



Бесплатная электронная книга

# УЧУСЬ sed

Free unaffiliated eBook created from  
**Stack Overflow contributors.**

#sed

.....	1
<b>1: sed</b> .....	<b>2</b>
.....	2
.....	2
.....	2
Examples.....	2
, .....	2
<b>2: BSD / macOS Sed GNU Sed POSIX Sed</b> .....	<b>4</b>
.....	4
.....	4
Examples.....	9
.....	9
'a'.....	10
<b>3:</b> .....	<b>11</b>
.....	11
Examples.....	11
.....	11
<b>4:</b> .....	<b>12</b>
.....	12
Examples.....	12
.....	12
.....	12
, .....	13
, .....	14
.....	15
<b>5:</b> .....	<b>17</b>
.....	17
.....	17
Examples.....	17
/ .....	17
'l' .....	18

<b>6:</b>	.....	<b>20</b>
Examples	.....	20
.....	.....	20
<b>7:</b>	.....	<b>21</b>
Examples	.....	21
.....	.....	21
.....	.....	21
.....	.....	22
- .....	.....	22
<b>8: sed</b>	.....	<b>24</b>
Examples	.....	24
- eXchange	.....	24
<b>9:</b>	.....	<b>25</b>
Examples	.....	25
.....	.....	25
<b>10:</b>	.....	<b>26</b>
.....	.....	26
.....	.....	26
.....	.....	26
<b>, ed</b>	.....	<b>26</b>
Examples	.....	27
.....	.....	27
.....	.....	27
.....	.....	28
.....	.....	28
<b>11:</b>	.....	<b>30</b>
Examples	.....	30
, .....	.....	30
.....	.....	<b>31</b>

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [sed](#)

It is an unofficial and free sed ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official sed.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to [info@zzzprojects.com](mailto:info@zzzprojects.com)

# глава 1: Начало работы с sed

## замечания

## Рекомендации

- [FreeBSD sed man-страница](#)
- [Рукописная страница NetBSD sed](#)
- [Открытая панель управления OpenBSD](#)
- [Illumos sed man-page](#)
- [macOS \(OS X\) man-страница](#)
- [Plan9 sed man-страница](#)
- [Руководство пользователя GNU sed](#)

## Версии

название	Первый выпуск	Версия	Дата выхода
POSIX sed	1992	<a href="#">IEEE Std 1003.1, выпуск 2013</a>	<a href="#">2013-04-19</a>
BSD sed	1992	FreeBSD 10.3 / NetBSD 7.0 / OpenBSD 5.9	2016-04-04
GNU sed	1989	4.2.2	2012-12-22

## Examples

### Привет, мир

Одним из наиболее распространенных способов использования Sed является замена текста, которая может быть достигнута с помощью команды `s`.

В терминале введите `echo "Hello sed" | sed 's/sed/World/'` и нажмите Enter :

```
$ echo "Hello sed" | sed 's/sed/World/'
Hello World
```

«Hello World» должен быть выведен на терминал.

Строка «Hello, sed» отправляется по каналу в качестве входа в команду sed, которая

заменяет слово `sed` на `World` .

Синтаксис базовой команды подстановки - это `s` за которой следует строка или шаблон для поиска и текст подстановки. `s` команда и строки разделены разделителем по умолчанию `/` разделителем.

Прочитайте Начало работы с sed онлайн: <https://riptutorial.com/ru/sed/topic/934/начало-работы-с-sed>

# глава 2: BSD / macOS Sed против GNU Sed против спецификации POSIX Sed

## Вступление

Чтобы процитировать сообщение с @ SnoringFrog по созданию темы:

«Одна из самых больших ошибок при использовании sed - это сценарии, которые не работают (или преуспевают неожиданным образом), потому что они были написаны для одного, а не для другого. Простое сокращение более значительных различий было бы хорошим».

## замечания

*macOS* использует версию *BSD sed* <sup>[1]</sup>, которая во многом отличается от версии *GNU sed*, поставляемой с дистрибутивами *Linux*.

Их **общим знаменателем** является функциональность, установленная **POSIX**: см. [sed POSIX](#).

**Самый портативный подход - использовать только функции POSIX**, но, тем не менее, **ограничивает функциональность**:

- В частности, POSIX определяет **поддержку только для базовых регулярных выражений**, которые имеют много ограничений (например, отсутствие поддержки `|` (чередование) вообще, без прямой поддержки `+` и `?`) И различных требования вылетающих.
  - Оговорка: **GNU sed (без `-r`)**, **не поддерживает `\|`, `\+` и `\?`**, который **НЕ соответствует POSIX; используйте `--posix` для отключения** (см. ниже).
- **Использовать только функции POSIX**:
  - (обе версии): используйте *только* параметры `-n` и `-e` (в частности, не используйте `-E` или `-r` чтобы включить поддержку *расширенных* регулярных выражений)
  - **GNU sed**: добавьте опцию `--posix` для обеспечения только функции POSIX (вам это не нужно, но без нее вы можете случайно использовать функции, отличные от POSIX, не заметив, *caveat*: `--posix` сам по себе не совместим с POSIX)
  - Использование только функций POSIX означает более жесткие требования к форматированию (из-за многих удобств, доступных в **GNU sed**):

- Такие последовательности символов управления, как `\n` и `\t` , обычно НЕ поддерживаются.
- Команды ярлыков и ветвлений (например, `b` ) *должны* сопровождаться *фактической* новой строкой или продолжением через отдельный параметр `-e` .
- Подробнее см. Ниже.

Однако в *обеих* версиях реализованы *расширения* для стандарта POSIX:

- *какие* расширения, которые они реализуют, различаются (GNU `sed` реализует больше).
- даже те расширения, которые они *оба* реализуют, *частично отличаются синтаксисом* .

Если вам нужна поддержка платформ BOTH (обсуждение различий):

- Несовместимые функции:
  - Использование **опции** `-i` **без аргумента** (обновление на месте без резервного копирования) несовместимо:
    - BSD `sed` : ДОЛЖНО использовать `-i ''`
    - GNU `sed` : ДОЛЖНО использовать только `-i` (равнозначно: `-i ''` ) - использование `-i ''` НЕ работает.
  - `-i` **разумно включает строку нумерации строк ввода** в GNU `sed` и последние версии BSD `sed` (например, на FreeBSD 10), но не имеет значения **для macOS с 10.12** .  
Обратите внимание, что в отсутствие `-i` *всех* версий число строк *кумулятивно* отображается во входных файлах.
  - Если **последняя строка ввода не имеет завершающей новой строки** (и печатается):
    - BSD `sed` : *всегда добавляет новую строку* на выходе, даже если строка ввода не заканчивается на один.
    - GNU `sed` : *сохраняет статус конечной новой строки* , т. Е. Добавляет новую строку только в том случае, если строка ввода заканчивается на одном.
- Общие характеристики:
  - Если вы ограничите свои сценарии `sed` тем, что поддерживает BSD `sed` , они, как правило, будут работать и в GNU `sed` - с заметным исключением использования *расширенных* функций регулярного выражения, связанных с платформой, с `-E` . Очевидно, вы также откажетесь от расширений, характерных для версии GNU. См. Следующий раздел.



