

NÉV

sed – folyamszerkesztő (stream editor)

ÁTTEKINTÉS

sed [-n] [-g] [-e script] [-f sfile] [file] ...

LEÍRÁS

A 'sed' program a megnevezett fájlokat (alapértelmezés szerint a standard bemenetet) a szabványos kimenetre másolja, de közben egy parancsokat tartalmazó szkriptnek megfelelően megszerkeszti.

A '-e' opció az egyszerű szerkesztést jelenti: a szerkesztő parancsot a következő argumentumból veszi. Amennyiben több '-e' is van a parancssorban, megjelenésük sorrendjében hajtja őket végre. Amennyiben csak egyetlen '-e' opció van és nincs '-f', a '-e' elhagyható.

A '-f' opció azt eredményezi, hogy a parancsokat az "sfile" fájlból veszi. Amennyiben több is van belőlük, megjelenésük sorrendjében kerülnek végrehajtásra. A '-e' és '-f' opciók keverhetők.

A '-g' opció azt eredményezi, mintha minden helyettesítési parancsnak 'g' végződése lenne.

A '-n' opció elnyomja az alapértelmezett kimenetet.

A szkript soronként egy parancsot tartalmaz az alábbi forma szerint:

```
[cím [, cím] ] függvény [argumentumok]
```

Általában a sed ciklikusan bemásol az aktuális szövegpuferbe egy sort a bemenetről, azután sorra alkalmazza azokat a parancsokat, amelyek a cím(ek) alapján a sorra vonatkoznak, végül a puffert a szabványos kimenetre másolja és törli.

A '-n' opció elnyomja a normális kimenetet, így csak a 'p' és 'w' parancsok kimenete készül el.

Néhány parancs (n, N) saját sorolvasást végez, és néhány másik (d, D) a szkript minden további utasítását átugorja. (A 'D' parancs elnyomja a szövegpufer törlését, amely normálisan bekövetkezne a következő ciklus előtt.)

Azt is hasznos tudni, hogy létezik egy második pufer ('hold space'), amelyik másolható, hozzáfűzhető vagy felcserélhető az aktuális szövegpuferrel.

A címzés egy decimális számjegy, '\$'-jel, vagy tartalmi címzés lehet. A decimális jegy a legelső beolvasott sor esetén 1, és folyamatosan egyesével növekszik az összes bemeneti fájl sorain keresztül. A '\$'-jel az utolsó bemeneti sort jelenti.

A tartalmi címzés egy '/szabályos kifejezés/' az ed (1) stílusában, a következő különbségekkel:

- (1) A '0' jelsorozat egy puferbe ágyazott újsor-jellel egyezik meg, ' pedig egy tab-bal.
- (2) Címekek nélküli parancssor minden puffert kiválaszt.
- (3) Egy címzéssel rendelkező parancssor minden olyan puffert kiválaszt, amely megfelel a címzésnek.
- (4) Két címzéssel rendelkező parancssor egy tartományt választ ki, melybe beleértendők a megadott határok is. A tartomány annál az első bemeneti puffernél kezdődik, amely megfelel az első címzésnek, és annál végződik, amely ezután először felel meg a másodiknak. Amennyiben a második címzés egy szám, amely kisebb vagy egyenlő, mint az elsőnek kiválasztott sor száma, csak egy sor kerül kiválasztásra. Mihelyt a második cím is megegyezést mutat, a sed program újakezdi az első címzéssel való egyezés keresését; így tetszőleges számú, a címzésnek megfelelő tartomány egyezést mutat a címzéssel.

A '!' negáló operátor azt eredményezi, hogy a parancs azokra a sorokra vonatkozik, amelyeket a címzés nem választott ki.

A függvények következő listájában zárójelben megadjuk a függvény által megengedett címzések maximális számát.

A "szöveg"-gel jelölt argumentumok egy vagy több sorból állhatnak, melyek közül az utolsó kivételével mind ‘\’-re kell végződjön. (A ‘\’ jel elrejt a sorvégjelet.)

