

Lab. di Sistemi Operativi
- Esercitazione n° 1 -
a.a. 2016/2017

“Comandi Shell”

Sommario

④ Comandi shell:

④ ls, cd, pwd

④ echo

④ touch

④ cat

④ head

④ tail

④ cut

④ sort

Esercizio n° 0

- ④ 0a) Creare una cartella **EsercitazioneLSO-1** nella directory di lavoro
- ④ 0b) Creare un file testo chiamato **provaFile.txt** di almeno 5 righe utilizzando touch, echo, cat e ridirezione
- ④ 0c) Creare un file testo chiamato **provaFile2.txt** che ha il contenuto di **provaFile.txt** ripetuto 4 volte

Esercizio n° 0

- 0d) visualizzare il contenuto delle variabili di ambiente PATH, USER, SHELL, HOME
- 0e) creare un file testo chiamato **provaVar.txt** che contiene questi dati
- 0f) creare una variabile `useruser` in cui valore sia il nome dello USER ripetuto due volte

- head & tail -

head & tail

Comando/Sintassi	Descrizione
head [-numero] file	visualizza le prime 10 (o -numero) linee di un file
tail [-numero] file	visualizza le ultime 10 (o -numero) linee di un file

Esempio d'uso head:

```
head -40 filename  
oppure  
head -n 40 filename
```

Esempio d'uso tail:

```
tail -30 filename
```

Esercizio n° 1

© Scrivere una combinazione di comandi Unix che consenta di visualizzare:

1. la terza e la quarta riga del file `provaFile1.txt`
2. le penultime 3 righe del file `provaFile1.txt`
3. l' *n*-esima riga del file `provaFile1.txt`

Soluzione 1

```
head -4 provaFile1.txt | tail -2
```

Soluzione 2

```
tail -4 provaFile1.txt | head -3
```

Soluzione 3

```
head -n provaFile1.txt | tail -1
```

Esercizio n° 1

- Ⓢ Definire una sequenza di comandi che assegna ad una variabile primofile il nome del primo file che compare nel listing di ls.
- Ⓢ Definire una sequenza di comandi che assegna ad una variabile ultimofile il nome dell'ultimo file che compare nel listing di ls.

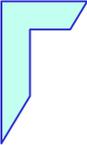
Soluzione 4

```
primofile=$(ls -1 | head -1)
```

Soluzione 5

```
ultimofile=$(ls -1 | tail -1)
```

- /etc/passwd -



/etc/passwd

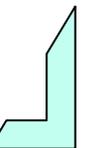
- ⊙ Il file /etc/passwd è il database degli utenti su ogni sistema Unix.
- ⊙ Ad ogni user è dedicata una riga che definisce quali sono i suoi principali attributi:

riga file passwd:

Username:Password:UserID:GroupID:Info:HomeDirectory:Shell

Esempio:

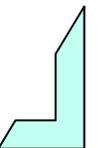
```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
[...]
diego:x:501:503:/home/diego:/bin/bash
```



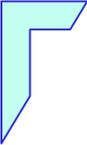


/etc/passwd

- Ⓢ **Username:** Nome dell' user, la login con cui può accedere al sistema;
- Ⓢ **Password:** Campo riservato alla password dell'utente. Può essere scritta direttamente in forma criptata o esserci semplicemente una x (la password c'è ma è scritta altrove, di solito in /etc/shadow). Se c'è un * (asterisco) significa che l'utente o non ha una password o la password non è valida (in questo caso non gli è permesso di login);
- Ⓢ **UserID:** ID dell' user;
- Ⓢ **GroupID:** ID del gruppo di appartenenza;
- Ⓢ **Info:** Contiene informazioni sull'utente non necessarie al sistema (nome esteso, numero di telefono, mail ecc...);
- Ⓢ **HomeDirectory:** Indica la directory della home dell'utente;
- Ⓢ **Shell:** Indica la shell di default per quell'utente.



