FILE SYSTEM

- File is a container for storing information
- If you name a file foo and write three characters a,b and c into it,then foo will contain only the string abc and nothing else
- File's size is not stored in the file, nor even its name.All file attributes are kept in separate area of the hard disk,not directly accessible to humans, but only to the kernel

Files are divided into three categories

- 1. Ordinary File(regular file): An ordinary file is a file contain only data as a stream of characters.
- 2. Directory File: Directories contain files and other directories,but strictly speaking,**it contains their names and number associated with each name.**For users familiar with Windows , UNIX directories are equivalent to folders.
- 3. Device File: Some device file provide access to hardware such as hard drives, CD-ROM drives, modems, and Ethernet adapters.

Ordinary File

- **Ordinary File**: most common file type
- All program you write belong to this type
- Ordinary File – divided into two types
- -> Text File
- -> Binary File

Text File

- Text File contain only printable characters
- All C and Jave program sources,shell& perl scripts are text files
- It contains line of charcters where every line is terminated with newline character also known as linefeed(LF)

Binary File

- It contains both printable and unprintable characters that cover the entire ASCII range(0 to 255)
- Object code and executables that produce by compiling C programs are also binary files
- Picture, sound and video files are binary files

Directory File

- A directory file contain an entry for every file and subdirectory that it houses
- If you have 20 files in a directory, there will be a 20 entries in the directory.
- Each entry has two components:
- -> filename
- -> unique identification number for the file or directory(called inode number)
- [use ls -i to obtain inode number]

Device File

- Device Filenames are found inside a single directory structure,/dev

Home variable:Home Directory

- When you log on to the system , UNIX automatically places you in a directory called **home directory**
- **$ echo $HOME**
- /home/lenovo   [Absolute pathname, where first slash is a synonym for root, remaining slashes act as delimiters to file and directory names]
-

pwd:Checking your current directory

- pwd(print working directory) command
- $pwd
- /home/lenovo

cd:Changing current directory

- $pwd
- /home/lenovo
- cd progs
- $pwd
- /home/lenovo/progs

mkdir :Making Directories

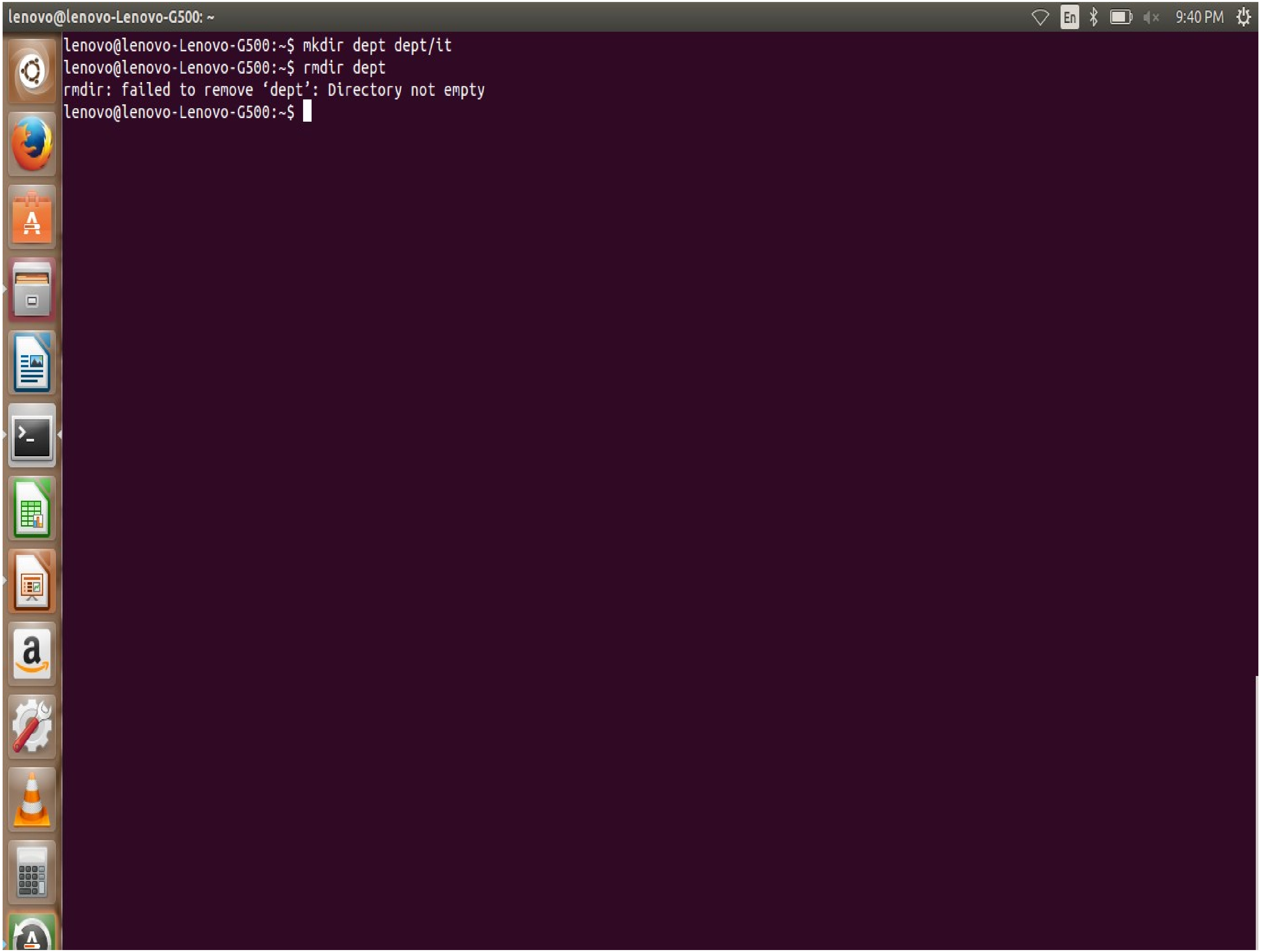
- mkdir patch //directory patch is created under current directory
- mkdir a b c //Output:?