## Рекомендации по кросс-платформенной поддержке (OS X / BSD, Linux), обусловленные более строгими требованиями версии BSD :

Обратите внимание, что сокращенные *macOS* и *Linux* иногда используются ниже, чтобы сослаться на версии *sed* и BSD соответственно, потому что они являются версиями *акций* на каждой платформе. Тем не менее, можно установить GNU *sed* на macOS, например, используя [Homebrew](#) `cbrew install gnu-sed`.

**Примечание . За исключением случаев использования флагов `-r` и `-E` (расширенные регулярные выражения), приведенные ниже инструкции *относятся* к написанию сценариев *sed* совместимых с POSIX .**

- Для соответствия POSIX вы должны ограничить себя [POSIX BREs \( базовые регулярные выражения \)](#) , которые, к сожалению, как следует из названия, являются довольно базовыми.

**Предостережение :** не предполагайте, что `\|` , `\+` и `\?` поддерживаются: хотя GNU *sed* поддерживает их (если используется `--posix` ), BSD *sed* не выполняет - эти функции *не* совместимы с POSIX.

В то время как `\+` и `\?` **могут быть эмулированы** в соответствии с POSIX:

`\{1,\}` для `\+` ,

`\{0,1\}` для `\?` ,

`\|` (чередование) **не может**, к сожалению.

- Для более мощных регулярных выражений **используйте `-E`** (а не `-r` ) для поддержки **ERE ( расширенные регулярные выражения )** (GNU *sed* не документирует `-E` , но он работает там как псевдоним `-r` ; *более новая* версия BSD *sed* , например, на FreeBSD 10, теперь также поддерживают `-r` , но версия MacOS от 10.12 *не* делает).

**Предостережение .** Несмотря на то, что использование `-r` / `-E` означает, что ваша команда по определению *не* совместима с POSIX, вы все равно должны **ограничивать себя POSIX ERE (расширенные регулярные выражения)** . К сожалению, это означает, что вы не сможете использовать несколько полезных конструкций, в частности:

- потому что они *специфичны для платформы* (например, `\<` на Linux, `[[:<]]` на OS X).
- обратные ссылки *внутри регулярных выражений* (в отличие от «обратных ссылок» на совпадения записей в заменяющей строке вызовов функций `s` ), поскольку BSD *sed* не поддерживает их в *расширенных* регулярных выражениях (но, что любопытно, делает это в *базовые* , где они предназначены для POSIX).

- **Управляющие символы, такие как `\n` и `\t` :**

- В **регулярных выражениях** (как в шаблонах для выбора линии, так и в первом аргументе функции `s` ) предположим, что только `\n` распознается как escape-последовательность (редко используется, поскольку пространство шаблонов обычно представляет собой *одну* строку (без завершения `\n` ) но не внутри

символьного класса , так что, например, `[\n]` не работает; (если ваш вход не содержит управляющих символов, кроме `\t` , вы можете эмулировать `[\n]` с помощью `[:print][:blank:]` , в противном случае - символы управления соединением в виде *литералов* <sup>[2]</sup> ) - **обычно включают контрольные символы в виде литералов , либо с помощью связанных строк в ANSI C (например, `'\t'` ) в оболочек, поддерживающих его ( `bash`, `ksh`, `zsh` ), или с помощью подстановок команд с помощью `printf` (например, `"$(printf '\t')"` ) .**

- Только Linux:

```
sed 's/\t/-/' <<<${a\tb} # -> 'a-b'
```

- OSX и Linux:

```
sed 's/'$'\t'/'-/' <<<${a\tb} # ANSI C-quoted string
```

```
sed 's/'"$$(printf '\t')"'/'-/' <<<${a\tb} # command subst. with printf
```

- В строках замещения, используемых с командой `s` , предположим, что **NO управляющих последовательностей escape-символов поддерживается** , так что опять же включите контрольные символы. как *литералы* , как указано выше.

- Только Linux:

```
sed 's/-/\t/' <<<${ab} # -> 'a<tab>b'
```

- macOS и Linux:

```
sed 's/-/'$'\t'/' <<<'a-b'
```

```
sed 's/-/'"$$(printf '\t')"'/' <<<'a-b'
```

- То же самое для **текстовых аргументов `i` и `a` функции: не использовать последовательности управления символьные** - смотри ниже.

- Ярлыки и ветвление** : метки, а также *аргумент* имени метки для функций `b` и `t` должны сопровождаться **либо *литеральной* новой линией, либо сплайсинговым в `'\n'` .** Кроме того, используйте несколько опций `-e` и завершайте их сразу после имени метки.

- Только Linux:

```
sed -n '/a/ bLBL; d; :LBL p' <<<${a\nb} # -> 'a'
```

- macOS и Linux:

- EITHER (фактические строки новой строки):

```
sed -n '/a/ bLBL d; :LBL p' <<<${a\nb}
```

- ИЛИ (вложенные экземпляры `$(\n)`):

```
sed -n '/a/ bLBL'$'\n''d; :LBL'$'\n''p' <<<${a\nb}
```

- ИЛИ (несколько `-e` опций):

```
sed -n -e '/a/ bLBL' -e 'd; :LBL' -e 'p' <<<${a\nb}
```

- Функции `i` и `a` для вставки / добавления текста** : **следуйте за именем функции по `\` , за которым следует либо *буквальная* новая строка, либо сплайсинг в `'\n'` прежде чем указывать текстовый аргумент.**

- Только Linux:

```
sed '1 i new first line' <<<${a\nb} # -> 'new first line<nl>a<nl>b'
```

- OSX и Linux:

```
sed -e '1 i\'$'\n''new first line' <<< '$a\nb'
```

- **Замечания:**

- Без `-e` текстовый аргумент необъяснимо не завершен с использованием новой строки при выводе на macOS (ошибка?).
- **Не используйте escape-символы**, такие как `\n` и `\t` в текстовом аргументе, поскольку они поддерживаются только в Linux.
- Если текстовый аргумент имеет фактические внутренние символы новой строки, `\` -escape их.
- Если вы хотите поместить дополнительные команды после текстового аргумента, вы должны закончить его с помощью (неэкспертированной) новой строки (будь то литеральный или сращиваемый) или продолжить с отдельной опцией `-e` (это общее требование, которое применяется ко всем версиям) ,

- Внутри **списков функций** (несколько вызовов функций, заключенных в `{...}` ), **обязательно завершайте последнюю функцию перед закрытием** `;` ,

- Только Linux:

- `sed -n '1 {p;q}' <<< '$a\nb' # -> 'a'`

- macOS и Linux:

- `sed -n '1 {p;q;} ' <<< '$a\nb'`

---

## GNU `sed` -специальные функции, отсутствующие в BSD `sed` :

Возможности GNU вы пропустите, если вам нужно поддерживать обе платформы:

- Различные **варианты соответствия и замены регулярных выражений** (как в шаблонах для выбора линии, так и в первом аргументе функции `s` ):

