

```

#
#      _ _ _ _ _ ( _ | _ ) _ _ | ( _ ) _ | | _ _
#      / _ \ / _ | / _ | | | / _ | | | / _ \ / _ \
#      | ( _ | \ _ \ \ ( _ | | | \ _ \ | | | ( _ | | _ /
#      \ _ , _ | _ _ / \ _ | | | | | _ _ / _ | | \ _ , _ |
#
# CORSO SCRIPTING / BASH / AWK / SED and more...
#      .:. .:. .:. .:. .:. .:. .:.
# samba <at> netstudent <dot> polito <dot> it

```

Questo corso si pone l'obiettivo di stimolare l'ingegno ad ogni persona che voglia automatizzare il proprio pc.

Per automatizzare il pc ci sono migliaia di modi e linguaggi ed ognuno di essi e' validissimo.

Vi mostrero' alcuni strumenti essenziali per fare delle banali operazioni che solitamente facciamo a mano mettendoci molto tempo, come ad esempio rimuovere molti file o rinominare tutti i file in una cartella con un nome ed un numero sequenziale. Tutto questo con semplici comandi da tastiera e un po' di ingegno .

Strumenti Utili ed Essenziali

- * term - open a new real world
- * vim - the editor
- * AWK - simple and powerfull parser
- * SED - stream editor

script

Gli script sono sequenze di comandi messe in un file. Il file a sua volta viene reso eseguibile e chiamato ad ogni necessita'.

Per rendere eseguibile uno script basta dare questo comando :

```
chmod +x <nome_file>
```

Nella cartella example ci sono una serie di script utili per fare esperimenti e prove.

Link:

<http://www.pluto.it/files/ildp/guide/abs/index.html> [Advanced Guide]

<http://www.shell-fu.org/> [Bash tips&tricks]

vim

E' sempre utile conoscere un editor da terminale, perche' puo' sempre capitare di trovarsi senza grafica e dover fare delle operazioni su file.

VI e' installato anche nei sistemi minimali, VIM e' il suo successore (la M sta per Migliorato) e' presente su Ubuntu/Debian/Slakware/Redhat/ecc... di default
Per utilizzarlo basta dare il comando :

```
$ vim nome_file
e per uscire basta digitare
:wq ( write & quit) oppure :q! (quit without saving)
gU$ : rendere maiuscola tutta la riga
gu$ : rendere minuscola tutta la riga
yy : yank copia tutta la riga
100p <invio>: copia 100 volte la stessa riga
p: paste incolla
dd: taglia riga
man vim
vim -> :help
```

AWK

E' un utile strumento per il parsing di file, ovvero per l'analisi di un file e il filtraggio delle parti interessate.

AWK usa delle macro del linguaggio c ed e' molto intuitivo. Spesso indispensabile quindi e' comodo sapere almeno qualche comando per evitare di fare operazioni ripetitive sui file.

AWK e' presente su tutti in sistemi operativi unix based con tanto di manuale e nelle distribuzioni piu' nuove anche esempi .

```
$ man awk
$ ls /usr/share/doc/mawk/
```

SED

un editor non interattivo, ovvero che non apre il file completamente per modificarlo ma esegue le operazioni nelle linee specifiche.

Se ad esempio abbiamo un file di dimenstione 4GB, non possiamo aprirlo immediatamente con VIM ma nemmeno con un qualsiasi editor perchè ci metterebbe troppo tempo per caricare in memoria tutto il contenuto del file. SED serve per eseguire delle operazione sulle righe specifiche di un file.

Generalmente si usa SED per sostituzioni/inserimenti/cancellazioni la sua codifica e' molto intuitiva e per aiutare nella ricerca di pattern (schemi di ricerca) si usano delle espressioni regolari.

regular expression

sono dei modi per richiamare un gruppo di parole/stringhe, usano dei caratteri particolari per comprendere quello che si desidera ed e' più facile vedere alcuni esempi per capire come funzionano

[0-9] : cerco un numero compreso tra 0 e 9
[0-9][0-9] : cerco due numeri consecutivi compresi tra 0 e 9 , ad esempio 83 o 38 o 13 ecc...
[A-Z] : cerco una lettera maiuscola
^[A-Z] : cerco una lettera maiuscola all'inizio di una riga
ros? : cerco una parola che inizi con 'ros' e abbia un qualsiasi carattere es: rosa,rose,rosi (no: rosina/rosetaz/rosarcazz)
ros.* : cerca una parola che inizi con 'ros' e poi abbia qualsiasi carattere ripetuto, il . sta per qualsiasi carattere e l'* sta per ripetuto
es (rosine/rosando sono andato al bosco e mi son/rosina e' un giocatore che quando fa.... / ros)
^[0-9]\+ : cerco una serie di numeri ripetuti all'inizio di una riga es: (0000 ciao , 1234 pippo)

caratteri importanti:

^ : inizio riga
\$: fine riga
. : qualsiasi carattere
+ : carattere ripetuto (quale carattere? quello specificato precedentemente)
* : qualsiasi carattere ripetuto (quale carattere, quello specificato precedentemente)
[abc] : un insieme di caratteri (in questo caso: a,b oppure c)
\t : tabulatore
\n : a capo

```
#
#
# samba slide ended
#
# s@mba : <samba@netstundet.polito.it>
```