UNIX Tutorial One

1.1 Listing files and directories

When you first login, your current working directory is your home directory is your home directory. Your home directory has the same name as your username, for example, ch504_1, and it is where your personal files and subdirectories are saved.
To find out what is in your home directory, type

```
 ls (short for list)
```

The `ls` command lists all the contents of your current working directory.

`ls` does not, in fact, cause all the files in your home directory to be listed, but only those ones whose name does not begin with a dot(.). Files beginning with a dot (.) are known as hidden files and usually contain important program configuration information. They are hidden because you should not change them unless you are very familiar with UNIX !!!

To list all files in your home directory including those whose names begin with a dot, type

```
ls –a
```

`ls` is an example of a command which can take options: -a is an example of an option. The options change the behavior of the command.

1.2 Making directories

We will now make subdirectory in your home directory to hold the files you will be creating and using in the course of this tutorial. To make a subdirectory called `unixstuff` in your working directory type,

```
mkdir unixstuff
```

To see the directory you have just created, type

```
ls
```

1.3 Changing directory

The command `cd directory` means change the current working directory to ‘directory’. The current working directory may be thought of as the directory you are in, i.e. your current position in the file-system tree.

To change the directory you have just made, type

```
cd unixstuff
```

Type `ls` to see the contents (which should be empty)

Exercise 1a

Make another directory inside the `unixstuff` directory called `backups`

Still in the `unixstuff` directory, type

```
ls -a
```

As you can see, in the unixstuff directory (and in all other directories), there are two special directories (.) and (..)

In UNIX, (.) means the current directory, so typing

```
cd  .
```

[Note that there is a space between cd and the dot]

Means stay where you are (the `unixstuff` directory).

This may not seem very useful at first, but using (.) as the name of the current directory will save a lot of typing, as we shall see later in the tutorial.

(..) means the parent of the current directory, so typing

```
cd ..
```

Will take you one directory up the hierarchy (back to your home directory). Try it now.

Note : Typing cd with no argument always returns you to your home directory. This is very useful if you are lost in the file system.

pwd (print working directory)

Pathname enable you to work out where you are in relation to the whole file-system. For example, to find out the absolute pathname of your home directory, type cd to get back to your home directory and then type

pwd

The full pathname will look something like this

```
/home/ch504/ch504_#/unixstuff
```

Which means that `unixstuff` (your home directory) is in the directory `ch504_#` (the group directory), which is located in `ch504`.

Exercise 1b

Use the commands `ls`, `pwd` and `cd` to explore the file system.

1.4 Understanding pathnames

First type cd and get back to your home directory, then type

```
ls unixstuff
```

to list the contents of you r unixstuff directory

Now type

```
ls backups
```

You will get a message like this

```
backups: No such file or directory
```

The reason is, backups is not in your current working directory. To use the command on a file (or a directory) not in the current working directory (the directory you are currently in), you must either cd to the correct directory, or specify the full pathname. To list the contents of your backups directory

```
ls unixstuff/backups
```

~ (your home directory)

Home directories can also be referred to by the tilde (~) character. It can be used to specify paths starting at your home directory. So typing

```
ls ~/unixstuff
```

will list the contents of your unixstuff directory, no matter where you currently are in the file system.

`ls ~` (it lists the home directory contents)

`ls ~/..` (it lists the contents above the home directory)

1.5 Copying files

`cp file1 file2` is the command which makes a copy of file1 in the current working directory and calls it file2.
What we are going to do now, is to take a file stored in an open access area of the file system, and use the `cp` command to copy it to your `unixstuff` directory.
First, cd to your `unixstuff` directory.

```
cd ~/unixstuff
```

Then make a file named file1

```
vi file1
```

To save file1, type

```
esc :wq
```

To copy it to file2, type

```
cp file1 file2
```

Type ls and you will see the following files

```
file1 file2
```

1.6 Moving files

`mv file1 file2` (moves or renames) file1 to file2
To move a file from one place to another, use the mv command. This has the effect of moving rather than copying the file, so you end up with only one file rather than two.
It can also be used to rename a file, by moving the file to the same directory, but giving it a different name.
We are now going to move file1 file to your `backups` directory.
First, change directories to your `unixstuff` directory. Then, inside the `unixstuff` directory, type

```
mv file1 backups
```

Type `ls` and `ls backups` to see if it has worked.