- Параметр `i` для сопоставления регулярных выражений **case-INsensitive** ( **невероятно, BSD `sed` не поддерживает это вообще** ).
- Параметр `m` для многострочного совпадения (где `^` / `$` соответствует началу / концу *каждой строки* )
- Дополнительные параметры, относящиеся к функции `s` , см. В [https://www.gnu.org/software/sed/manual/sed.html#The-\\_0022s\\_0022-Command](https://www.gnu.org/software/sed/manual/sed.html#The-_0022s_0022-Command)

- **Эквивалентные последовательности**

- Вспомогательные escape-последовательности, такие как `\u` в аргументе замены функции `s///` которые допускают *манипуляции подстрокой* , в пределах; например, `sed 's/^\./\u&/' <<< 'dog' # -> 'Dog' - см.` [http://www.gnu.org/software/sed/manual/sed.html#The-\\_0022s\\_0022-command](http://www.gnu.org/software/sed/manual/sed.html#The-_0022s_0022-command)
- Управляющие последовательности escape-последовательностей: в дополнение к `\n` , `\t` , ..., escape-пунктам на основе кода; например, все следующие escape-последовательности (шестнадцатеричные, восьмеричные, десятичные)

представляют одну цитату ( ' ): \x27 , \o047 , \d039 - см.

<https://www.gnu.org/software/sed/manual/sed.html#Escapes>

- **Расширения адресов** , такие как `first~step` чтобы соответствовать каждой шаг-й строке, `addr, +N` чтобы соответствовать N строкам после `addr` , ... - см.

[Http://www.gnu.org/software/sed/manual/sed.html#Addresses](http://www.gnu.org/software/sed/manual/sed.html#Addresses)

---

[1] Версия macOS `sed` *старше* версии на других BSD-подобных системах, таких как FreeBSD и PC-BSD. К сожалению, это означает, что вы не можете предположить, что функции, которые работают в FreeBSD, например, будут работать [одинаково] на macOS.