- **mkdir dept dept/it dept/cs** //creates the directory tree
- mkdir dept/it dept/cs dept // Output:?

- System refuses to create a directory
- For ex: mkdir exam
- mkdir: Failed to make directory “exam”, Permission denied
- Reasons:
- 1)directory “exam” may already exist
- 2)There may be an ordinary file by that name in current directory
- 3)The permissions set for the current directory don't permit the creation of files and directories by the user

rmdir: Removing Directories

- rmdir patch //removes a patch directory,it must be empty



```
lenovo@lenovo-Lenovo-G500:~$ mkdir dept dept/it
lenovo@lenovo-Lenovo-G500:~$ rmdir dept
rmdir: failed to remove ‘dept’: Directory not empty
lenovo@lenovo-Lenovo-G500:~$ 
```

- rmdir program/data program/info program
- Output?
- rmdir program program/data program/info

  Output?

What is a path?

- A path is a unique location to a file or a folder in a file system of an OS. A path to a file is a combination of / and alpha-numeric characters.

What is an absolute path?

- An absolute path is defined as the specifying the location of a file or directory from the root directory(/). In other words we can say absolute path is a complete path from start of actual filesystem from / directory.

What is the relative path?

- Relative path is defined as path related to the present working directory(pwd).
- Suppose I am located in /var/log and I want to change directory to **/var/log/kernel**. I can use relative path concept to change directory to **kernel**

- **$pwd**
- **/var/log**
- **cd kernel**

- **Note: If you observe there is no / before kernel which indicates it’s a relative directory to present working directory.**

- Changing directory to /var/log/kernel using absolute path concept.
- cd /var/log/kernel
- **Note: We can use an absolute path from any location where as if you want to use relative path we should be present in a directory where we are going to specify relative to that present working directory.**

- **Problem1:**
- **Present location is /abc/xyz, I am want to remove /abc/xyz/read/hello.txt file.**

- Problem2
- My present location is /etc/samba and now I want to change directory to /etc

Using . And .. in relative Pathnames

- .(single dot) ---> This represents current directory
- ..(two dots)----> This represents the parent directory

- 1) $pwd
- /home/lenovo/program/data/text
- cd ..
- $pwd
- => Output ?
- 2)$pwd
- /home/lenovo/program/data/text
- cd .
- cd ../..
- pwd
- => Output ?

ls: Listing Directory Contents

- ls command is used to obtain a list of all filenames in the current directory[number first, uppercase and then lowercase]

ls Options

- 1) ls -x   //output in multiple columns
- 2) Identifying Directories and Executables(-F)
- ls -Fx
- Two symbols * and / as type indicators
- * indicates that the file contains executable code
- / indiactes a directory

- 3) Showing Hidden Files Also(-a)
- $ls -axF
- ls doesnot normally show all files in a directory.There are hidden files(filenames beginning with a dot)
- where -a option (all) lists all hidden files as well

- 4)Listing Directory Contents:
- $ls -x Music Desktop
- Music:
- Desktop:
- Recursive Listing(-R):The -R(recursive) option lists all files and subdirectories in a directory tree
- $ls -xR

UNIX File System

- /bin and /usr/bin: These are the directories where all the commonly used UNIX commands(binaries, hence the name bin) are found.
- Note that the PATH variable always shows these directories in its list

- /sbin and /usr/sbin:If there's a command that you can't execute but system administrator can, then it would probably be in one of these directories.

- /etc:This directory contains the configuration files of the system
- Your login name and password are stored in files
- /etc/passwd and /etc/shadow

- /dev:This directory contains all device files
- These files don't occupy space on disk
- There could be more subdirectories like pts,dsk and rdsk in this directory

- /lib and /usr/lib:contain all library files in binary form.
- You'll need to link your C programs with files in these directories

- /usr/include: contains the standard header files used by C programs

- Overtime , the contents of these directories would change as more software and utilities are added to the system.
- Users also work with their own files; they write programs,send and receive mail and also create temporary files.These files are available in the second group

- /tmp-The directories where users are allowed to create temporary files
- /var- the variable part of the file system.Contains all your print jobs and outgoing and incoming mail
- /home-On many systems users are housed here.