- comando cut -



Esercizio n° 2

- ⓐ Con un opportuno comando Unix visualizzare:
1. il contenuto del file `/etc/passwd`
 2. estrarre il primo campo dal file `/etc/passwd`

Soluzione 1

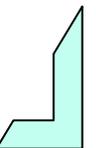
```
cat /etc/passwd
```

Soluzione 2

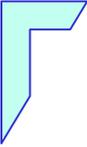
`-f` seguito dal numero del campo estrarre il campo indicato

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

il separatore `-d` (delimiter) seguito dal simbolo del separatore nel file



- comando: chmod -



Esercizio n° 3

- Ⓢ Creare un file LSO.txt nella directory di lavoro:
1. togliere i permessi di lettura al file per l'utente corrente
 2. togliere i permessi di scrittura al file per l'utente corrente
 3. ridare i permessi di scrittura al file per l'utente corrente ed al gruppo a cui appartiene

Soluzione 1

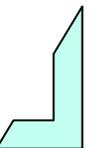
```
chmod u-r LSO.txt
```

Soluzione 2

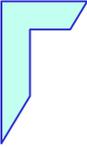
```
chmod u-w LSO.txt
```

Soluzione 3

```
chmod u+w,g+w LSO.txt
```



- comandi: ln, rm -



Esercizio n° 4

- Ⓢ Creare una nuova directory Esempio/
- Ⓢ Creare un link al file LSO.txt nella directory di Esempio/
- Ⓢ Aggiungere i permessi di esecuzione a Esempio/LSO.txt
- Ⓢ Cancellare il file LSO.txt

Soluzione 1

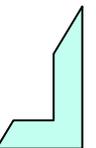
In LSO.txt Esempio

Soluzione 2

chmod a+x LSO.txt

Soluzione 3

rm LSO.txt; rm Esempio/LSO.txt

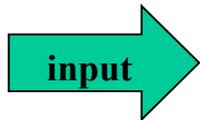


- comando di ordinamento: sort -

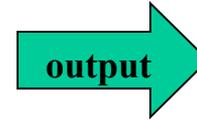
Esercizio n° 4

- Realizzare una combinazione di comandi unix per visualizzare l'ultimo file in ordine alfabetico presente nella directory di lavoro

lab.txt
lab1.txt
lab2.txt
lso.txt
pluto.txt
prova.txt



Sequenza di
Comandi Unix

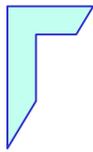


Risultato da ottenere

prova.txt

Soluzione

```
ls | sort -r | head -1
```



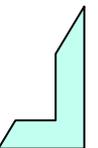
Esercizio n° 5

- Ⓢ Ordinare le righe del file di testo `votoStudenti.txt` (anche in senso inverso):

```
Gianni 20  
Bruno 15  
Carlo 10  
Alice 30
```

Soluzione: ordinamento alfabetico default
`sort votoStudenti.txt`

```
Alice 30  
Bruno 15  
Carlo 10  
Gianni 20
```



Soluzione Esercizio n° 5

Soluzione: ordinamento inverso

```
sort -r votoStudenti.txt
```

```
Gianni 20  
Carlo 10  
Bruno 15  
Alice 30
```

inverte il senso di ordinamento

Soluzione: ordinamento inverso su file

```
sort -r votoStudenti.txt -o file_risultato.txt
```

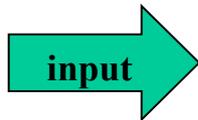
ordinamento inverso con scrittura del risultato nel file _risultato.txt invece che sullo standard output

Esercizio n° 6

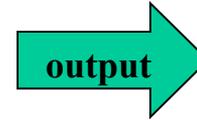
- Realizzare una combinazione di comandi unix che, dato un file di testo esistente **elenco.txt**, crea un nuovo file **nominativo.txt**, contenente la riga di **elenco.txt** che viene lessicograficamente per seconda.

