

GNU Coreutils

José Pedro Oliveira
(jpo@di.uminho.pt)

Grupo de Sistemas Distribuídos
Departamento de Informática
Escola de Engenharia
Universidade do Minho

Sistemas Operativos
2006-2007

José Pedro Oliveira

GNU Coreutils

Introdução
Output de ficheiros inteiros
Manipulação parcial de ficheiros

Utilitários GNU: opções

Opções de estilo longo comuns a todos os utilitários GNU

- `--help` - página de ajuda
- `--version` - versão do utilitário

Composição de opções de linha de comando

As opções de linha de comando podem, geralmente, ser concatenadas numa única. Por exemplo, as opções

- `-a -b`
- podem ser substituídas por
- `-ab`

José Pedro Oliveira

GNU Coreutils

GNU Coreutils

GNU Coreutils: introdução

Os utilitários da actual *package* **GNU Coreutils** faziam parte das *packages*

- GNU Textutils
- GNU Fileutils

Estilo de opções de linha de comando

- Opções de estilo curto (*short options*)
Exemplo: `-n`
- Opções de estilo longo (*long options*)
Exemplo: `--help`

José Pedro Oliveira

GNU Coreutils

Introdução
Output de ficheiros inteiros
Manipulação parcial de ficheiros

Comando `cat`
Comando `tac`
Comando `nl`
Comando `od`

Conteúdo

- 1 Output de ficheiros inteiros
 - Comando `cat`
 - Comando `tac`
 - Comando `nl`
 - Comando `od`
- 2 Manipulação parcial de ficheiros
 - Comando `head`
 - Comando `tail`

José Pedro Oliveira

GNU Coreutils

Comando cat

Comando cat

Permite concatenar e escrever ficheiros para o *standard output*. O *standard input* é utilizado caso não seja especificado nenhum ficheiro ('-' significa *standard input*).

Algumas opções

- n numera todas as linhas
- E mostra caracter '\$' no fim de cada linha
- T mostra os caracteres TAB

Comando cat: exercícios

Executar os seguintes comandos

- cat
- cat > /tmp/teste.txt
- cat /etc/fstab
- cat < /etc/fstab
- cat /etc/fstab | cat -n
- cat /etc/fstab | cat | cat -n
- cat /etc/fstab /etc/mtab
- cat -n /etc/fstab /etc/mtab > /tmp/resultado.txt

STDIN: indicação de fim de ficheiro: CTRL+D

Comando cat: exemplos

\$ cat /etc/mtab

```
/dev/hda10 / ext3 rw 0 0
none /proc proc rw 0 0
usbdevfs /proc/bus/usb usbdevfs rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
...
```

\$ cat -nE /etc/mtab

```
1 /dev/hda6 / ext3 rw 0 0$
2 none /proc proc rw 0 0$
3 none /dev/pts devpts rw,gid=5,mode=620 0 0$
4 usbdevfs /proc/bus/usb usbdevfs rw 0 0$
...
```

Comando tac

Comando tac

Permite concatenar e escrever ficheiros para o *standard output* invertendo a ordem dos registos. O *standard input* é utilizado caso não seja especificado nenhum ficheiro ('-' significa *standard input*).

Comando **tac**: exemplos

```
$ tac /etc/mtab
```

```
none /dev/shm tmpfs rw 0 0
/dev/hda11 /home ext3 rw 0 0
/dev/hda12 /exports ext3 rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
usbdevfs /proc/bus/usb usbdevfs rw 0 0
none /proc proc rw 0 0
/dev/hda10 / ext3 rw 0 0
```

Comando **nl**: exemplos

```
$ nl /etc/mtab
```

```
1 /dev/hda10 / ext3 rw 0 0
2 none /proc proc rw 0 0
3 usbdevfs /proc/bus/usb usbdevfs rw 0 0
4 none /dev/pts devpts rw,gid=5,mode=620 0 0
5 /dev/hda12 /exports ext3 rw 0 0
6 /dev/hda11 /home ext3 rw 0 0
...
```

Comando **nl**

Comando **nl**

Permite escrever cada ficheiro para o *standard output*, numerando as linhas. O *standard input* é utilizado caso não seja especificado nenhum ficheiro.

