UNIX

What is the purpose of the session ?

1. As a tester why I need Unix knowledge?
2. How Unix knowledge will be helpful to increase my job opportunities?
3. As a fresher or testing fresher do I need testing knowledge?

What we are not covering in this session

1. This session is not covering Advanced Unix

2. This session is not designed with development focus

How to reach out to me in case of doubts in this Unix

- qtramana@gmail.com

UNIX is a CUI operating system.

Operating System:

An operating system can be defined as the software that controls the H/W resources of the computer and provides an environment under which programs can run.

Flavors of UNIX:

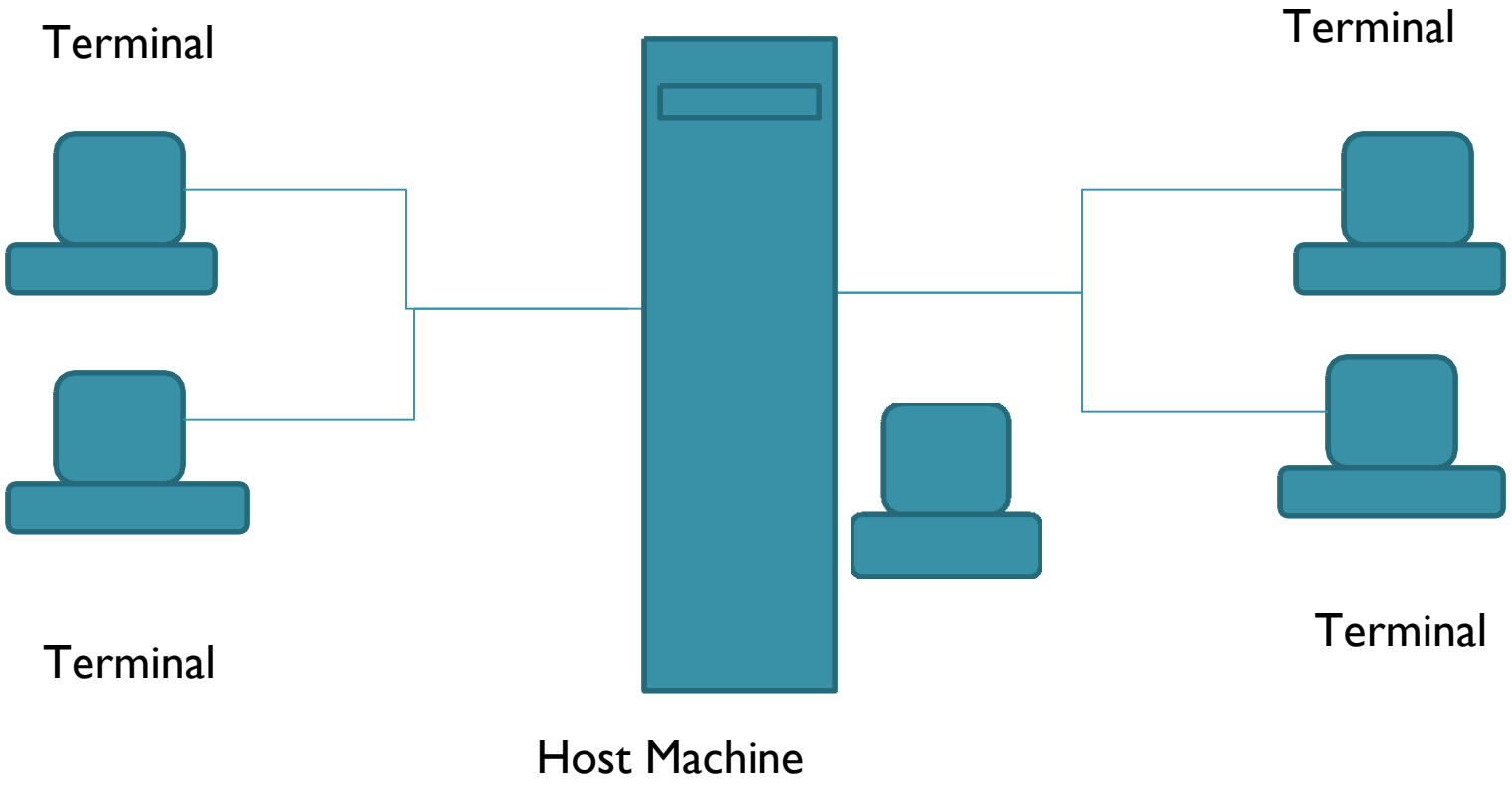
Aix by IBM
Macos by apple
Red hat linus by red hat s/w
Solaries by sun solaries

Features of Unix

The Unix OS offers several features, the important of which are discussed below.

