SOME UNIX UTILITY

awk

It scans a file for a specified and then carries out an associated action.

e.g.

csh> awk '/pat/ {print}' file1

- Seacrhes the file "file1" for all lines containing the pattern "pat" and outputs the selected lines. As the output has not been redirected, it is sent to the screen

csh> awk 'length > 40 {print}' file1

- Displays all lines from the file "file1" that are longer than 40 characters.

csh> awk '{ print $1 $2 $NF}' file1

- Searches all lines and then output the 1st, 2nd and the last fileds.

grep and egrep

It examines utility examines input for lines containing specified target strings of characters.

e.g.

get all "csh" processes

```
csh> ps -aux|grep csh
shlam    16965  0.0  0.3   56   48 p4 S      0:00 grep csh
shlam    22700  0.0  0.0  380   64 p5 IW     0:01 - (csh)
shlam    22661  0.0  1.1  396  208 p4 S      0:09 - (csh)
shlam    17710  0.0  0.0  372   64 q0 IW     0:00 -bin/csh (csh)
shlam    24007  0.0  0.0  412   64 p3 IW     0:02 - (csh)
shlam    27463  0.0  1.1  392  212 p3 I      0:02 -sh (csh)
```

get all processes not belong to root

```
csh> ps -aux|grep -v root
USER       PID %CPU %MEM   SZ  RSS TT STAT   TIME COMMAND
shlam     3720  0.0  0.3  260   48 p5 I      0:06 telnet ieug41
shlam    27460  0.0  0.0   52    0 p3 IW     0:00 sh myshell
daemon     186  0.0  0.6  164  104 ?  I      1:16 /etc/syslog
```

If we want to match two patterns, we need to use "egrep"

e.g.
We want to get all "nfs" or "csh" processes.

```
csh> ps -aux|egrep "csh|nfs"
root      97  0.0  0.4  140   72 ?  S     12:55 /etc/nfsd 4
root      96  0.0  0.4  140   72 ?  S     13:01 /etc/nfsd 4
root      95  0.0  0.4  140   72 ?  S     13:06 /etc/nfsd 4
root      98  0.0  0.4  140   72 ?  S     13:00 /etc/nfsd 4
shlam  16996  0.0  0.6  208  108 p4 S      0:00 egrep csh|nfs
shlam  22700  0.0  0.0  380   40 p5 IW     0:01 - (csh)
shlam  22661  0.0  1.0  396  184 p4 S      0:09 - (csh)
shlam  24007  0.0  0.6  412  104 p3 I      0:02 - (csh)
shlam  27463  0.0  1.0  392  188 p3 I      0:02 -sh (csh)
```

<u>sed</u>

It allows you to make changes to a file or input without an editor

e.g.

csh> sed '4,7 d' file1 > file2

- uses "files1" for its input and executes the command "4,7 d" which deletes the fourth thorugh seven lines. The output is placed in the file "file2".

csh> sed 's/str1/str2/g' file1 > file2

- replace all occurrences of the word "str1" in "file1" withe the word "str2", storing the output in "file2".

csh> sed 's/^/>/g' file1 > file2

- insert the ">" character at the begining of each line in "file1", sotring the output in "file2".

<u>wc</u>

It counts the number of lines, words, and characters in a file.

e.g

csh> wc file1

- outputs the no. of lines, words, and characters in the file "file1"

csh> ps -ax|grep -v PID|wc -l

- count the no. of process at a host

Compression utility

Here is a list of the most common compression programs, showing the corresponding decompression program and the filename extension used for files compressed with the program.

Compression Porgram	Decompression Program	Filename Extension
compress	uncompress	.Z
gzip	gzip -d	.gz
pack	unpack	.z
packit	unpit	.pit
pkzip	unzip41	.ZIP
Stuffit	unsit	.Sit
zoo210	zoo210	.zoo

We can also use "zcat" to unpack our *.tar.Z file.

e.g

csh> zcat file.tar.Z|tar xvf -