Algumas opções

-b estilo - selecciona o estilo de numeração

a - numera todas as linhas

t - numera apenas as linhas não vazias
(por omissão)

-w número - número de caracteres utilizados para o número de linha (por omissão: 6)

Comando **od**

Comando **od**

O comando **od** escreve uma representação de cada ficheiro. O *standard input* é utilizado caso não seja especificado nenhum ficheiro.

Algumas opções

-b - bytes octais como output

-c - caracteres ASCII ou sequências de escape como output

-d - decimal shorts como output

-o - octal shorts como output

-x - hexadecimal shorts como output

Comando od: exemplos

```
$ od -c /etc/mtab
```

```
0000000 / d e v / h d a 1 0 ...  
0000020 3 r w 0 0 \n n o ...  
0000040 r o c p r o c r w ...  
...
```

```
$ od -b /etc/mtab
```

```
0000000 057 144 145 166 057 150 144 141 061 060 040 ...  
0000020 063 040 162 167 040 060 040 060 012 156 157 ...  
0000040 162 157 143 040 160 162 157 143 040 162 167 ...  
...
```

Comando head

Comando head

Imprime as primeiras linhas de cada ficheiro (por omissão: 10). O *standard input* é utilizado caso não seja especificado nenhum ficheiro.

Algumas opções

`-n n` - só as primeiras *n* linhas

Conteúdo

- 1 Output de ficheiros inteiros
 - Comando cat
 - Comando tac
 - Comando nl
 - Comando od
- 2 Manipulação parcial de ficheiros
 - Comando head
 - Comando tail

Comando head: exemplos

```
$ head -n 3 /etc/passwd
```

```
root:x:0:0:root:/root:/bin/bash  
bin:x:1:1:bin:/bin:/sbin/nologin  
daemon:x:2:2:daemon:/sbin:/sbin/nologin
```

```
$ head -3 /etc/passwd
```

```
root:x:0:0:root:/root:/bin/bash  
bin:x:1:1:bin:/bin:/sbin/nologin  
daemon:x:2:2:daemon:/sbin:/sbin/nologin
```

Comando tail

Comando tail

Imprime as últimas linhas de cada ficheiro (por omissão: 10). O *standard input* é utilizado caso não seja especificado nenhum ficheiro.

Algumas opções

`-n n` - só as últimas *n* linhas

`-f` - entra em ciclo a tentar ler mais caracteres no fim do ficheiro (presume-se que o ficheiro esteja a aumentar de tamanho).

Exercícios

Executar as seguintes tarefas

- listar no ecrã as linhas 31 a 40 do ficheiro `/etc/inittab`
- listar no ecrã as linhas 36 a 40 do ficheiro `/etc/inittab` numeradas (incluindo as linhas em branco)

Comando tail: exemplos

```
$ tail -n 3 /etc/passwd
```

```
snort:x:1007:1007:Snort:/var/log/snort:/bin/false
mach:x:100:102:mach user:/var/lib/mach:/bin/bash
dbus:x:81:81:System message bus:/sbin/nologin
```

```
# tail -f /var/log/messages
```

```
...
Apr 23 11:07:25 pateta modprobe: modprobe: Can't locate module sound-service-1-0
Apr 23 14:45:34 pateta modprobe: modprobe: Can't locate module sound-slot-1
Apr 23 14:45:34 pateta modprobe: modprobe: Can't locate module sound-service-1-0
Apr 23 15:51:37 pateta automount[6655]: expired /home/users/nount/jpo
Apr 23 15:58:47 pateta automount[3251]: attempting to mount entry /home/users/nount/jpo
Apr 23 16:01:07 pateta automount[6697]: expired /home/users/nount/jpo
```