- ❑Multiuser Capability
- ❑Multitasking Capability
- ❑Communication
- ❑Security
- ❑Portability

Multiuser Capability



- Several users can use the same computer simultaneously *i.e. , more than one keyboard and terminal can be connected to one computer.*
- The same data to be shared by all

Multitasking Capability

Performing tasks simultaneously rather than sequentially.

A multi tasking operating system allows more than one program to be running at a time

Communication

Communication between different terminals

Security

Unix provides 3 levels of security to protect data.

Assigning passwords and login names to individual users.

At file level

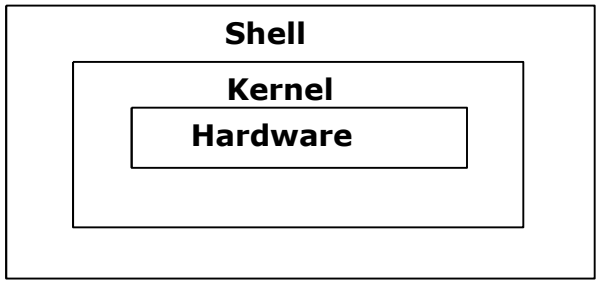
File encryption utility.

Portability

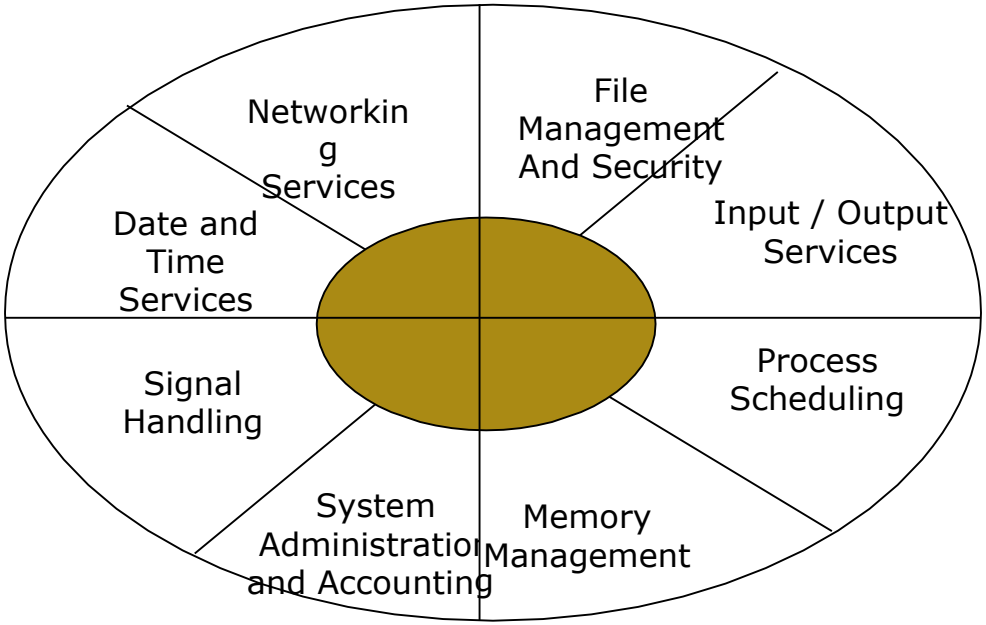
It can be ported (Transfer from one system to another) to almost any computer system.

Architecture of the UNIX operating system

User



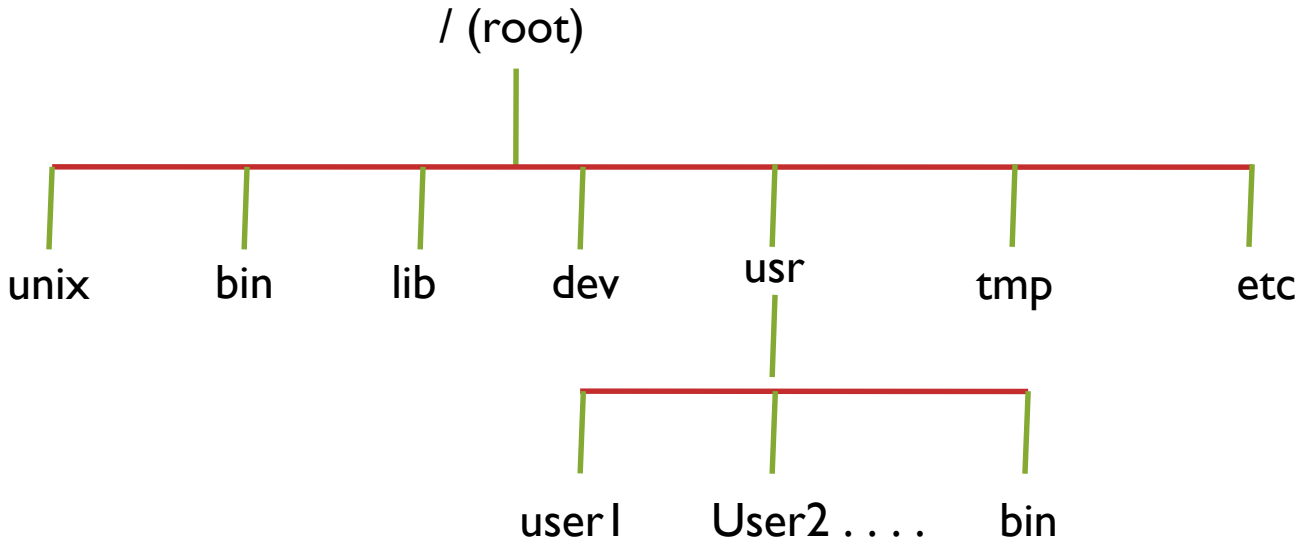
Shell:
The shell reads your commands
And interprets them as a requests
And then conveys them to the
kernel which ultimately execute them