[2] Строка с цитированием ANSI C

`$'\001\002\003\004\005\006\007\010\011\013\014\015\016\017\020\021\022\023\024\025\026\027\030\031\032\033\034\035\036\037\040\041\042\043\044\045\046\047\050\051\052\053\054\055\056\057\060\061\062\063\064\065\066\067\070\071\072\073\074\075\076\077\100\101\102\103\104\105\106\107\110\111\113\114\115\116\117\120\121\122\123\124\125\126\127\130\131\132\133\134\135\136\137\140\141\142\143\144\145\146\147\150\151\152\153\154\155\156\157\160\161\162\163\164\165\166\167\170\171\172\173\174\175\176\177\200\201\202\203\204\205\206\207\210\211\213\214\215\216\217\220\221\222\223\224\225\226\227\230\231\232\233\234\235\236\237\240\241\242\243\244\245\246\247\250\251\252\253\254\255\256\257\260\261\262\263\264\265\266\267\270\271\272\273\274\275\276\277\300\301\302\303\304\305\306\307\310\311\313\314\315\316\317\320\321\322\323\324\325\326\327\330\331\332\333\334\335\336\337\340\341\342\343\344\345\346\347\350\351\352\353\354\355\356\357\360\361\362\363\364\365\366\367\370\371\372\373\374\375\376\377\400\401\402\403\404\405\406\407\410\411\413\414\415\416\417\420\421\422\423\424\425\426\427\430\431\432\433\434\435\436\437\440\441\442\443\444\445\446\447\450\451\452\453\454\455\456\457\460\461\462\463\464\465\466\467\470\471\472\473\474\475\476\477\500\501\502\503\504\505\506\507\510\511\513\514\515\516\517\520\521\522\523\524\525\526\527\530\531\532\533\534\535\536\537\540\541\542\543\544\545\546\547\550\551\552\553\554\555\556\557\560\561\562\563\564\565\566\567\570\571\572\573\574\575\576\577\600\601\602\603\604\605\606\607\610\611\613\614\615\616\617\620\621\622\623\624\625\626\627\630\631\632\633\634\635\636\637\640\641\642\643\644\645\646\647\650\651\652\653\654\655\656\657\660\661\662\663\664\665\666\667\670\671\672\673\674\675\676\677\700\701\702\703\704\705\706\707\710\711\713\714\715\716\717\720\721\722\723\724\725\726\727\730\731\732\733\734\735\736\737\740\741\742\743\744\745\746\747\750\751\752\753\754\755\756\757\760\761\762\763\764\765\766\767\770\771\772\773\774\775\776\777\800\801\802\803\804\805\806\807\810\811\813\814\815\816\817\820\821\822\823\824\825\826\827\830\831\832\833\834\835\836\837\840\841\842\843\844\845\846\847\850\851\852\853\854\855\856\857\860\861\862\863\864\865\866\867\870\871\872\873\874\875\876\877\900\901\902\903\904\905\906\907\910\911\913\914\915\916\917\920\921\922\923\924\925\926\927\930\931\932\933\934\935\936\937\940\941\942\943\944\945\946\947\950\951\952\953\954\955\956\957\960\961\962\963\964\965\966\967\970\971\972\973\974\975\976\977\1000\1001\1002\1003\1004\1005\1006\1007\1010\1011\1013\1014\1015\1016\1017\1020\1021\1022\1023\1024\1025\1026\1027\1030\1031\1032\1033\1034\1035\1036\1037\1040\1041\1042\1043\1044\1045\1046\1047\1050\1051\1052\1053\1054\1055\1056\1057\1060\1061\1062\1063\1064\1065\1066\1067\1070\1071\1072\1073\1074\1075\1076\1077\1100\1101\1102\1103\1104\1105\1106\1107\1110\1111\1113\1114\1115\1116\1117\1120\1121\1122\1123\1124\1125\1126\1127\1130\1131\1132\1133\1134\1135\1136\1137\1140\1141\1142\1143\1144\1145\1146\1147\1150\1151\1152\1153\1154\1155\1156\1157\1160\1161\1162\1163\1164\1165\1166\1167\1170\1171\1172\1173\1174\1175\1176\1177\1200\1201\1202\1203\1204\1205\1206\1207\1210\1211\1213\1214\1215\1216\1217\1220\1221\1222\1223\1224\1225\1226\1227\1230\1231\1232\1233\1234\1235\1236\1237\1240\1241\1242\1243\1244\1245\1246\1247\1250\1251\1252\1253\1254\1255\1256\1257\1260\1261\1262\1263\1264\1265\1266\1267\1270\1271\1272\1273\1274\1275\1276\1277\1300\1301\1302\1303\1304\1305\1306\1307\1310\1311\1313\1314\1315\1316\1317\1320\1321\1322\1323\1324\1325\1326\1327\1330\1331\1332\1333\1334\1335\1336\1337\1340\1341\1342\1343\1344\1345\1346\1347\1350\1351\1352\1353\1354\1355\1356\1357\1360\1361\1362\1363\1364\1365\1366\1367\1370\1371\1372\1373\1374\1375\1376\1377\1400\1401\1402\1403\1404\1405\1406\1407\1410\1411\1413\1414\1415\1416\1417\1420\1421\1422\1423\1424\1425\1426\1427\1430\1431\1432\1433\1434\1435\1436\1437\1440\1441\1442\1443\1444\1445\1446\1447\1450\1451\1452\1453\1454\1455\1456\1457\1460\1461\1462\1463\1464\1465\1466\1467\1470\1471\1472\1473\1474\1475\1476\1477\1500\1501\1502\1503\1504\1505\1506\1507\1510\1511\1513\1514\1515\1516\1517\1520\1521\1522\1523\1524\1525\1526\1527\1530\1531\1532\1533\1534\1535\1536\1537\1540\1541\1542\1543\1544\1545\1546\1547\1550\1551\1552\1553\1554\1555\1556\1557\1560\1561\1562\1563\1564\1565\1566\1567\1570\1571\1572\1573\1574\1575\1576\1577\1600\1601\1602\1603\1604\1605\1606\1607\1610\1611\1613\1614\1615\1616\1617\1620\1621\1622\1623\1624\1625\1626\1627\1630\1631\1632\1633\1634\1635\1636\1637\1640\1641\1642\1643\1644\1645\1646\1647\1650\1651\1652\1653\1654\1655\1656\1657\1660\1661\1662\1663\1664\1665\1666\1667\1670\1671\1672\1673\1674\1675\1676\1677\1700\1701\1702\1703\1704\1705\1706\1707\1710\1711\1713\1714\1715\1716\1717\1720\1721\1722\1723\1724\1725\1726\1727\1730\1731\1732\1733\1734\1735\1736\1737\1740\1741\1742\1743\1744\1745\1746\1747\1750\1751\1752\1753\1754\1755\1756\1757\1760\1761\1762\1763\1764\1765\1766\1767\1770\1771\1772\1773\1774\1775\1776\1777\1800\1801\1802\1803\1804\1805\1806\1807\1810\1811\1813\1814\1815\1816\1817\1820\1821\1822\1823\1824\1825\1826\1827\1830\1831\1832\1833\1834\1835\1836\1837\1840\1841\1842\1843\1844\1845\1846\1847\1850\1851\1852\1853\1854\1855\1856\1857\1860\1861\1862\1863\1864\1865\1866\1867\1870\1871\1872\1873\1874\1875\1876\1877\1900\1901\1902\1903\1904\1905\1906\1907\1910\1911\1913\1914\1915\1916\1917\1920\1921\1922\1923\1924\1925\1926\1927\1930\1931\1932\1933\1934\1935\1936\1937\1940\1941\1942\1943\1944\1945\1946\1947\1950\1951\1952\1953\1954\1955\1956\1957\1960\1961\1962\1963\1964\1965\1966\1967\1970\1971\1972\1973\1974\1975\1976\1977\2000\2001\2002\2003\2004\2005\2006\2007\2010\2011\2013\2014\2015\2016\2017\2020\2021\2022\2023\2024\2025\2026\2027\2030\2031\2032\2033\2034\2035\2036\2037\2040\2041\2042\2043\2044\2045\2046\2047\2050\2051\2052\2053\2054\2055\2056\2057\2060\2061\2062\2063\2064\2065\2066\2067\2070\2071\2072\2073\2074\2075\2076\2077\2100\2101\2102\2103\2104\2105\2106\2107\2110\2111\2113\2114\2115\2116\2117\2120\2121\2122\2123\2124\2125\2126\2127\2130\2131\2132\2133\2134\2135\2136\2137\2140\2141\2142\2143\2144\2145\2146\2147\2150\2151\2152\2153\2154\2155\2156\2157\2160\2161\2162\2163\2164\2165\2166\2167\2170\2171\2172\2173\2174\2175\2176\2177\2200\2201\2202\2203\2204\2205\2206\2207\2210\2211\2213\2214\2215\2216\2217\2220\2221\2222\2223\2224\2225\2226\2227\2230\2231\2232\2233\2234\2235\2236\2237\2240\2241\2242\2243\2244\2245\2246\2247\2250\2251\2252\2253\2254\2255\2256\2257\2260\2261\2262\2263\2264\2265\2266\2267\2270\2271\2272\2273\2274\2275\2276\2277\2300\2301\2302\2303\2304\2305\2306\2307\2310\2311\2313\2314\2315\2316\2317\2320\2321\2322\2323\2324\2325\2326\2327\2330\2331\2332\2333\2334\2335\2336\2337\2340\2341\2342\2343\2344\2345\2346\2347\2350\2351\2352\2353\2354\2355\2356\2357\2360\2361\2362\2363\2364\2365\2366\2367\2370\2371\2372\2373\2374\2375\2376\2377\2400\2401\2402\2403\2404\2405\2406\2407\2410\2411\2413\2414\2415\2416\2417\2420\2421\2422\2423\2424\2425\2426\2427\2430\2431\2432\2433\2434\2435\2436\2437\2440\2441\2442\2443\2444\2445\2446\2447\2450\2451\2452\2453\2454\2455\2456\2457\2460\2461\2462\2463\2464\2465\2466\2467\2470\2471\2472\2473\2474\2475\2476\2477\2500\2501\2502\2503\2504\2505\2506\2507\2510\2511\2513\2514\2515\2516\2517\2520\2521\2522\2523\2524\2525\2526\2527\2530\2531\2532\2533\2534\2535\2536\2537\2540\2541\2542\2543\2544\2545\2546\2547\2550\2551\2552\2553\2554\2555\2556\2557\2560\2561\2562\2563\2564\2565\2566\2567\2570\2571\2572\2573\2574\2575\2576\2577\2600\2601\2602\2603\2604\2605\2606\2607\2610\2611\2613\2614\2615\2616\2617\2620\2621\2622\2623\2624\2625\2626\2627\2630\2631\2632\2633\2634\2635\2636\2637\2640\2641\2642\2643\2644\2645\2646\2647\2650\2651\2652\2653\2654\2655\2656\2657\2660\2661\2662\2663\2664\2665\2666\2667\2670\2671\2672\2673\2674\2675\2676\2677\2700\2701\2702\2703\2704\2705\2706\2707\2710\2711\2713\2714\2715\2716\2717\2720\2721\2722\2723\2724\2725\2726\2727\2730\2731\2732\2733\2734\2735\2736\2737\2740\2741\2742\2743\2744\2745\2746\2747\2750\2751\2752\2753\2754\2755\2756\2757\2760\2761\2762\2763\2764\2765\2766\2767\2770\2771\2772\2773\2774\2775\2776\2777\2800\2801\2802\2803\2804\2805\2806\2807\2810\2811\2813\2814\2815\2816\2817\2820\2821\2822\2823\2824\2825\2826\2827\2830\2831\2832\2833\2834\2835\2836\2837\2840\2841\2842\2843\2844\2845\2846\2847\2850\2851\2852\2853\2854\2855\2856\2857\2860\2861\2862\2863\2864\2865\2866\2867\2870\2871\2872\2873\2874\2875\2876\2877\2900\2901\2902\2903\2904\2905\2906\2907\2910\2911\2913\2914\2915\2916\2917\2920\2921\2922\2923\2924\2925\2926\2927\2930\2931\2932\2933\2934\2935\2936\2937\2940\2941\2942\2943\2944\2945\2946\2947\2950\2951\2952\2953\2954\2955\2956\2957\2960\2961\2962\2963\2964\2965\2966\2967\2970\2971\2972\2973\2974\2975\2976\2977\3000\3001\3002\3003\3004\3005\3006\3007\3010\3011\3013\3014\3015\3016\3017\3020\3021\3022\3023\3024\3025\3026\3027\3030\3031\3032\3033\3034\3035\3036\3037\3040\3041\3042\3043\3044\3045\3046\3047\3050\3051\3052\3053\3054\3055\3056\3057\3060\3061\3062\3063\3064\3065\3066\3067\3070\3071\3072\3073\3074\3075\3076\3077\3100\3101\3102\3103\3104\3105\3106\3107\3110\3111\3113\3114\3115\3116\3117\3120\3121\3122\3123\3124\3125\3126\3127\3130\3131\3132\3133\3134\3135\3136\3137\3140\3141\3142\3143\3144\3145\3146\3147\3150\3151\3152\3153\3154\3155\3156\3157\3160\3161\3162\3163\3164\3165\3166\3167\3170\3171\3172\3173\3174\3175\3176\3177\3200\3201\3202\3203\3204\3205\3206\3207\3210\3211\3213\3214\3215\3216\3217\3220\3221\3222\3223\3224\3225\3226\3227\3230\3231\3232\3233\3234\3235\3236\3237\3240\3241\3242\3243\3244\3245\3246\3247\3250\3251\3252\3253\3254\3255\3256\3257\3260\3261\3262\3263\3264\3265\3266\3267\3270\3271\3272\3273\3274\3275\3276\3277\3300\3301\3302\3303\3304\3305\3306\3307\3310\3311\3313\3314\3315\3316\3317\3320\3321\3322\3323\3324\3325\3326\3327\3330\3331\3332\3333\3334\3335\3336\3337\3340\3341\3342\3343\3344\3345\3346\3347\3350\3351\3352\3353\3354\3355\3356\3357\3360\3361\3362\3363\3364\3365\3366\3367\3370\3371\3372\3373\3374\3375\3376\3377\3400\3401\3402\3403\3404\3405\3406\3407\3410\3411\3413\3414\3415\3416\3417\3420\3421\3422\3423\3424\3425\3426\3427\3430\3431\3432\3433\3434\3435\3436\3437\3440\3441\3442\3443\3444\3445\3446\3447\3450\3451\3452\3453\3454\3455\3456\3457\3460\3461\3462\3463\3464\3465\3466\3467\3470\3471\3472\3473\3474\3475\3476\3477\3500\3501\3502\3503\3504\3505\3506\3507\3510\3511\3513\3514\3515\3516\3517\3520\3521\3522\3523\3524\3525\3526\3527\3530\3531\3532\3533\3534\3535\3536\3537\3540\3541\3542\3543\3544\3545\3546\3547\3550\3551\3552\3553\3554\3555\3556\3557\3560\3561\3562\3563\3564\3565\3566\3567\3570\3571\3572\3573\3574\3575\3576\3577\3600\3601\3602\3603\3604\3605\3606\3607\3610\3611\3613\3614\3615\3616\3617\3620\3621\3622\3623\3624\3625\3626\3627\3630\3631\3632\3633\3634\3635\3636\3637\3640\3641\3642\3643\3644\3645\3646\3647\3650\3651\3652\3653\3654\3655\3656\3657\3660\3661\3662\3663\3664\3665\3666\3667\3670\3671\3672\3673\3674\3675\3676\3677\3700\3701\3702\3703\3704\3705\3706\3707\3710\3711\3713\3714\3715\3716\3717\3720\3721\3722\3723\3724\3725\3726\3727\3730\3731\3732\3733\3734\3735\3736\3737\3740\3741\3742\3743\3744\3745\3746\3747\3750\3751\3752\3753\3754\3755\3756\3757\3760\3761\3762\3763\3764\3765\3766\3767\3770\3771\3772\3773\3774\3775\3776\3777\3800\3801\3802\3803\3804\3805\3806\3807\3810\3811\3813\3814\3815\3816\3817\3820\3821\3822\3823\3824\3825\3826\3827\3830\3831\3832\3833\3834\3835\3836\3837\3840\3841\3842\3843\3844\3845\3846\3847\3850\3851\3852\3853\3854\3855\3856\3857\3860\3861\3862\3863\3864\3865\3866\3867\3870\3871\3872\3873\3874\3875\3876\3877\3900\3901\3902\3903\3904\3905\3906\3907\3910\3911\3913\3914\3915\3916\3917\3920\3921\3922\3923\3924\3925\3926\3927\3930\3931\3932\3933\3934\3935\3936\3937\3940\3941\3942\3943\3944\3945\3946\3947\3950\3951\3952\3953\3954\3955\3956\3957\3960\3961\3962\3963\3964\396`