Conteúdo

- Sumarização de ficheiros
 - Comando wc
 - Comando md5sum
 - Comando sha1sum
- Operando sobre ficheiros ordenados
 - Comando sort
 - Comando uniq

Comando wc

Comando wc

Permite contar número de bytes, caracteres, palavras e linhas de cada ficheiro especificado (ou do stdin caso nenhum seja especificado)

Algumas opções

- c - contar número de bytes
- m - contar número de caracteres
- w - contar número de palavras
- l - contar número de linhas

José Pedro Oliveira

GNU Coreutils

Sumarização de ficheiros
Operando sobre ficheiros ordenados

Comando wc
Comando md5sum
Comando sha1sum

Comando md5sum

Comando md5sum

Calcula e verifica *checksums* MD5 (de 128 bits).

Algumas opções

- b - le ficheiros em modo binário (DOS/Windows)
- t - le ficheiros em modo texto (por omissão)
- c - verifica *checksums* MD5 contra lista

José Pedro Oliveira

GNU Coreutils

Exemplos

```
$ wc /etc/inittab  
53 229 1666 /etc/inittab
```

```
$ wc -l /etc/inittab  
53 /etc/inittab
```

```
$ wc -l -w /etc/inittab  
53 229 /etc/inittab
```

```
$ wc -lw /etc/inittab  
53 229 /etc/inittab
```

```
$ wc /etc/{fstab,mtab}  
10 60 689 /etc/fstab  
8 48 248 /etc/mtab  
18 108 937 total
```

```
$ wc -l /etc/{fstab,mtab}  
10 /etc/fstab  
8 /etc/mtab  
18 total
```

José Pedro Oliveira

GNU Coreutils

Sumarização de ficheiros
Operando sobre ficheiros ordenados

Comando wc
Comando md5sum
Comando sha1sum

Exemplos

```
$ md5sum /etc/fstab /etc/mtab  
f75de93ef239c23962e1136db0d7d9fa /etc/fstab  
b7565e70cc32a306ee1786c9feb1e2e /etc/mtab
```

Executar os seguintes comandos

- 1 cd /tmp; cp -p /etc/mtab .
- 2 md5sum mtab > CHECKSUMS
- 3 md5sum -c CHECKSUMS
- 4 alterar um caracter do ficheiro mtab
- 5 md5sum -c CHECKSUMS

José Pedro Oliveira

GNU Coreutils

Comando sha1sum

Comando sha1sum

Calcula e verifica *checksums* SHA1 (de 160 bits).

Algumas opções

- b - le ficheiros em modo binário (DOS/Windows)
- t - le ficheiros em modo texto (por omissão)
- c - verifica *checksums* MD5 contra lista

Conteúdo

- Sumarização de ficheiros
 - Comando wc
 - Comando md5sum
 - Comando sha1sum
- Operando sobre ficheiros ordenados
 - Comando sort
 - Comando uniq

Exemplos

```
$ sha1sum /etc/inittab
```

```
78ef239097844c223671e99a79d6b533dced8d3b /etc/inittab
```

```
$ md5sum /etc/inittab
```

```
92a39a223f68e67e9e6c412443851aeb /etc/inittab
```

Comando sort

Comando sort

Permite ordenar linhas de ficheiros de texto.

Algumas opções

- b - ignora caracteres vazios (blanks) no início das linhas
- f - comparação *case-insensitive*
- n - força contexto numérico nas comparações
- r - inverte o resultado das comparações

Exemplos

```
$ sort /etc/passwd
```

```
adm:x:3:4:adm:/var/adm:/sbin/nologin
apache:x:48:48:Apache:/var/www:/sbin/nologin
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
...
```

```
$ sort -r /etc/passwd
```

```
xfs:x:43:43:X Font Server:/etc/X11/fs:/sbin/nologin
webalizer:x:67:67:Webalizer:/var/www/html/usage:/sbin/nologin
vcsa:x:69:69:virtual console memory ...:/dev:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
...
```

José Pedro Oliveira

GNU Coreutils

Sumarização de ficheiros
Operando sobre ficheiros ordenados

Comando sort
Comando uniq

Comando uniq

Comando uniq

Permite remover linhas duplicadas de um ficheiro ordenado.

