

COMMANDES LINUX / UNIX

COMMANDES DE BASE

cat	afficher le contenu d'un fichier	-n affiche les numéros de ligne
cd	changer de répertoire	.. répertoire parent . répertoire courant
cp	copie de fichiers	-R récursif
chmod	modifier les droits	ugo+--=rwx -R récursif
find	trouver des fichiers	find ~ -name "file*.txt" -print
ls	lister les fichiers	-a fichier cachés -l format long
mkdir	créer un répertoire	
mv	déplacer ou renommer un fichier	
pwd	affiche le répertoire courant	
rm	supprimer un/des fichier(s)	-f forcer la suppression -r récursif
rmdir	supprimer un répertoire	

COMMANDES RESEAUX

ftp	transfert de fichier entre 2 machines	
nslookup	recherche d'adresse IP à partir d'un nom de machine	
ping	vérifier si une machine est connectée sur le réseau	
rcp	copie à distance de fichiers	rcp file.txt richer@sirius:/docs
ssh	connexion à distance	ssh -X login@machine

COMMANDES PROCESSUS

kill	mettre fin à un programme	kill -9 no-de-processus
ps	liste des programmes	ps -ef
top	voir l'ensemble des programmes qui s'exécutent	

COMMANDES SHELL

cut	récupérer des champs d'un fichier	cut -d':' -f1,3
grep	rechercher un motif	grep "toto" etudiants.txt
sort	trier les lignes d'un fichier	-r ordre inverse
uniq	ne pas prendre en compte les lignes identiques qui se suivent	
tr	remplacer ou supprimer des caractères	tr -d '\015' <file.txt
wc	compter les lignes (-l), mots (-w) et lettres (-m)	

ARCHIVAGE / COMPRESSION

tar	création d'une archive	tar -cvf mon-archive.tar *
	extraction	tar -xvf mon-archive.tar
	création + compression	tar -cvzf mon-archive.tgz
	décompression + extraction	tar -xvzf mon-archive.tgz
gzip	compression	gzip un-fichier.txt
	décompression	gzip -d fichier.txt.gz

UTILITAIRES / PROGRAMMES

gftp	file transfer protocol (transfert de fichiers entre 2 machines)
firefox	navigateur web
gedit / kedit	éditeur

PROGRAMME AWK – MOYENNE

awk permet d'analyser les lignes d'un fichier -f fichier-awk

```
BEGIN { somme=0; n=0; }  
/[0-9]+/ { n=n+1; somme=somme+$1; }  
END { moyenne=somme/n; print moyenne; }
```

SHELL

\$1 \$2 ... \$9	arguments du programmes
\$*	ensemble des arguments du programmes
\$#	nombre d'arguments du programme
test -f FICHIER	vérifie si le fichier existe
test ENTIER -eq 0	ENTIER est il égal à 0
test ENTIER -ne 0	ENTIER est il différent de 0
test ENTIER1 -gt ENTIER2	ENTIER1 est il supérieur à ENTIER2
test -n CHAINE	CHAINE est elle de longueur 0
test CHAINE1 = CHAINE2	CHAINE1 est elle égale à CHAINE2
test CHAINE1 != CHAINE2	CHAINE1 est elle différente de CHAINE2

EXEMPLE DE PROGRAMMES SHELL

```
#!/bin/bash  
if test $# -eq 0 ;  
then  
    echo "il manque des arguments"  
    exit 1;  
fi  
i=1;  
for arg in $* ;  
do  
    echo "argument $i = $arg"  
    i=`expr $i + 1`  
done  
  
while [ $# -ne 0 ] ;  
do  
    echo $1;  
    shift  
done
```