Kernel: (Heart of nix)
Which interacts with the actual
H/W In machine language.

Functions of Kernel:
It manages files.
Manages Memory.
Scheduling of various programs.

The Unix file system



unix: Unix kernel itself
Bin: Directory contains executable files
Lib: Directory all the libery functions provided by Unix.
Dev: Directory contains files that controls various I/P, O/P devices
Bin: Which contains additional Unix commands.
Etc: binary executable files.

Basic Commands

logname : It prints the login name of the user (Current user name)

pwd: It displays the present working directory.

date : it displays system date and time (current date and time)

who am i: It displays current user name, terminal number, date and time at which you logged in.

who : To display data about all the users who have logged into the system currently.

Creating Files:

touch: It creates zero byte file.

Syntax: touch filename

Example: touch file1

This creates a file called file1 , the size of the file would be zero bytes.
Note: touch does not allow you to store anything in a file.

How to create several empty files?

What if you want to store a few lines in a file?

cat: to create file and another to display the contents of an existing file.

Syntax: Cat > filename

Command: cat > test

******

******

******

Ctrl+d

cat >> test -- to append data to the file.

cat> sample1 sample2 > newsample

This would create newsample, which contains contents of sample1 followed by sample2

What if newsample already contains something?

It would be overwritten

cat filename --> to view the contents of an existing file.
cat file1 file2 file3.

Removing files:

Command: rm

rm removes the given file or files supplied to it.

Syntax: rm filename

Ex: rm file1
It removes file1.

rm –i filename → i- interactively

rm file1 file2 file3
It removes 3 files at a time

Creating directory:

mkdir: make directory/creates a directory

Syntax: mkdir directory name

Ex: mkdir hyderabad

Creating multiple directories:
mkdir dir1 dir2 dir3

mkdir –p dir1/dir2/dir3/dir4

Creates all the parent directories specified in the given path.

Changing directory:

Syntax: cd directory name
Ex: cd your desired directory name.
Cd → current users home directory

Cd .. → to change into parent directory.

Removing directory :

rmdir directory name : to delete a directory but directory should be empty

rm –r directory name

rm –r dir1 -> recursively (–r) removes all contents of dir1 and also dir1 itself

Copy a file:

Syntax: cp source file target file
Command: cp sample1 sample2

This will copy contents of sample1 into a sample2. if sample2 already existed it overwrites.

cp –i sample1 sample2 → if sample2 already existed then it asks the confirmation.

Rename a file:
If you want to rename the file test to sample we would say:

mv test sample

mv command also has the power to rename directories.
mv olddir newdir

Note: moving a file implies removing it from its current location and copying it at a new location
mv file1 file2 newdir

Word count:

Syntax: wc filename

It counts the number of lines, words and characters in the specified file or files.

Wc –l file1

Wc file1

Wc file1 file2 file3

Wc –lwc file1 fil2

Sort:

1.Sort command can be used for sorting the contents of a file.
2.It can merge multiple sorted files and store the result in the
specified output file.
3.Sort can display unique lines.
Note: Sorting is done according to ASCII collating sequence.

Syntax: sort filename
Command: Sort myfile
Sorting multiple files:

sort file1 file2 file3
sort –o myresuly file1 file2 file3
sort –u –o result file1 file2 file3
sort –m file1 file2
-m → Merge file1 content with file2.

Removing duplicate lines using uniq:

uniq utility compare only adjacent lines, duplicate lines must be next to each other in the file. To solve this problem you can use command as follows

$ sort personame | uniq

Listing files and directories:

Lists the content of the current or specified directory.

ls

ls –a → to display files and directories including hidden files.
. → stands for current directory
.. → parent of the current directory.

Note: These two entries automatically get created in the dir whenever the dir is created.