Algumas opções

- i - comparação *case-insensitive*
- d - imprime apenas linhas duplicadas
- u - imprime apenas linhas não repetidas (únicas)

José Pedro Oliveira

GNU Coreutils

Exemplos

```
$ du -sk *
```

```
224 Config-Frontend-0.18
1512 Event-0.88
500 Locale-Maketext-0.37
660 SQL-Statement-1.08
188 Time-Local-1.10
```

```
$ du -sk * | sort
```

```
1512 Event-0.88
188 Time-Local-1.10
224 Config-Frontend-0.18
500 Locale-Maketext-0.37
660 SQL-Statement-1.08
```

```
$ du -sk * | sort -n
```

```
188 Time-Local-1.10
224 Config-Frontend-0.18
500 Locale-Maketext-0.37
660 SQL-Statement-1.08
1512 Event-0.88
```

José Pedro Oliveira

GNU Coreutils

Sumarização de ficheiros
Operando sobre ficheiros ordenados

Comando sort
Comando uniq

Exemplos

Ficheiro teste.txt

```
linha 1
linha 1
linha 2
linha 3
linha 1
linha 1
```

\$ uniq teste.txt

```
linha 1
linha 2
linha 3
linha 1
```

\$ sort teste.txt | uniq

```
linha 1
linha 2
linha 3
```

José Pedro Oliveira

GNU Coreutils

Conteúdo

- 5 Operando sobre campos de uma linha
 - Comando cut
 - Comando paste
 - Comando join

- 6 Operando sobre caracteres
 - Comando expand
 - Comando unexpand

Descrição do ficheiro `/etc/passwd`

Formato de um registo (linha)

```
root:x:0:0:root:/root:/bin/bash
```

Descrição dos campos

- 1 login
- 2 passwd
- 3 uid
- 4 gid
- 5 comentário
- 6 homedir
- 7 interpretador de comandos de login (shell)

Comando `cut`

Comando `cut`

Permite remover secções de cada linha de um ficheiro.

Algumas opções

-c lista - lista de caracteres (colunas, posição)

-f lista - lista de campos

-d delim - usa *delim* como delimitador em vez do caracter TAB

Exemplos

```
$ cut -d: -f1 /etc/passwd
```

```
root  
bin  
daemon  
...
```

```
$ cut -d: -f1,7 /etc/passwd
```

```
root:/bin/bash  
bin:/sbin/nologin  
daemon:/sbin/nologin  
...
```

Exemplos

```
$ cut -d: -f1,3-4 /etc/passwd
```

```
root:0:0
bin:1:1
daemon:2:2
...
```

```
$ cut -d: -f1,5- /etc/passwd
```

```
root:root:/root:/bin/bash
bin:bin:/bin:/sbin/nologin
daemon:daemon:/sbin:/sbin/nologin
...
```

Exemplos

Ficheiro **colunas.txt**

```
abc 11 1001.00 EUR
def 21 2001.00 EUR
ghi 31 3001.00 EUR
jkl 41 4001.00 EUR
```

```
$ cut -c12-18 colunas.txt
```

```
1001.00
2001.00
3001.00
4001.00
```

```
$ cut -c7-8,9,12-18 colunas.txt
```

```
11 1001.00
21 2001.00
31 3001.00
41 4001.00
```

Comando paste

Comando paste

Escreve sequencialmente linhas de ficheiros alternados, separadas por TABs, para o standard output.