1.7 Removing files and directories

`rm` (remove), `rmdir` (remove directory)
To delete(remove) a file, use the `rm` command. As an example, we are going to create a copy of the file1 file then delete it.
Inside your `unixstuff` directory, type

```
cp file1 tempfile1
ls  (to check if it has created)
rm tempfile1
ls (to check if it has deleted the file)
```

You can use the `rmdir` command to remove a directory (make sure it is empty first). Try to remove the `backups` directory. You will not be able to do it since UNIX will not let you remove a non-empty directory.
If you want to delete a directory which already contains some files then type the command `rm -rf tempdir` (first create a directory named tempdir and then inside it create a file named tempfile)

Exercise 1c
Create a directory called tempstuff using mkdir, then remove it using the rmdir command.

1.8 Displaying the contents of a file on the screen

First copy file1 from home directory by typing the command,

```
cp  /home/ch504/ch504_55/file1 ~
```

clear (clear screen)

before you start the next section, you may like to clear the terminal window of the previous commands so the output of the following commands can be clearly understood.

At the prompt, type

```
clear
```

This will clear all text and leave you with the prompt at the top of the window.

cat (concatenate)

The command cat can be used to display the contents of a file on the screen. Type

```
cat file1
```

less

The command less writes the contents of a file onto the screen a page at a time. Type

```
less
```

Press the spce-bar if you want to see another page, type q if you want to quit reading. As you can see, less is used in preference to cat for long files.

head

The head command writes the first ten lines of a file to the screen. First clear the screen then type

```
head file1
```

Then type

```
head -5 file1
```

What difference did the -5 do to the head command?

tail

The tail command writes the last ten line of a file to the screen.

Clear the screen and then type

```
tail file1
```

How can you view the last 15 lines of the file?

1.9 Searching the contents of the file

Simple searching using less

Using less, you can search though a text file for a keyword (pattern). For example, to search through file1 for the word science, type

```
less file1
```

Then still in less (i.e. don’t press q to quit), type a forward slash [/], followed by the word to search

```
/md
```

As you can see, less finds and highlights the keyword. Type [n] to search for the next occurrence of the word.

grep (don’t ask why it is called grep)

grep is one of many standard UNIX utilities. It searches files for specified words or patterns. First clear the screen, then type

```
grep md file1
```

as you can see, grep has printed out each line containing the word science.

Try typing

```
grep Md file1
```

the grep command is case sensitive amd so it can distinguish between Science and science.

To ignore the upper/lower case distinctions, use the –i option, i.e. type

```
grep –i md file1
```

to search for a phrase or pattern, you must enclose it in single quotes (the apostrophe symbol). For example to search for spinning top, type

```
grep –i ‘Non-equilibrium’ file1
```

some of the other options of grep are:

-v display those lines that do not match

-n precede each matching line with the line number

-c print only the local count of unmatched line

wc (word count)

Ahandy little utility is the wc command, short for word count. To do a word count on file1, type

```
wc −w file1
```

To find out how many lines the file has, type

```
wc −l file1
```

Summary

Command	**Meaning**
ls	List files and directories
ls -a	List all files and directories
mkdir	Makes a directory
cd *directory*	Changes to named *directory*
cd ~	Changes to home directory
cd ..	Changes to parent directory
pwd	Displays the path of current working directory
cp *file1 file2*	Copy *file1* and call it *file2*
mv *file1 file2*	Move or rename *file1* to *file2*
rm *file*	Remove a *file*
rmdir *directory*	Removes a *directory*
cat *file*	Displays a *file*
more *file*	Displays a file a page at a time
head *file*	Displays the first few lines of a *file*
tail *file*	Displays the last few lines of a *file*
vi *file*	Displays a *file*
grep ‘keyword’ *file*	Search a file for keywords
wc *file*	Count the number of lines/words/characters in *file*