## Добавить литерал в строку с функцией 'a'

Примечание. Для краткости команды используют [здесь строки \( <<< \)](#) и строки с котировкой ANSI C ( `$'...'` ). Обе эти функции оболочки работают в `bash` , `ksh` и `zsh` .

```
# GNU Sed
$ sed '1 a appended text' <<<'line 1'
line 1
appended text

# BSD Sed (multi-line form)
sed '1 a\
appended text' <<<'line 1'

# BSD Sed (single-line form via a Bash/Ksh/Zsh ANSI C-quoted string)
sed $'1 a\\nappended text' <<<'line 1'
```

Обратите внимание, что для BSD Seed требуется `\` последующей *фактической новой строкой* для добавления добавляемого текста.

То же самое относится к связанным функциям `i` (insert) и `c` (delete and insert).

Прочитайте [BSD / macOS Sed против GNU Sed против спецификации POSIX Sed онлайн: <https://riptutorial.com/ru/sed/topic/9436/bsd---macos-sed-против-gnu-sed-против-спецификации-posix-sed>](https://riptutorial.com/ru/sed/topic/9436/bsd---macos-sed-против-gnu-sed-против-спецификации-posix-sed)

---

# глава 3: Ветвление

## Вступление

Операция ветвления `sed` может помочь контролировать поток программы.

## Examples

### Повторить несколько регулярных выражений с помощью безусловной ветви

Предположим, что у меня есть файл с именем `in.txt` :

```
$ cat in.txt
a
b
a
c
a
d
```

Я хочу заменить `a\nc deleted` , но не `a\nb` или `a\nd` .

```
$ sed -e ':loop          # create a branch/label named `loop`
${
N                        # append the next line of input into the pattern space
/\n$/!b loop           # If it is not the last line go to the `loop` branch again
}
s/a\nc/"deleted"/' in.txt # do replacing in the pattern space

a
b
"deleted"    # see! succeed
a
d
```

Прочитайте Ветвление онлайн: <https://riptutorial.com/ru/sed/topic/8821/ветвление>

---

# глава 4: Диапазон адресов и адресов

## Вступление

Команды Sed могут быть указаны, чтобы действовать только на определенных линиях, используя *адреса* или *диапазоны адресов*.