A szövegbeli backslash (fordított törtvonal) jeleket a program úgy kezeli, mint az ‘s’ parancs helyettesítési stringjében levőket, és ezek a backslash jelek használhatók a kezdeti szóközök és tabulátorok (whitespace karakterek) megvédésére. Alapértelmezés szerint ugyanis ezeket a bevezető jeleket figyelmen kívül hagyja a sed.

Az "rfile" és "wfile" jelölésű argumentumok a parancssor legvégén kell álljanak. A sed minden "wfile"-t létrehoz a feldolgozás megkezdése előtt. Legfeljebb 10 "wfile" argumentum lehet.

A következőkben ismertetjük a sed által ismert függvényeket.

a "szöveg" (1)

Csatolás (append). A "szöveg"-et a kimenetre írja a következő bemeneti sor olvasása előtt.

b "címke" (2)

Elágazás (branch) a "címket" viselő ‘:’ parancsra. Amennyiben nincs megadva címke, elágazás a szkript végére.

c "szöveg" (2)

Csere (change). Törli az aktuális szövegpuffert. 0 vagy 1 címzéssel, illetve egy kétcímzéses tartomány végén elhelyezi a "szöveg"-et a kimeneten. Elkezd a következő ciklust.

d (2)

Törli (delete) az aktuális szövegpuffert. Elkezd a következő ciklust.

D (2)

Törli az aktuális szövegpuffer első sorát, azaz az első újsor-karakter előtt álló karaktereket és ezt az újsor-jelet. Elkezd a következő ciklust.

g (2)

Helyettesíti az aktuális szövegpuffer tartalmát a második puffer (‘hold space’) tartalmával.

G (2)

Az aktuális szövegpuffer tartalmához hozzáfűzi a második puffer (‘hold space’) tartalmát.

h (2)

Az aktuális szövegpuffer tartalmát a második pufferbe másolja.

H (2)

Az aktuális szövegpuffer tartalmát a második pufferhez fűzi.

i "szöveg" (1)

Beilleszt (insert). Kiírja a "szöveg"-et a szabványos kimenetre.

l (2)

Listáz. Kiküldi a mintaterületet a szabványos kimenetre. Az ‘s’ parancshoz hasonlóan egy ‘w’ opció követheti. A nem nyomtatható karakterek az alábbiak szerint kerülnek kifejtésre:

```
\b -- backspace (ASCII 08)
\t -- tab      (ASCII 09)
\n -- newline  (ASCII 10)
\r -- return   (ASCII 13)
\e -- escape   (ASCII 27)
\xx -- a kétjegyű xx hexadecimális számmal megadható ASCII karakter.
```

n (2)

Az aktuális szövegpuffer tartalmát a szabványos kimenetre másolja, és beolvassa helyette a következő bemeneti sort.

N (2)

A következő bemeneti sort az aktuális szövegpuffer tartalmához fűzi, miközben egy újsor-jelet illeszt közéjük. Az aktuális sor száma megváltozik.

- p (2)
Nyomatat (print). Az aktuális szövegpuffer tartalmát a szabványos kimenetre másolja.
- P (2)
Az aktuális szövegpuffer első sorát, azaz az első újsor-karakter előtt álló karaktereket a szabványos kimenetre másolja.
- q (1)
Kilépés (quit). Elágazás a szkript végére. Nem kezd új ciklust.
- r "rfile" (1)
Belovassa (read) az "rfile" tartalmát és a kimenetre írja, még mielőtt a következő bemeneti sort beolvasná.
- s /szabályos kifejezés/helyettesítés/zászlók (2)

A szabályos kifejezés előfordulásait helyettesíti az aktuális szövegpufferben. A '/' kivételével minden karakter használható. Teljesebb leírást lásd az ed (1) kézikönyv lapján. A zászlók a következők lehetnek: (Bármelyikük előfordulhat, de az is lehet, hogy egyikük sem.)