Ad esempio se il contenuto di un file **elenco.txt** è il seguente:

valeria
aldo
roberta
bruno
sandro
paola

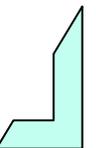


Sequenza di
Comandi Unix



bruno

il comando deve creare il file **nominativo.txt** con il seguente contenuto: **bruno**



Soluzione Esercizio n° 6

Soluzione:

```
sort elenco.txt | head -2 | tail -1 > nominativo.txt
```

ordina il contenuto del file **elenco** in ordine alfabetico

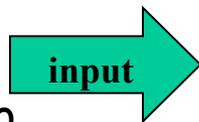
scrive il risultato sul file **nominativo**

aldo

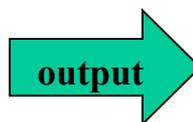
bruno

paola
roberto

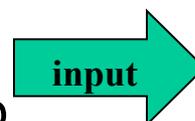
sandro
valeria



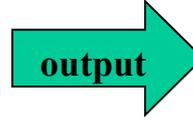
head -2



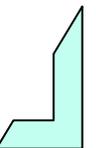
aldo
bruno



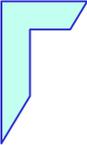
tail -1



bruno



- comando sort: con chiavi di ordinamento -



- sort: chiavi di ordinamento -

- Ⓢ Tramite l'opzione -k è possibile definire una chiave di ordinamento su una porzione della linea contenuta nel file
- Ⓢ Specificando più opzioni -k è possibile definire più chiavi di ordinamento da usare in cascata.

Sintassi sort con opzione -k

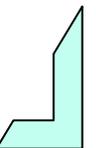
```
sort -k campo_inizio [tipo] [,campo_fine [tipo]]
```

campo_inizio, campo_fine

chiavi per restringere l'ordinamento su una porzione di linea

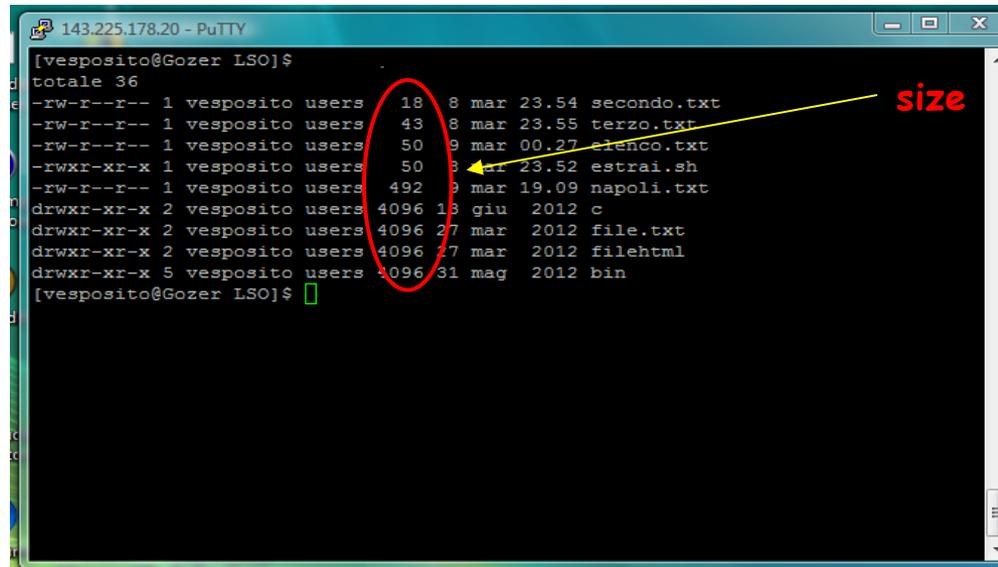
tipo

è il tipo di ordinamento applicato sui campi specificati



Esercizio n° 7

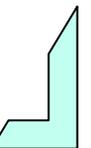
- Realizzare una combinazione di comandi Unix per visualizzare i file presenti nella directory di lavoro con un ordine di dimensione crescente.



```
143.225.178.20 - PuTTY
[vesposito@Gozer LSO]$
totale 36
-rw-r--r-- 1 vesposito users 18  8 mar 23.54 secondo.txt
-rw-r--r-- 1 vesposito users 43  8 mar 23.55 terzo.txt
-rw-r--r-- 1 vesposito users 50  9 mar 00.27 elenco.txt
-rwxr-xr-x 1 vesposito users 50  8 mar 23.52 estrai.sh
-rw-r--r-- 1 vesposito users 492  9 mar 19.09 napoli.txt
drwxr-xr-x 2 vesposito users 4096 13 giu  2012 c
drwxr-xr-x 2 vesposito users 4096 27 mar  2012 file.txt
drwxr-xr-x 2 vesposito users 4096 27 mar  2012 filehtml
drwxr-xr-x 5 vesposito users 4096 31 mag  2012 bin
[vesposito@Gozer LSO]$
```