## Examples

### Конкретная линия

```
$ cat ip.txt
address
range
substitution
pattern
sample
```

- N-я строка

```
$ sed -n '2p' ip.txt
range

$ sed '3d' ip.txt
address
range
pattern
sample
```

- Последняя линия

```
$ sed -n '$p' ip.txt
sample
```

### Конкретный диапазон линий

```
$ cat ip.txt
address
range
substitution
pattern
sample
```

- Указанный диапазон включает номера строк

```
$ sed -n '2,4p' ip.txt
range
```

```
substitution
pattern
```

- `$` может использоваться для указания последней строки. Пространство может использоваться между адресом и командой для ясности

```
$ sed -n '3,$ s/[aeiou]//gp' ip.txt
sbsttttn
pttrn
smpl
```

## GNU sed

- $i$ -я строка на  $i + j$ -я строка

```
$ sed '2,+2d' ip.txt
address
sample
```

- $i$ -я строка и  $i + j$ ,  $i + 2j$ ,  $i + 3j$  и т. д.

```
$ sed -n '1~2p' ip.txt
address
substitution
sample
```

## Строки, соответствующие шаблону регулярного выражения

```
$ cat ip.txt
address
range
substitution
pattern
sample
Add Sub Mul Div
```

- Линии, соответствующие шаблону

```
$ sed '/add/d' ip.txt
range
substitution
pattern
sample
Add Sub Mul Div

$ sed -n '/t/p' ip.txt
substitution
pattern

$ sed -n '/[A-Z]/ s| ||gp' ip.txt
Add/Sub/Mul/Div
```



- Диапазон шаблонов

```
$ sed -n '/add/,/sub/p' ip.txt
address
range
substitution

$ sed -n '/a/,/e/p' ip.txt
address
range
pattern
sample
```

## Заметка

- Во втором примере он соответствовал двум диапазонам - линиям 1,2 и линиям 4,5
- См. [Использование разных разделителей](#) о том, как использовать другие символы вместо / для указания шаблона

## GNU sed

- Нечувствительность к регистру

```
$ sed -n '/add/Ip' ip.txt
address
Add Sub Mul Div

$ sed -n '/add/I,/sub/p' ip.txt
address
range
substitution
Add Sub Mul Div
```

## Определение диапазона с использованием как числа, так и рисунка

```
$ cat ip.txt
address
range
substitution
pattern
sample
Add Sub Mul Div
```

- Номер строки для сопоставления строк

```
$ sed -n '2,/pat/p' ip.txt
range
substitution
pattern
```

- **Линейный шаблон совпадения с номером строки**

```
$ sed '/pat/, $d' ip.txt
address
range
substitution
```

## GNU sed

- **Соответствующий шаблон линии плюс количество строк после него**

```
$ sed -n '/add/I,+1p' ip.txt
address
range
Add Sub Mul Div
```

- **0 может использоваться в качестве номера стартовой линии для окончания конца диапазона, когда шаблон соответствует первой строке ввода**

```
$ sed -n '0,/r/p' ip.txt
address

$ sed -n '1,/r/p' ip.txt
address
range

$ sed -n '0,/u/p' ip.txt
address
range
substitution
```

## Отрицательный диапазон адресов

```
$ cat ip.txt
address
range
substitution
1234
search pattern
sample
Add Sub Mul Div
```

- **Удаление строк, отличных от указанного адреса**

```
$ sed '/[0-9]/!d' ip.txt
1234

$ sed -n '/[0-9]/p' ip.txt
1234

$ sed '$!d' ip.txt
Add Sub Mul Div
```

```
$ sed -n '$p' ip.txt
Add Sub Mul Div
```

- Поиск и замена строк, не соответствующих шаблону

```
$ sed '/ /! s/^/#/' ip.txt
#address
#range
#substitution
#1234
search pattern
#sample
Add Sub Mul Div

$ sed '/add/,/sub/! s/[aeiou]//gi' ip.txt
address
range
substitution
1234
srch pttrn
smpl
dd Sb Ml Dv
```

Прочитайте Диапазон адресов и адресов онлайн: <https://riptutorial.com/ru/sed/topic/3120/диапазон-адресов-и-адресов>

## глава 5: Дополнительные опции

### Синтаксис

- -a - (BSD sed) Создавать / обрезать все файлы, написанные до обработки
- -E | -r - Использовать расширенные регулярные выражения
- -i | -I - См. [Раздел «Редактирование на месте»](#)
- -l - (BSD sed) Использовать вывод с линейной буферизацией
- -l length - (GNU sed) Укажите длину для `1` командной строки
- -s - (GNU sed) Обработка файлов как отдельных потоков
- -u - Не буферизовать вывод
- -z - (GNU sed) Используйте символ NUL для разделения записей
- --quiet | --silent - (GNU sed) Синонимы для `-n`
- --expression = command - (GNU sed) Синоним для `-e`
- --file = command\_file - (GNU sed) Синоним для `-f`
- --follow-symlinks - (GNU sed) Следуйте символическим ссылкам
- --in-place [= extension] - (GNU sed) Синоним для `-i`
- --line-length = length - (GNU sed) Синоним `-l`
- --separate - (GNU sed) Синоним для `-s`
- --unbuffered - (GNU sed) Синоним для `-u`
- --null-data - (GNU sed) Синоним для `-z`
- --help - (GNU sed) Использование печати
- --version - (GNU sed) Версия для печати

### замечания

Опция `-E` должна быть стандартизована в следующей крупной версии, см.

[Соответствующую проблему](#) .

### Examples

#### Задержка создания / усечения файлов

Файлы, написанные с помощью команды `w` , создаются / усекаются перед выполнением любых команд.

```
$ sed 'w created-file' < /dev/null && ls created-file && rm created-file
created-file
```

Из стандарта:

Каждый `wfile` должен быть создан до начала обработки. Реализации должны

BSD<sub>sed</sub> предоставляет возможность `-a` отсрочить создание / усечение файлов до тех пор, пока они не будут записаны с помощью команды `w`.

Прочитайте Дополнительные опции онлайн: <https://riptutorial.com/ru/sed/topic/7922/дополнительные-опции>

---

## глава 6: Команда добавления

### Examples

#### Вставить строку после первого совпадения

Учитывая файл `file.txt` со следующим содержимым:

```
line 1
line 2
line 3
```

Вы можете добавить новую строку после первой соответствующей строки с `a` команды.

Для портативного использования команда должна следовать сразу экранированного перевода строки, с текстом-на-присоединять на своей собственной линии или линий. `a`

```
sed '
/line 2/a\
new line 2.2
' file.txt
```

#### GNU sed

Некоторые версии `sed` позволяют добавлять текст в приложение с `a` команды:

```
sed '/line 2/a new line 2.2' file.txt
```

Вышеприведенные команды выводят:

```
line 1
line 2
new line 2.2
line 3
```

Прочитайте Команда добавления онлайн: <https://riptutorial.com/ru/sed/topic/2835/команда-добавления>

# глава 7: подмена

## Examples

### Замена с использованием переменных оболочки

Переменные внутри одиночные кавычки ' не получает расширено POSIX совместимых оболочек, поэтому с помощью переменной оболочки в sed подстановки требует использования двойных кавычек " вместо одиночных кавычек ' :

```
$ var="he"
$ echo "hello" | sed "s/$var/XX/"
XXllo
```

```
$ var="he"
$ echo "hello" | sed 's/$var/XX/'
hello
```

Будьте осторожны при вводе команд при оценке переменных:

```
$ var='./&/;x;w/etc/passwd
> x;s/he'
$ echo "hello" | sed "s/$var/XX/"
sed: /etc/passwd: Permission denied
```

Если выше было выполнено как root, выход был бы неотличим от первого примера, и содержимое файла /etc/passwd было бы уничтожено.