g -- Globális. A string minden nem átfedő megjelenését helyettesíti. Ha nincs megadva, csak az első megjelenés helyettesítődik.

p -- Kinyomatja (print) a minta területet ha helyettesítés történt.

w -- Írás (write). Az aktuális szövegpuffert egy argumentumként megadott fájlhoz fűzi hozzá, ahogy a 'w' parancsban, ha helyettesítés történik. Ha nincs fájl argumentum, a szabványos kimenetre történik az írás.
- t "címke" (2)
Feltételes elágazás. Elágazás a címkével jelzett ':' paracshoz, ha valamilyen helyettesítés történt a bemeneti fájl legújabb olvasása vagy a legutóbbi 't' vagy 'T' végrehajtása óta. Ha nincs megadva címke, akkor a szkript végére történik elágazás.
- T "címke" (2)
Ellentéte a 't' parancsnak. Elágazás a címkével jelzett ':' paracshoz, ha semmilyen helyettesítés sem történt a bemeneti fájl legújabb olvasása vagy a legutóbbi 't' vagy 'T' végrehajtása óta. Ha nincs megadva címke, akkor a szkript végére történik elágazás.
- w "wfile" (2)
Írás (write). Az aktuális szövegpuffert a "wfile" fájl tartalmához fűzi.
- W "wfile" (2)
Az aktuális szövegpuffer első sorát a "wfile" fájl tartalmához fűzi.
- x (2)
Felcseréli (exchange) az aktuális szövegpuffer tartalmát a második puffer ('hold space') tartalmával.
- y /string1/string2/ (2)
Fordít. A string1-ben előforduló karaktereket a string2-ben azonos pozícióban előfordulóval helyettesíti. A stringek hossza egyenlő kell legyen.
- ! "parancs" (2)
Kivétel. Azokra a sorokra alkalmazza a függvényt (vagy csoportot, ha a függvény '{')', amelyek nem lettek kiválaszva a címzés(ek)kel.
- : "címke" (0)
Ez a parancs semmit sem csinál, csak a 'b' és 't' parancsok elágazásaihoz visel címkét.

```
= (1)
   Az aktuális sorszámot a szabványos kimenetre írja, mint egy külön sort.
{ (2)
   Parancs csoprotosítás. A következő '}'-ig adott parancsokat végrehajtja a címzés(ek) által
   meghatározott sorokra.
```

Az üres parancsot figyelmen kívül hagyja.

ÁTVIHETŐSÉG

Ez az eszköz a BSD 4.1 UNIX sed-je alapján fordított tervezéssel készült, és (amennyire a szerző tudása és a tesztek meg tudják határozni) kompatibilis azzal. A BSD 4.1 sed-jének minden dokumentált lehetősége adott.

Egy dokumentálatlan tulajdonság (az első megjegyzést bevezető 'n' azonos hatását a '-n parancs-ssor-opció'-val) nem került implementálásra.

A BSD sed-hez képest javítások is történtek, melyeket itt terjedelmi okokból nem tárgyalunk.

MEGJEGYZÉS

Ez a program egy 'freeware' komponense a GNU és MINIX operációs rendszereknek. A felhasználó itt megkapja a jogokat a program használatára, módosítására és terjesztésére a következő feltételek mellett:

1. A szerzősége vonatkozó megjegyzést minden forrásfájlban változatlanul meg kell hagyni.
2. A lefordított változat forráskód nélkül nem terjeszthető.

LÁSD MÉG

ed(1), grep(1), awk(1), lex(1), regexp(5)

SZERZŐ

Ezt a kézikönyv lapot Eric S. Raymond <esr@snark.thyrsus.com> írta, habár a sed csomag a GNU-tól származik.

MAGYAR FORDÍTÁS

Horváth András <horvatha@rs1.szif.hu>