виде **-список_образцов**.

ИСПОЛЬЗОВАНИЕ

Опция **-epattern_list** имеет тот же эффект, что и операнд **список_образцов**, но позволяет начинать **список_образцов** с дефиса. Она также пригодится в случаях, когда удобнее задавать несколько образцов в виде отдельных аргументов.

Можно задавать несколько опций **-e** и **-f**. При этом утилита **grep** использует все заданные образцы при сопоставлении с входными строками. (Учтите, что порядок проверки не задается. Если реализация находит среди образцов пустую строку, она может искать сначала именно ее, тем самым, сопоставление будет найдено для каждой строки, а остальные образцы, по сути, - проигнорированы.)

Опция **-q** даёт средства простого определения, находится ли образец (или строка) в группе файлов. При поиске в нескольких файлах она обеспечивает более высокую производительность (поскольку позволяет завершить работу, как только будет найдено первое соответствие) и не требует дополнительных усилий пользователя при формировании набора файлов-аргументов (поскольку **grep** вернет нулевой статус выхода при обнаружении соответствия даже если при работе с предыдущими операндами-файлами произошла ошибка доступа или чтения.)

Работа с большими файлами

Описание поведения утилиты при работе с файлами размером от 2 Гбайтов (2**31 байтов) см. на странице справочного руководства **largefile(5)**.

ПРИМЕРЫ

Пример 1: Поиск всех вхождений слова

Чтобы найти все вхождения слова "Posix" (независимо от регистра) в файле text.mm и выдать номера соответствующих строк:

```
example% /usr/bin/grep -i -n posix text.mm
```

Пример 2: Поиск пустых строк

Чтобы найти все пустые строки в стандартном входном потоке:

```
example% /usr/bin/grep ^$
```

или

```
example% /usr/bin/grep -v .
```

Пример 3: Поиск строк, содержащих фиксированные подстроки

Обе следующих команды выдают все строки, содержащие подстроки abc, def или и ту, и другую:

```
example% /usr/xpg4/bin/grep -E 'abc def'
```

```
example% /usr/xpg4/bin/grep -F 'abc def'
```

Пример 4: Поиск строк, соответствующих образцу

Обе следующих команды выдают все строки abc или def:

```
example% /usr/xpg4/bin/grep -E '^abc$ ^def$'
```

```
example% /usr/xpg4/bin/grep -F -x 'abc def'
```

ПЕРЕМЕННЫЕ СРЕДЫ

Описание следующих переменных среды **LC_COLLATE**, **LC_CTYPE**, **LC_MESSAGES** и **NLSPATH**, влияющих на работу команды **grep**, см. на странице справочного руководства **environ(5)**.

СТАТУС ВЫХОДА

Команда завершается со следующими статусами выхода:

- 0** Найдена одна или несколько соответствующих строк.
- 1** Соответствующие строки не найдены.
- 2** Выявлены синтаксические ошибки или недоступные файлы (даже если были найдены соответствующие строки).

АТТРИБУТЫ

Описание следующих атрибутов см. на странице справочного руководства **attributes(5)**:

/usr/bin/grep

ТИП АТТРИБУТА	ЗНАЧЕНИЕ АТТРИБУТА
Доступен в пакете	SUNWcsu
CSI	включено

/usr/xpg4/bin/grep

ТИП АТТРИБУТА	ЗНАЧЕНИЕ АТТРИБУТА
Доступен в пакете	SUNWxcu4
CSI	включено

ССЫЛКИ

[egrep\(1\)](#), [fgrep\(1\)](#), [sed\(1\)](#), [sh\(1\)](#), [attributes\(5\)](#), [environ\(5\)](#), [largefile\(5\)](#), [regex\(5\)](#), [regexp\(5\)](#), [XPG4\(5\)](#)

ПРИМЕЧАНИЯ

`/usr/bin/grep`

Строки ограничены только размером доступной виртуальной памяти. Если обрабатывается строка со встроенными нулевыми символами, **grep** будет вести поиск только до первого такого символа; если эта часть строки соответствует образцу, будет выдана вся строка.

`/usr/xpg4/bin/grep`

Если файл содержит строки длиннее **LINE_MAX** байтов или двоичные данные, результаты работы непересказуемы. Значение **LINE_MAX** определено в файле **`/usr/include/limits.h`**.

Copyright 2002 В.Кравчук, [OpenXS Initiative](#), перевод на русский язык