### обратная ссылка

Используя экранированные скобки, вы можете определить группу захвата в шаблоне, который может быть повторно привязан в строке замещения с помощью \1 :

```
$ echo Hello world! | sed 's/\(Hello\) world!/\1 sed/'
Hello sed
```

С несколькими группами:

```
$ echo one two three | sed 's/\(one\) \(two\) \(three\)/\3 \2 \1/'
three two one
```

### BSD sed GNU sed

При использовании расширенных регулярных выражений (см. [Дополнительные параметры](#)) скобки выполняют группировку по умолчанию и не должны быть экранированы:



```
$ echo one two three | sed -E 's/(one) (two) (three)/\3 \2 \1/'
three two one
```

Слова, состоящие из буквы, цифр и подчеркиваний, могут быть сопоставлены с использованием выражения `[[[:alnum:]]_]\{1,\}` :

```
$ echo Hello 123 reg_exp | sed 's/\([[[:alnum:]]_]\{1,\}\) \([[[:alnum:]]_]\{1,\}\) \([[[:alnum:]]_]\{1,\}\)/\3 \2 \1/'
reg_exp 123 Hello
```

## GNU sed

Последовательность `\w` эквивалентна `[[[:alnum:]]_]`

```
$ echo Hello 123 reg_exp | sed 's/(\w\w*) (\w\w*) (\w\w*)/\3 \2 \1/'
reg_exp 123 Hello
```

## Использование разных разделителей

POSIX / IEEE Open Group Base Specification [говорит](#) :

### **[2addr] s / BRE / замена / флаги**

Подставьте заменяющую строку для экземпляров BRE в пространстве шаблонов.

**Любой символ, отличный от обратной косой черты или новой строки**, может использоваться вместо косой черты, чтобы разграничить BRE и замену. В BRE и замене сам разделитель BRE может использоваться как буквенный символ, если ему предшествует обратная косая черта.

Бывают случаи, когда разделитель `/` для замены `sed` находится в BRE или замене, вызывая ошибки, такие как:

```
$ echo "2/3/4" | sed "s/2/3/X/"
sed: -e expression #1, char 7: unknown option to `s'
```

Для этого мы можем использовать разные разделители, такие как `#` или `_` или даже пробел:

```
$ echo "2/3/4" | sed "s#2/3#X#"
X/4
$ echo "2/3/4" | sed "s_2/3_X_"
X/4
$ echo "2/3/4" | sed "s 2/3 X "
X/4
```

## Флаги шаблонов - замена замещения

Если мы хотим заменить только первое вхождение в строке, мы используем `sed` как обычно:

```
$ cat example
aaaaabbbbb
aaaaaccccc
aaaaaddddd
$ sed 's/a/x/' example
xaaaabbbbb
xaaaaccccc
xaaaaddddd
```

Но что, если мы хотим заменить все вхождения?

Мы просто добавляем флаг `g` pattern в конце:

```
$ sed 's/a/x/g' example
xxxxxbbbbb
xxxxxccccc
xxxxxddddd
```

И если мы хотим заменить одно конкретное событие, мы можем фактически указать, какой из них:

```
$ sed 's/a/x/3' example
aaxaabbbbb
aaxaaccccc
aaxaaddddd
```

`/3` - `/3` -е место.

## GNU sed

Из `info sed` см. [Руководство GNU sed](#) для онлайн-версии

в стандарте POSIX не указывается, что должно произойти, когда вы смешиваете модификаторы `g` и `NUMBER`, и в настоящее время не существует широко согласованного значения в реализациях `sed`. Для GNU `sed` взаимодействие определяется как: игнорировать совпадения до `NUMBER`th, а затем сопоставлять и заменять все совпадения с `NUMBER`th on.

```
$ sed 's/b/y/2g' example
aaaaabyyyy
aaaaaccccc
aaaaaddddd

$ sed 's/c/z/g3' example
aaaaabbbbb
aaaaacczzz
aaaaaddddd
```

Прочитайте подмена онлайн: <https://riptutorial.com/ru/sed/topic/1096/подмена>

---

## глава 8: Расширенные команды sed

### Examples

Вставьте новую строку перед сопоставлением шаблона - с помощью eXchange

Учитывая файл file.txt со следующим содержимым:

```
line 1  
line 2  
line 3
```

Вы можете добавить новую строку, используя следующую команду:

```
sed '/line 2/{x;p;x;}' file.txt
```

Выведенная выше команда выведет

```
line 1  
  
line 2  
line 3
```

#### Объяснение:

Команда `x` - это eXchange. `sed` имеет буфер, который вы можете использовать для хранения некоторых строк. Эта команда обменивает этот буфер с текущей строкой (так что текущая строка переходит в этот буфер, а содержимое буфера становится текущей строкой).

Команда `p` печатает текущую строку.

Прочитайте Расширенные команды sed онлайн: <https://riptutorial.com/ru/sed/topic/3946/расширенные-команды-sed>

# глава 9: Регулярные выражения

## Examples

### Использование разных разделителей

Учитывая такой файл:

```
$ cat file
hello/how/are/you
i am fine
```

Вы можете использовать `/pattern/` для соответствия определенным строкам:

```
$ sed -n '/hello/p' file
hello/how/are/you
```

Если шаблон содержит косые черты, вы можете использовать другой разделитель, используя `\cBREc` :

```
$ sed -n '\#hello/how#p' file
hello/how/are/you
$ sed -n '\_hello/how_p' file
hello/how/are/you
```

Как определено POSIX в:

#### Регулярные выражения в sed

В контексте адреса, конструкция `\cBREc` , где `c` - любой символ, отличный от `\cBREc` слэша, или, должен быть идентичен `/BRE/` . Если символ, обозначенный знаком `c`, отображается после обратного слэша, то он должен считаться буквенным символом, который не должен прерывать BRE. Например, в контексте адрес «`\ xabc \ xdefx`» второй `x` обозначает себя, так что BRE является «`abcxdef`».

Прочитайте Регулярные выражения онлайн: <https://riptutorial.com/ru/sed/topic/7720/регулярные-выражения>

## глава 10: Редактирование на месте

### Синтаксис

- `sed -l extension` - FreeBSD sed (непрерывный линейный счетчик)
- `sed -l [ расширение ]` - NetBSD и Illumos sed (непрерывный линейный счетчик)
- `sed -i extension` - команда FreeBSD
- `sed -i [ расширение ]` - NetBSD, OpenBSD, Illumos, BusyBox и GNU sed
- `sed -i in-place [= extension]` - Illumos, BusyBox и GNU sed

### параметры

параметр	подробности
<code>extension</code>	Сохраните файл резервной копии с указанным расширением или нет файла резервной копии, если <code>extension</code> является строкой нулевой длины.

