Brief Intro to Linux

Qusay H. Mahmoud (Dr. Q)
Based on material I have used at Carleton University in Summer 2000

Operating Systems aspects:

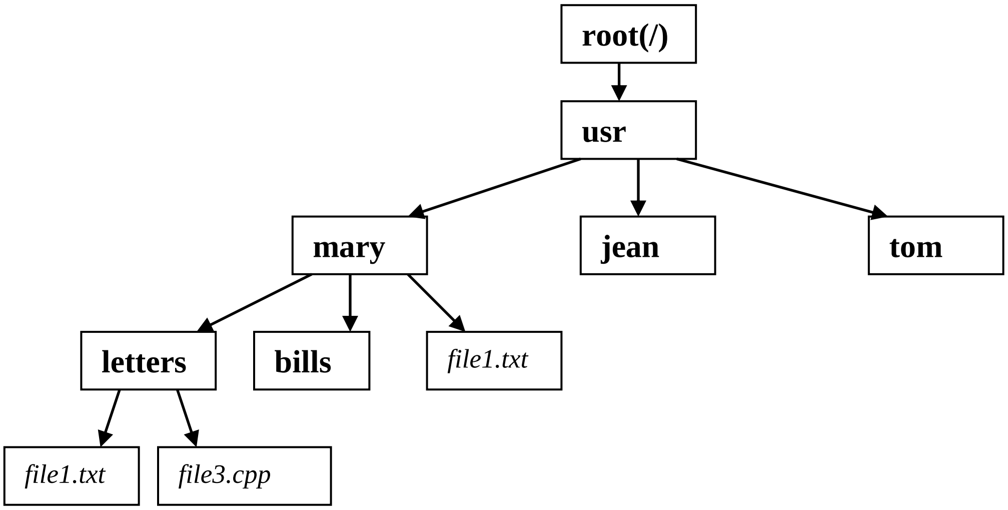
- File management: How files are organized, saved, retrieved, deleted etc.
- Command processor: interprets commands and executes them.
    - possibly command line
    - possibly GUI (mouse click)

- I/O system gives O/S the ability to interact with devices.

Linux file management (basically UNIX)

- files
- directories (folders)

The directory tree or hierarchy



The "path" is the full name of the file, which includes the "path" to the file.
Thus the full path for file1 in the letters directory is

/usr/mary/letters/file1

- Note that there may be more then one file of the same name in different directories.
- File names are case sensitive. Files are usually lower case.

Some Basic Linux Commands

The basic command prompt is >

pwd   gives the current position (your current directory)

```
>pwd
/usr/mary/letters
>
```

cd   change the directory

to change to the root directory:	> cd /
to change to tom	>cd /usr/tom
for your home directory	>cd

ls   list the files and directories in the current directory

>ls -l   (the -l is called an option)   this gives info about each file,
the permissions,size, dates etc.

>ls -l   myfile   gives info about that file.

cp   copy a file

```
>cp /usr/mary/letters/file1.txt  /usr/tom/newletter.txt
```

mv   changes name in the same directory and move a file to another directory

```
>mv  file1.txt   file2.txt
```

rm   erases files. Always be careful here.

```
>rm  file1.txt
>rm   file1.txt   file2.txt
```

mkdir   create a new directory with the given name and in the current directory. So be careful where you do this.

```
>mkdir  assignments
```

rmdir   gets rid of an empty directory. If there are files first remove them.

Permissions

- Each file and directory has permissions for security.
- The permissions can be seen by using the command ls -l.
- The owner can change these.

The permissions consist of three groups of 3 letters: rwx

r read permission.
w write permission may change the file
x execute permission may execute a program.

The three groups are:

- u user you
- g group a group of people recognized by Linux
- o others

example:

rwxrwxrwx everyone can read, write, execute
rwxr-xr-x user can do all, group and others may read and execute but not change files
r-x------ user may not change file, others have no access.

<u>chmod</u> To change permissions.

>chmod g+w to the group add the write permission.
>chmod go-rx to the group and others remove r,x permission.
>chmod g+w, o-r add w to the group, remove r from others.

<u>cat</u> to display the file (but it may scroll past you) and concatenate files

>cat file1.txt outputs the file to the screen
>cat file1.txt file2.txt outputs both files to the screen sequentially

<u>more</u> to look at files one screen at a time

>more filename.txt outputs the file to the screen one screenfull at a time.

Then use:

- spacebar to advance a screen
- The return to advance a line
- b to backup one screen
- h to see all the possible commands

- q to exit the command

Redirection

- The standard input is from the keyboard
- The standard output is to the monitor.
- You may want to change this.
- You may want to read data from a file.
- You may want file to be redirected into a file.
- Use the redirect operators <, >

examples:

to create a file: use cat and direct the input into a named file.

```
> cat >letter
hello mom
how are you?
<ctrl-d>
```

to concatenate two files and create a third. Redirect the output of cat from the monitor to the desired file.

```
>cat file1.txt file2.txt  >  file3.txt
```

this can be used in running a program to get the input from one file and redirect output to a second file.

man an online help manual. Since there may be a lot of stuff it works like "more" in terms of scrolling.

```
>man man     get help on the manual
>man cat    get help on the cat command.
```

grep search a file or files for a string.

```
>grep  hello  letter
```

Some Naming Conventions

Wildcards   *

a group of files may be accessed using the wildcard naming convention.
*   anything

ls g*	list all files beginning with g
ls go.*	list all files named go with any extension
ls *	list all files

caution be very careful when deleting files and using rm as you may delete alot of files.

shortcuts

going back and forth between directories may be tedious so use the following.

your current working directory is a dot .
your home directory is a tilde ~
the directory one level up is two dots   ..

examples:

to go back one level   >cd ..
to go to your home directory   >cd ~
to use a file in your home directory when you are elsewhere use the file name with ~ > more ~mydirectory/myfile