Ls –ltr l → it displays files and dir in long format

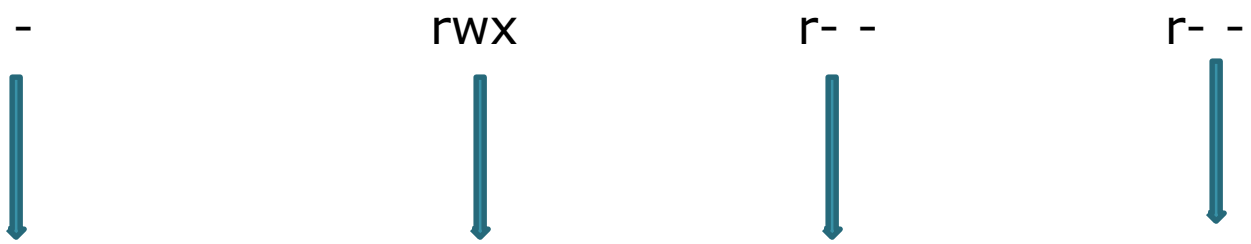
Ls –l |grep ^d → to display only directories
-→ ordinary file

Permissions:

There are 3 classes of file permissions for the 3 classes of users.

1. Owner/user
2. Group
3. Others

Ls –l



Permissions	**weight**
R- read	- 4
W- write	-2
X- execute	-1

Changing file permissions:

Chmod to change file/dir permissions.

Chmod 700 filename

gzip: This command compresses the given file and replaces it with the compressed version name

Syntax: gzip filename

gunzip: to unzip compressed file

Syntax: gunzip filename.gz

Filter commands:

1.Grep

2.Sort

3.cut

Grep: Globally search a regular expression.
syntax: grep "word-to-find" {file-name}

Command: grep hyderabad sample1
grep will locate all lines for the " hyderabad " pattern and print all (matched) such line on-screen.

grep "." myfile.txt
The above command returns every line of myfile.txt.

Options

-c → it returns only number of matches.

-i → ignores case while searching

-v → returns lines that do not match the test

Cut command:

It cuts or pick up a given number of character or fields from the specified file.

Command: cut –f 2 file1 → it displays second filed in file1.
Cut –f 2,4 file1 → it displays 2,4th fields in file1.

Here, cut command assumes that fields are separated by tab character.

cut –f 1-5 file1 → it displays 1 to 5th fields in file1.

Let us say, each piece of information is separated by a “,” then command would be

cut –f 1-5 –d”,” file1

Head: Head prints the first N number of data lines of the given input. By default, it prints first 10 lines of each given file.

syntax: head –n filename
Command: head -20 file1 → it displays first 20 lines from file1

Tail : Tail prints the last N number of lines from given input. By default,
it prints last 10 lines of each given file.

Syntax: tail -5 filename
Command: tail -5 file1

Process: Process is kind of program or task carried out by your PC
Process & PID defined as:
"An instance of running command is called **process** and the number printed by shell is
called **process-id (PID)**, this PID can be use to refer specific running process."

Running command in the background: to run command in background , you end it with an &

Command: cp file1 file2 &

How to kill job/stop job:

syntax: kill processid
Ex: kill 1020

Ps → To see currently running process

diff Command:

The UNIX diff command compares the contents of two text files and

outputs a list of differences.

it's telling you how to change the first file to make it match the second file.

Options: -w → ignore all white spaces
-i → Ignore cases

Syntax: diff file1 file2

The first line of the diff output will contain:

line numbers corresponding to the first file,
a letter (a for add, c for change, or d for delete), and
line numbers corresponding to the second file.

Lines preceded by

"<" means that the text appears in file1, and

lines preceded by

">" indicates that it comes from file2.

The three dashes ("**---**") separate the lines of file 1 and file 2.