### замечания

Редактирование на месте - это распространенное, но нестандартное расширение, присутствующее в большинстве последних систем.

Из [руководства BSD sed](#)

(такой раздел появляется во всех текущих руководствах BSD `sed` и их производных)

Не рекомендуется указывать расширение нулевой длины при редактировании файлов, поскольку это может привести к повреждению или частичному контенту в ситуациях, когда дисковое пространство исчерпано и т. Д.

## Не забывайте, что могущественный `ed`

Разумеется, для `sed` и для редактирования на месте используются `sed`, но когда стандарт UNIX расширяется, мы всегда должны спрашивать, почему старый стандарт UNIX не включил эту функцию. Хотя UNIX не совершенен, ортогональность и полнота инструментов были разработаны настолько близко к совершенству, по крайней мере, для целей, которые были видны в 1970 году: *редактирование текста и автоматическое редактирование текста были наверняка видны в это время.*

На самом деле идея `sed` заключается не в редактировании *файла* на месте, а в

редактировании *потока* . Вот почему имя `sed` является коротким *редактором потока* . Уберите `s` , и вы получите инструмент, который был фактически предназначен для редактирования *файлов* : `ed` :

```
printf 'g/what to replace/s//with what to replace/g\nw\nq\n' | ed file
```

или `cat file_edit_commands | ed file` .

## Examples

### Замена строк в файле на месте

```
sed -i s/"what to replace"/"with what to replace"/g $file
```

Мы используем `-i` для выбора редактирования на месте в `$file` file. В некоторых системах требуется добавить суффикс после флага `-i` который будет использоваться для создания резервной копии исходного файла. Вы можете добавить пустую строку как `-i ''` чтобы опустить создание резервной копии. Посмотрите на *примечания* в этом разделе о опции `-i` .

Терминатор `g` означает глобальную находку / замену в каждой строке.

```
$ cat example
one
two
three
total
$ sed -i s/"t"/"g"/g example
$ cat example
one
gwo
ghree
gogal
```

### Портативное использование

Редактирование на месте, хотя и является общим, является нестандартной функцией. Жизнеспособной альтернативой было бы использовать промежуточный файл для хранения оригинала или вывода.

```
sed 'sed commands' > file.out && mv file.out file
# or
mv file file.orig && sed 'sed commands' file.orig > file
```

Чтобы использовать параметр `-i` как с синтаксисом GNU, так и с FreeBSD, необходимо указать расширение и добавить его в параметр `-i` . Следующее будет принято обоими и создаст два файла, оригинальную версию в `file.orig` и отредактированную версию в `file` :

```
sed -i.orig 'sed commands' file
```

См. Базовый пример с файлом `file` :

```
$ cat file
one
two
three
$ sed -i.orig 's/one/XX/' file
$ cat file                                # the original file has changed its content
XX
two
three
$ cat file.orig                          # the original content is now in file.orig
one
two
three
```

Более сложный пример, заменяющий каждую строку номером строки:

```
$ printf 'one\ntwo\n' | tee file1 | tr a-z A-Z > file2
$ sed -ni.orig = file1 file2
$ cat file1.orig file2.orig
one
two
ONE
TWO
$ cat file1 file2
1
2
1
2
```

## Зачем нужен резервный файл

Для того , чтобы использовать в месте редактирования без файла резервной копии, `-i` необходим дать нулевую длину аргумент а и FreeBSD `sed` требует аргумента `-i` , либо добавляются или отдельно, в то время как дополнительный аргумент расширение GNU требует аргумент дополнить `-i` . Оба поддерживают добавление аргумента к `-i` , но без необходимости `-i''` command неотличим от `-i extension` , поэтому аргумент нулевой длины не может быть добавлен к `-i` .

## Редактирование на месте без указания файла резервной копии переопределяет разрешения только для чтения

`sed -i -e cmd file` будет изменять `file` даже если его разрешения установлены только для чтения.

Эта команда ведет себя аналогично

```
sed -e cmd file > tmp; mv -f tmp file
```

скорее, чем

```
sed -e cmd file > tmp; cat tmp > file; rm tmp
```

В следующем примере используется `gnu sed` :

```
$ echo 'Extremely important data' > input
$ chmod 400 input # Protect that data by removing write access
$ echo 'data destroyed' > input
-bash: input: Permission denied
$ cat input
Extremely important data (#pew! Data is intact)
$ sed -i s/important/destroyed/ input
$ cat input
Extremely destroyed data (#see, data changed)
```

Это можно смягчить, создав резервную копию, указав `SUFFIX` с опцией `i` :

```
$ sed -i.bak s/important/destroyed/ input
$ cat input
Extremely destroyed data
$ cat input.bak
Extremely important data
```

Прочитайте Редактирование на месте онлайн: <https://riptutorial.com/ru/sed/topic/3640/редактирование-на-месте>



---

# глава 11: Удалить команду

## Examples

### Удалить одну строку, содержащую шаблон

Учитывая файл **file.txt** со следующим содержимым:

```
line 1  
line 2  
line 3
```

Вы можете удалить строку из файла с помощью команды `d`.

Соответствующий шаблон окружен разделителем по умолчанию `/` / разделителем, а команда `d` следует за шаблоном:

```
sed '/line 2/d' file.txt
```

Выведенная выше команда выводит:

```
line 1  
line 3
```

Чтобы изменить файл *на месте*, используйте параметр `-i`:

```
sed -i '/line 2/d' file.txt
```

Прочитайте Удалить команду онлайн: <https://riptutorial.com/ru/sed/topic/2177/удалить-команду>

## кредиты

S. No	Главы	Contributors
1	Начало работы с sed	<a href="#">Community</a> , <a href="#">fedorqui</a> , <a href="#">kdhp</a> , <a href="#">SLePort</a>
2	BSD / macOS Sed против GNU Sed против спецификации POSIX Sed	<a href="#">mklement0</a>
3	Ветвление	<a href="#">Ekeyme Mo</a>
4	Диапазон адресов и адресов	<a href="#">Benjamin W.</a> , <a href="#">Sundeep</a>
5	Дополнительные опции	<a href="#">kdhp</a> , <a href="#">mklement0</a>
6	Команда добавления	<a href="#">kdhp</a> , <a href="#">Slawomir Jaranowski</a>
7	подмена	<a href="#">Emil Burzo</a> , <a href="#">fedorqui</a> , <a href="#">kdhp</a> , <a href="#">SLePort</a> , <a href="#">Sundeep</a> , <a href="#">thanasisp</a>
8	Расширенные команды sed	<a href="#">Raju</a>
9	Регулярные выражения	<a href="#">fedorqui</a>
10	Редактирование на месте	<a href="#">AstraSerg</a> , <a href="#">Ekeyme Mo</a> , <a href="#">Emil Burzo</a> , <a href="#">fedorqui</a> , <a href="#">ghostarbeiter</a> , <a href="#">ikrabbe</a> , <a href="#">kdhp</a> , <a href="#">mklement0</a> , <a href="#">Oleg Arkhipov</a> , <a href="#">William Pursell</a>
11	Удалить команду	<a href="#">SLePort</a>