Algumas opções

- d *delim* - usa o delimitador *delim* em vez do caracter TAB
- s - faz paste de linhas de um ficheiro de cada vez, em vez de uma linha de cada ficheiro

Exemplo

Ficheiro **p1.txt**

```
a
b
```

```
$ paste p1.txt p2.txt
```

```
a 1
b 2
 3
```

Ficheiro **p2.txt**

```
1
2
3
```

```
$ paste -d: p1.txt p2.txt
```

```
a:1
b:2
:3
```

```
$ paste -s p1.txt p2.txt
```

```
a b
1 2 3
```

Comando join

Comando join

Para cada par de linhas com campos de junção iguais, escreve uma linha para o standard output. Por omissão o campo de junção é o primeiro, delimitado por *whitespace*.

Algumas opções

- i - realiza comparações *case-insensitive*
- 1 campo - realiza a junção no campo *campo* do ficheiro 1
- 2 campo - realiza a junção no campo *campo* do ficheiro 2
- t car - utiliza *car* como separador de campos

Conteúdo

- Operando sobre campos de uma linha
 - Comando cut
 - Comando paste
 - Comando join
- Operando sobre caracteres
 - Comando expand
 - Comando unexpand

Exemplo

Ficheiro f1.txt

```
a a1  
b b1  
c c1
```

Ficheiro f2.txt

```
a a2  
b b2  
c c2
```

\$ join f1.txt f2.txt

```
a a1 a2  
b b1 b2  
c c1 c2
```

Comando expand

Comando expand

Permite escrever ficheiros, ou o *standard input* se nenhum ficheiro tiver sido especificado, para o *standard output*, com os caracteres tabs convertidos para o correspondente número de espaços.

Algumas opções

- t tab - coloca tabs espaçados de *tab* colunas (por omissão: coloca um tab por cada 8 colunas)
- t tab1,tab2,... - coloca tabs nas colunas especificadas
- i - só converte tabs no início da linha

Exemplos

```
$ cat -T comtabs.txt
```

```
val11~Ival12~Ival13
~Ival22~Ival23
val31~Ival32~Ival33
```

```
$ expand -t 10 comtabs.txt | cat -T
```

```
val11    val12    val13
val22    val23
val31    val32    val33
```

```
$ expand -t 7,20 comtabs.txt | cat -T
```

```
val11  val12    val13
        val22    val23
val31  val32    val33
```

Exemplo

```
Ficheiro semtabs.txt
```

```
val11  val12  val13
val22  val23
val31  val32  val33
```

```
$ unexpand semtabs.txt | cat -T
```

```
val11  val12  val13
~Ival22  val23
val31  val32  val33
```

```
$ unexpand -a semtabs.txt | cat -T
```

```
val11~Ival12~Ival13
~Ival22~Ival23
val31~Ival32~Ival33
```

Comando unexpand

Comando unexpand

Permite escrever ficheiros, ou o *standard input* se nenhum ficheiro tiver sido especificado, para o *standard output*, com as strings de dois ou mais espaços ou caracteres tabs convertidos para o maior número de tabs possível seguido de tantos espaços quanto os necessários.

Algumas opções

-t *tab* - coloca tabs espaçados de *tab* colunas (por omissão: coloca um tab por cada 8 colunas)

-t *tab1,tab2,...* - coloca tabs nas colunas especificadas

-a - converte todos os grupos de dois ou mais espaços ou tabs (e não apenas o inicial) em tabs

Conteúdo

🔍 Referências

Referências

Online

- **GNU Core-utils**

<http://www.gnu.org/software/coreutils/manual/coreutils.html>

- **Using the GNU text utilities**

<http://www-106.ibm.com/developerworks/edu/l-dw-linux-gnutex-1.html>

- **Unix utilities (Parte I, II, III e IV)**

<http://www-106.ibm.com/developerworks/webservices/library/co-unix1.html>

<http://www-106.ibm.com/developerworks/webservices/library/co-unix2.html>

<http://www-106.ibm.com/developerworks/webservices/library/co-unix3.html>

<http://www-106.ibm.com/developerworks/webservices/library/co-unix4.html>