Soluzione: size crescente

ls -l | sort -k5 -n



Esercizio n° 8

- Ⓜ Applicare al file `votoStudenti.txt` un ordinamento numerico con chiave:

Gianni	20	← Secondo campo numerico
Bruno	15	
Carlo	10	
Alice	30	

chiave di ordinamento: secondo campo del file da ordinare

`sort -k2 -n esempio.txt`  `output`

Carlo	10
Bruno	15
Gianni	20
Alice	30

Ordina numericamente in quanto il campo scelto come chiave di ordinamento è numerico.

Esercizio n° 9

- Realizzate una combinazione di comandi Unix per visualizzare gli ultimi n (numero scelto) utenti collegati.

Soluzione

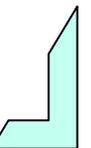
chiave di ordinamento in base al quinto campo

inverti il senso di ordinamento, producendo i risultati a partire dall' utente che si è collegato per ultimo

```
who | sort -k 5 -r | head -3
```

identifica gli utente attualmente collegati al sistema

Visualizza gli ultimi tre utenti connessi



- comandi: wc & mv -

Esercizio n° 10

- Ⓢ Con opportuni comandi Unix modificare il file di testo **votoStudenti.txt** formato da m righe rimuovendo le prime n righe (con $m > n$)

Gianni 20

Carlo 10

Bruno 15

Alice 30

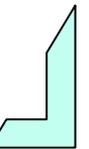
conta il numero di linee presenti nel file

Soluzione:

wc -l votoStudenti.txt (m righe è il valore restituito)

tail -k votoStudenti.txt > tmp (dove $k = m - n$)

mv tmp votoStudenti.txt (rinomina il file tmp nel file **votoStudenti.txt**)



- comando di ricerca: grep & find -

Ricordiamo che...

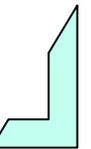
- Le righe di `/etc/passwd` si presentano nella seguente forma:

Username:Password:UserID:GroupID:Info:HomeDirectory:Shell

- Esempio:

Indica la shell di default per quell' utente.

