

Esercitazione di Venerdì 16 Marzo 2018

CAT

- 1) Usando la ridirezione dello standard output del filtro cat, creare un file di nome prova (inserire almeno 5-6 linee con più parole per linea).
- 2) Usando la ridirezione dello standard output del comando pwd e del comando ls -l, aggiungere tale output al file prova.
- 3) Usando il comando cat con le opportune ridirezioni, creare una copia del file prova e dargli nome p.txt.
- 4) Usando più volte la ridirezione dello standard output in append aggiungere il contenuto del file prova al file p.txt almeno altre 5 volte in modo da avere un contenuto di molte linee.

MORE

- 5) Usando la ridirezione dello standard input e il filtro more, visualizzare il contenuto del file p.txt.
- 6) Verificare il comportamento del comando more, usando il comando more p*.

SORT

- 7) Usando la ridirezione dello standard input e il filtro sort, verificare se il file di nome prova è ordinato alfabeticamente.
- 8) Usando la ridirezione dello standard input e il filtro sort, mostrare il contenuto del file prova ordinato secondo il normale ordinamento alfabetico.
- 9) Stessa cosa di 8 ma invertendo l'ordinamento.
- 10) Stessa cosa di 8, ma ignorando la differenza fra maiuscole e minuscole.
- 11) Stessa cosa di 8, ma ridirigendo lo standard output su un file di nome ordinato.
- 12) Stessa cosa di 7, ma sul file di nome ordinato.
- 13) Editare il file prova andando a duplicare e/o triplicare alcune linee.
- 14) Stessa cosa di 8, ma attuando l'ordinamento alfabetico senza replicazioni e scrivendo lo standard output su un file di nome ordinato-senza-doppi.

GREP

- 15) Usando la ridirezione dello standard input e il filtro grep, cercare le linee che contengono una certa stringa (o anche un semplice carattere) nel file ordinato-senza-doppi.
- 16) Stessa cosa di 15, ma mostrando i numeri di linea.
- 17) Assicurandosi di avere nel file ordinato-senza-doppi la stessa stringa scritta in maiuscola e in maiuscolo almeno 2-3 volte, stessa cosa di 15, ma cercando la stringa ignorando maiuscole/minuscole.
- 18) Stessa cosa di 15, ma cercando le linee che NON contengono una certa stringa (o anche un semplice carattere).
- 19) Stessa cosa di 15, ma cercando solo le linee che INIZIANO per una certa stringa (o anche un semplice carattere).
- 20) Stessa cosa di 15, ma cercando solo le linee che TERMINANO per una certa stringa (o anche un semplice carattere).
- 21) Stessa cosa di 15, ma cercando solo le linee che TERMINANO per il carattere '.' (PUNTO).
- 22) Utilizzando la soluzione di uno degli esercizi fra 15 e 21, ridirigere lo standard output in un file di nome prova-grep.

REV

- 23) Usando la ridirezione dello standard input e il filtro rev, verificare lo standard output utilizzando il file prova-grep.

HEAD e TAIL

- 24) Usando la ridirezione dello standard input e il filtro head, selezionare le prime 10 linee del file p.txt (vedi esercizio 4).
- 25) Stessa cosa di 24, ma selezionando la prima linea.
- 26) Stessa cosa di 24, ma selezionando le prime 3 linee.
- 27) Usando la ridirezione dello standard input e il filtro tail, selezionare le ultime 10 linee del file p.txt.
- 28) Stessa cosa di 27, ma selezionando l'ultima linea.
- 29) Stessa cosa di 27, ma selezionando le ultime 3 linee.
- 30) Utilizzando la soluzione di uno degli esercizi fra 24 e 29, ridirigere lo standard output in un file di nome prova-head o prova-tail a seconda dei casi.
- 31) Utilizzando il piping dei comandi, isolare in un file di nome p.txt.terza la terza linea a partire dall'inizio del file p.txt.

- 32) Utilizzando il piping dei comandi, isolare in un file di nome p.txt.terzultima la terza linea a partire dalla fine del file p.txt.

WC

- 33) Usando la ridirezione dello standard input e il filtro wc, contare le linee del file p.txt (vedi esercizio 4).
34) Stessa cosa di 33, ma contando i caratteri.
35) Stessa cosa di 35, ma contando le parole.
36) Usando la soluzione dell'esercizio 33, calcolare la somma del numero di linee dei file p.txt e prova.
37) Usando il comando wc su un file di nome pippo (che non esiste) ridirigendo lo standard error su /dev/null , verificare il valore di ritorno del comando/.
38) Utilizzando il comando ps in piping con tee t e in piping con wc -l verificare: a) il numero visualizzato; b) il contenuto del file di nome t creato dal filtro tee.

VARIABILI

- 39) Memorizzare in una variabile di nome x, il numero di linee del file p.txt.
40) Memorizzare in una variabile di nome y, il numero di linee del file prova.
41) Memorizzare in una variabile di nome z, la somma della variabile x e della variabile y e visualizzarne il valore.
42) Con un editor, scrivere un file comandi prova.sh che visualizzi il valore delle variabili x, y e z inserendo anche dei commenti significativi. Rendere eseguibile tale file comandi (verificare che sia eseguibile con ls -l prova.sh) e mandarlo in esecuzione. Quale è il risultato?
43) Rendere la variabile z una variabile di ambiente e riprovare ad eseguire prova.sh. Quale è il risultato?
44) Verificare con il comando env, la presenza di z nell'ambiente.
45) Editare il file prova.sh, aggiungendo un comando che modifica il valore della variabile z e lo visualizza nuovamente, sempre inserendo commenti significativi. Riprovare ad eseguire prova.sh. Quale è il risultato? Quale valore ha la variabile z nella shell interattiva?

METACARATTERI

- 46) Utilizzando la ridirezione a vuoto, creare diversi file con nomi che iniziano e terminano con varie lettere dell'alfabeto maiuscole e minuscole e con numeri; creare anche un paio di directory. Verificare quindi il comportamento dei comandi:
a) echo [a-z]*
b) echo [A-Z]*
c) echo [0-9]*
d) echo *[a-z]
e) echo *[A-Z]
f) echo *[0-9]
Rifare i comandi precedenti usando la negazione: ad esempio per a) echo [!a-z]*
Utilizzare nuovamente i pattern precedenti utilizzando però il comando ls -l: che cosa cambia?

INIBIZIONI

- 47) Con un editor, scrivere un file comandi prova-inibizioni.sh che assegna ad una variabile di shell di nome a la stringa ciao; quindi visualizzare con il comando echo il valore di a, della directory corrente e della espansione del metacarattere *. Inserire dei commenti significativi. Rendere eseguibile il file comandi e mandarlo in esecuzione. Quale è il risultato?
48) Modificare con un editor il file comandi prova-inibizioni.sh e ricopiare la linea con il comando echo andando a inibire TUTTE le sostituzioni. Mandarlo nuovamente in esecuzione. Quale è il risultato?
49) Modificare con un editor il file comandi prova-inibizioni.sh e ricopiare la linea con il comando echo andando a inibire SOLO l'ultima sostituzione. Mandarlo nuovamente in esecuzione. Quale è il risultato?
50) Assegnare ad una variabile di shell di nome l la stringa ls -l \$z. Fare eseguire il comando con \$l. Quale è il risultato? Assegnare alla variabile di shell z il valore prova-inibizioni.sh. Fare eseguire di nuovo il comando con \$l. Il risultato è cambiato rispetto a prima? Provare ora il comando eval \$l: cosa cambia?