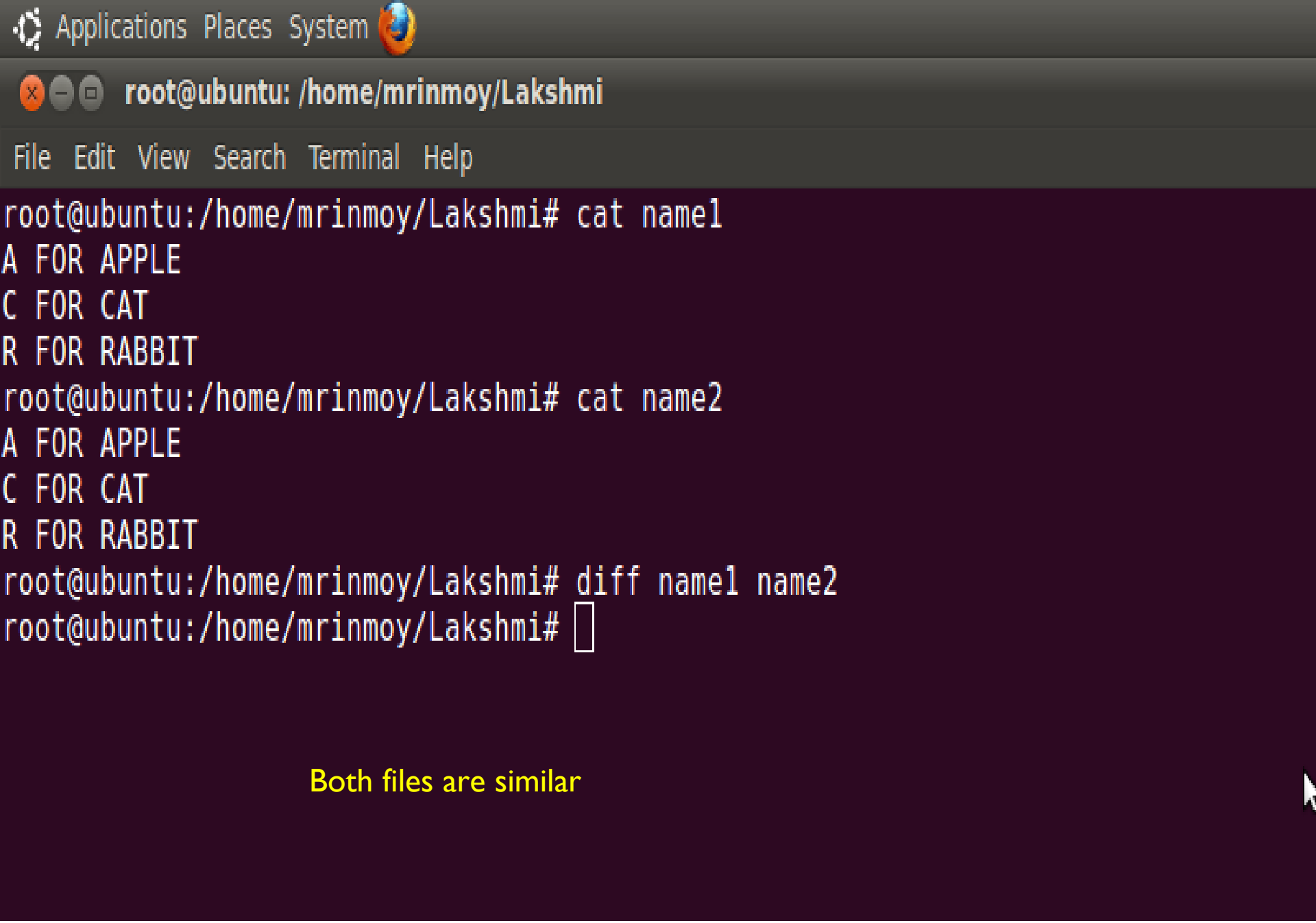
In our output , "**1,3c1,3**" means:

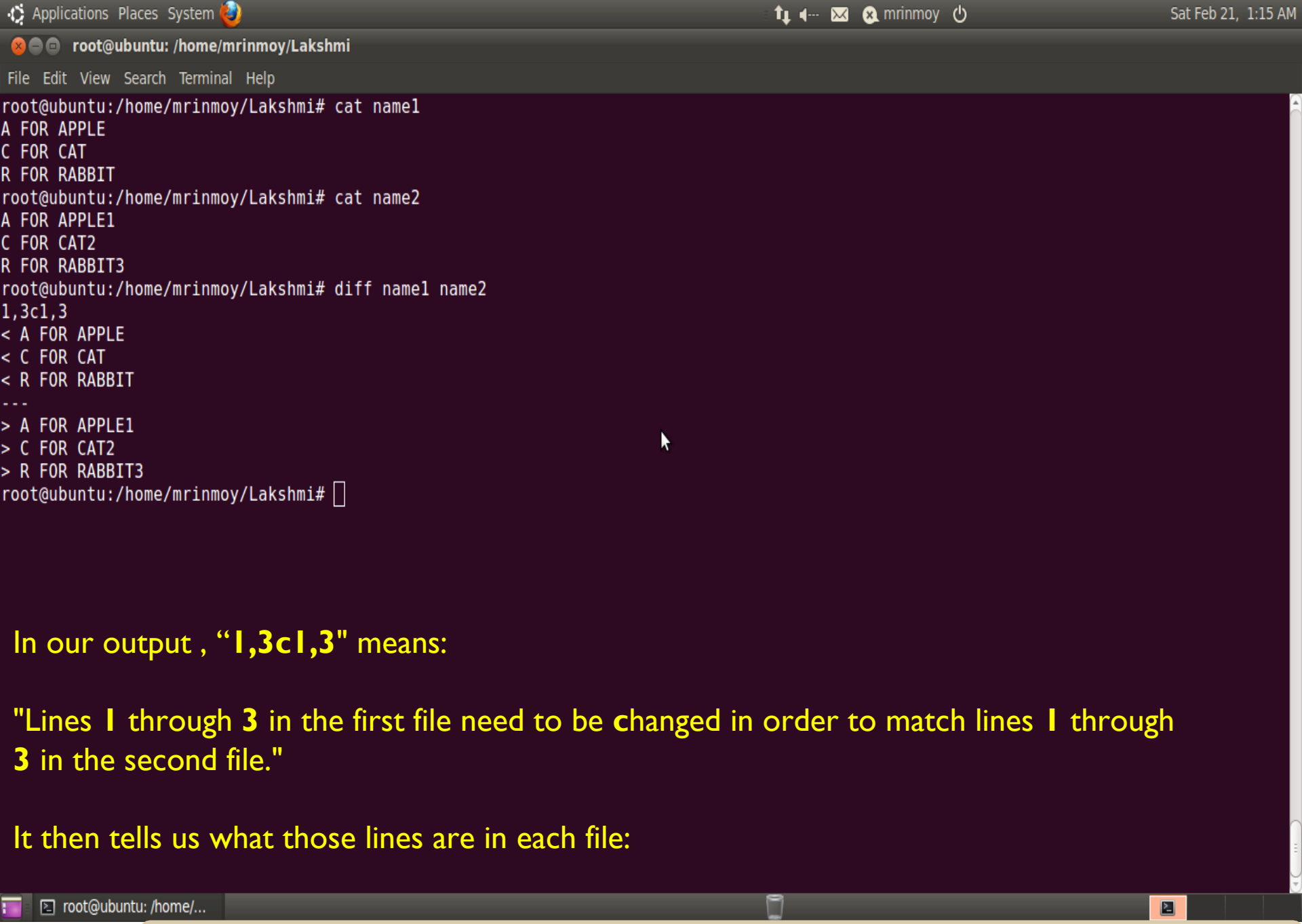
"Lines **1** through **3** in the first file need to be **c**hanged in order to match lines **1** through **3** in the second file." It then tells us what those lines are in each file:

2a3 > Here, the output is
telling us "After line **2** in the first file, a line needs to be **a**dded:
line **3**from the second file." It then shows us what that line is.

4d3 Here, the output is telling us

"You need to **d**elete line **4** in
the first file so that both files sync up at line **3**." It then shows us the contents
of the line that needs to be deleted.