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
[...]
diego:x:501:503::/home/diego:/bin/bash
```



Esercizio n° 11

- ④ Scrivere un comando Unix che visualizza tutti gli utenti diversi contenuti nel file `/etc/passwd`, che usano bash come shell di default

Sintassi

`grep [opzioni] "stringa" nome_file ...`

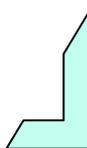
Soluzione

`grep "bash" /etc/passwd`

stringa che si vuole cercare all'interno del file

nome del file in cui cercare la stringa

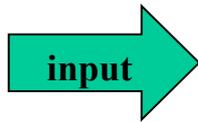
visualizza tutti gli utenti in `/etc/passwd` che hanno come shell di default bash.



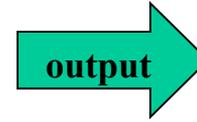
Esercizio n° 12

- Realizzate una combinazione di comandi Unix per visualizzare il terzultimo file presente nel sistema diverso dal file di nome **pluto.txt**, secondo il normale ordinamento prodotto dal comando che permette di visualizzare tutti i file presenti nel sistema.

lab.txt
lab1.txt
lab2.txt
lso.txt
pluto.txt
prova.txt

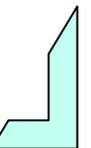


Sequenza di
Comandi Unix



Risultato da ottenere

lab2.txt

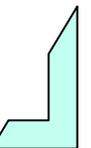
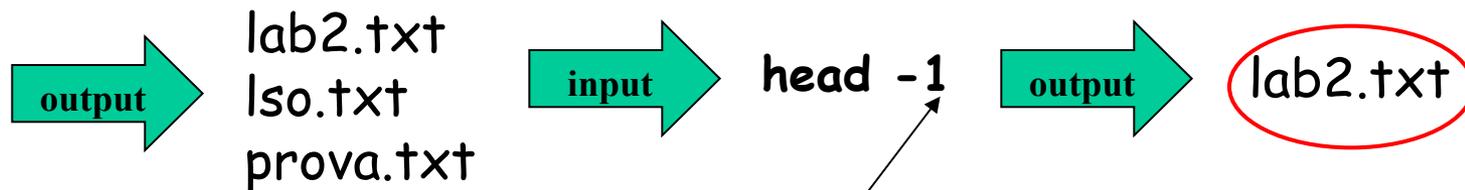
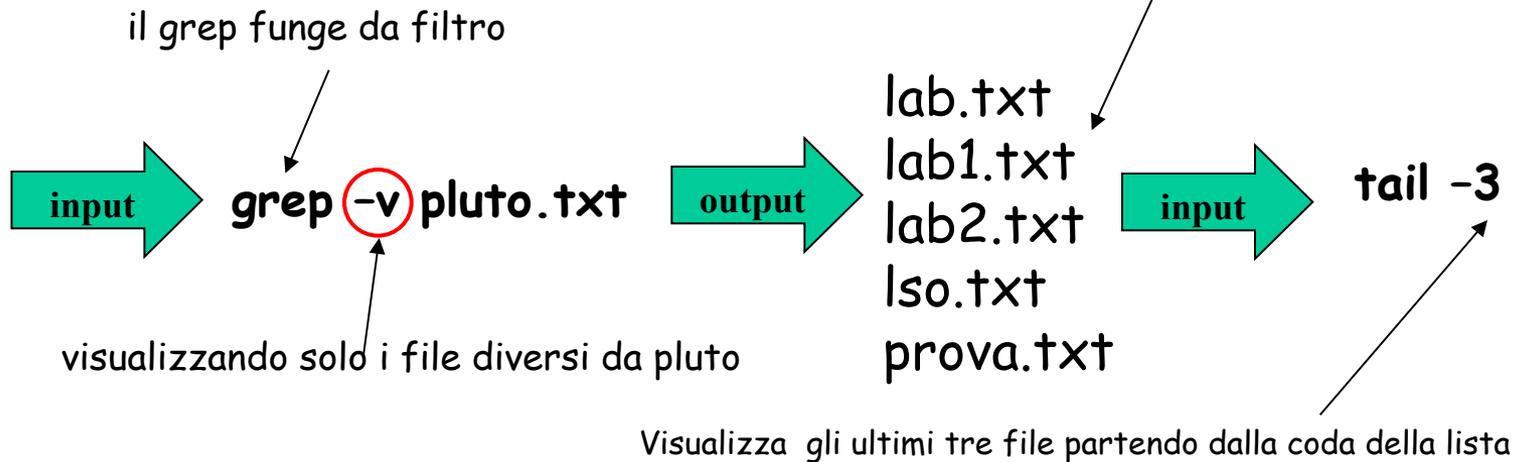


Soluzione Esercizio n° 12

Primo comando: **ls -l** ottengo la lista dei file nella directory di lavoro

lista dei file ad esclusione di quello di nome pluto.txt (filtrato dalla grep)

lab.txt
lab1.txt
lab2.txt
lso.txt
pluto.txt
prova.txt



Soluzione Esercizio n° 12

Soluzione

Elimina dalla lista dei file passati in input il file di nome pluto.txt

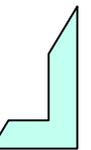
```
ls -l | grep -v pluto.txt | tail -3 | head -1
```

mostra tutti i file

pipe

Visualizza le ultime tre righe

Visualizza la prima riga



Esercizio n° 13

- Ⓢ Nella propria work directory e in tutte le sottodirectory elencare utilizzando un comando Unix tutti i file con estensione ".txt" (file di testo).

cerca nella directory corrente e nelle sottodirectory i file che terminano con .txt

Soluzione con (ls + grep)

```
ls -R | grep "\.txt$"
```

cerca nella directory corrente che ha il nome speciale "punto"

Soluzione con find

```
find . -name "*.txt"
```

al -name usando il carattere jolly "*" si passa non il singolo nome del file ma tutti i file con estensione "txt"

elenca nel path (directory corrente) tutti i file con estensione .txt



- Fine Esercitazione -