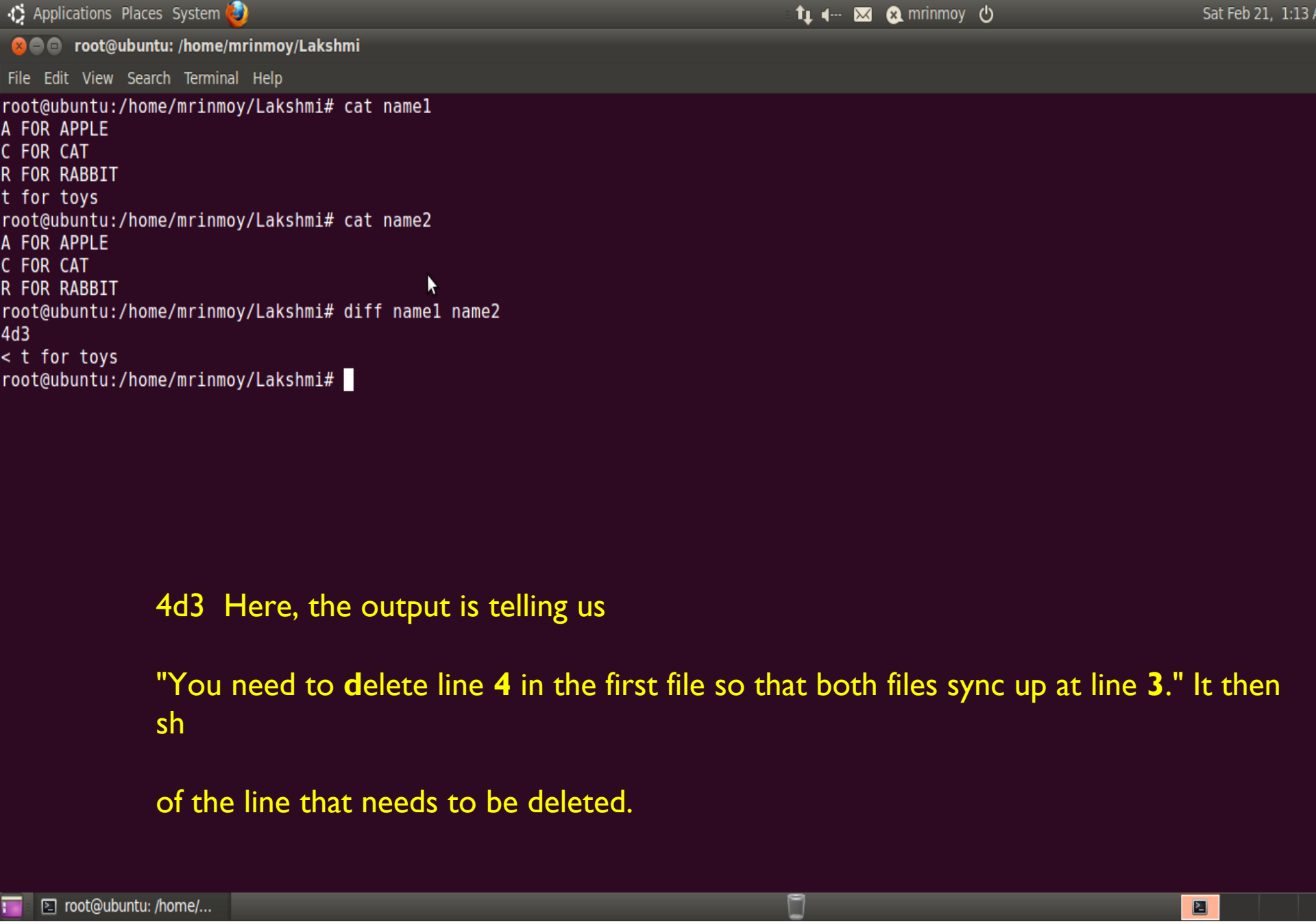

Applications Places System
root@ubuntu: /home/mrinmoy/Lakshmi
File Edit View Search Terminal Help
root@ubuntu:/home/mrinmoy/Lakshmi# cat name1
A FOR APPLE
C FOR CAT
R FOR RABBIT
root@ubuntu:/home/mrinmoy/Lakshmi# cat name2
A FOR APPLE
C FOR CAT
R FOR RABBIT
root@ubuntu:/home/mrinmoy/Lakshmi# diff name1 name2
root@ubuntu:/home/mrinmoy/Lakshmi#
Both files are similar



In our output , "**1,3c1,3**" means:

"Lines **1** through **3** in the first file need to be **c**hanged in order to match lines **1** through **3** in the second file."

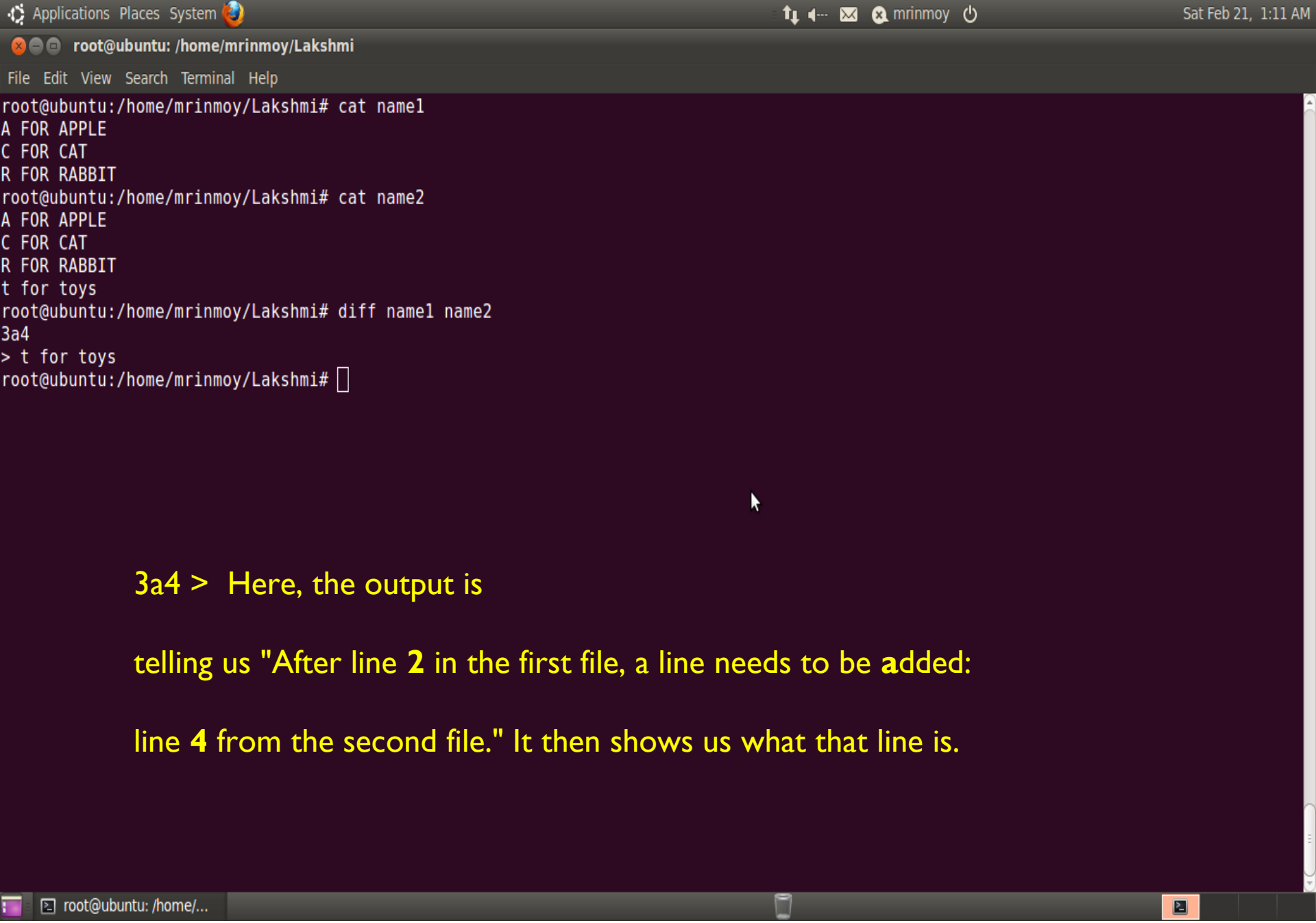
It then tells us what those lines are in each file:



4d3 Here, the output is telling us

"You need to **d**elete line **4** in the first file so that both files sync up at line **3**." It then sh

of the line that needs to be deleted.



3a4 > Here, the output is

telling us "After line **2** in the first file, a line needs to be **a**dded:

line **4** from the second file." It then shows us what that line is.

Shell

- A shell script is an executable file which contains shell commands.
- The script acts as a "program" by sequentially executing each command in the file.
- There are 3 most widely used shells:
    - Bourne shell (.sh)
    - C shell (.csh)
    - Korn shell (.ksh)

- **When to use shell script:**

    1. Customizing your work environment
    2. automating your daily task
    3. automating your repetitive task
    4. Performing same operations on many files.

echo: Echo command is used to used to display the message on the screen.

Vi editor:

Vi provides basic text editing capabilities.

Starting vi:

To start vi, enter:

vi filename
Where filename is the name of the file you want to edit. If the file does not exist, vi will create it for you. You can also start vi without giving any
filename. In this case, vi will ask for one when you quit or save your work.

Modes in vi editor:

While working with vi editor you would come across following two modes:

1. Command mode
2. Insert mode
3. ex command mode

1. **Command mode:**

This is the default mode in this mode all the keys pressed by the user are interpreted to be editor commands.

2. Insert mode:

Insertion of new text, editing of existed text and replacing of existed text.

3. ex command mode:

This mode permits us to give commands at the command line. bottom of the vi screen is called the command line. Vi used the command line to display messages and commands.

To enter insert mode, press: i

Note that in the bottom, vi indicates that you are in insert mode.
After entering text, press ESC to return to command mode.

Saving Your Work and quit (write and quit):-- **:wq (press enter)**
:q (quit)
:q! (forceful quit)

Vi's Exit Commands

Command	*Significance*
:w	Will write the contents of editing buffer into the file
:wq	Will write and quit
ZZ	Equivalent to :wq
:x	Also equivalent to :wq
:q	Quit
:q!	Quit without saving

Vi's Move Commands

Command	*Significance*
l or →	Move right one character
h or ←	Move left one character
j or ↓	Move down one line
k or ↑	Move up one line
0	Moves the beginning of the line
$	Moves the end of the current line
+	Moves the beginning of the next line
-	Moves the beginning of previous line

Vi's Modify Commands

Command	*Significance*
dw	Deletes from the cursor to the end of the word
3dw	Deletes three words
d$	Deletes to the end of the line
D	Same as d$
3d$	Deletes to the end of the third line ahead
d)	Deletes to the beginning of the next line
d}	Deletes to the beginning of the next paragraph
dd	Deletes the current line

Vi's Modify Commands Cont…

Command	*Significance*
yw	Yanks a word
3yw	Yanks three words
y$	Yanks to the end of the line
y)	Yanks to the end of the sentence
y}	Yanks to the end of the paragraph
y]]	Yanks to the end of the section
yy	Yanks the current line
3Y	Yanks three line, starting at the current line

Sample shell script:

#which is used to display current working directory , date and user information.

```
pwd
date
who
```

symbol marks the beginning of a comment.


Thank you!