SET-224 /CET-225
Operating Systems

LAB # 01

LAB Title

Installation of Linux Operating System (Ubuntu) and some basic commands.

Assessment of CLO: 04, PLO: 05

Student Name:	
Roll No.	

Semester		Session	

S. No.	Perf. Level Criteria	Excellent (2.5)	Good (2)	Satisfactory (1.5)	Needs Improvement (0 ~ 1)	Marks Obtained
1	Project Execution & Implementation	Fully functional, optimized, and well-structured.	Minor errors, mostly functional.	Some errors, requires guidance.	Major errors, non-functional, or not Performed.	
2	Results & Debugging Or Troubleshooting	Accurate results with effective debugging Or Troubleshooting.	Mostly correct, some debugging Or Troubleshooting needed.	Partial results, minimal debugging Or Troubleshooting.	Incorrect results, no debugging Or Troubleshooting, or not attempted.	
3	Problem-Solving & Adaptability (VIVA)	Creative approach, efficiently solves challenges.	Adapts well, minor struggles.	Some adaptability, needs guidance.	Lacks innovation or no innovation, unable to solve problems.	
4	Report Quality & Documentation	Clear, structured, with detailed visuals.	Mostly clear, minor gaps.	Some clarity issues, missing details.	Poorly structured, lacks clarity, or not submitted.	
				Total Marks Obtained Out of 10		

Experiment evaluated by

Instructor's Name	Engr.Bushra Aziz		
Date		Signature	

Copyright © Department of Engineering & Technology – UIT University Karachi

Objective: To get familiarize with virtual machines, installation of Linux and some basic commands in Linux.

Theory:

Virtual Machine

A virtual machine is a software computer that, like a physical computer, runs an operating system and applications. The virtual machine is comprised of a set of specification and configuration files and is backed by the physical resources of a host.

A Virtual Box or VB is a software virtualization package that installs on an operating system as an application. Virtual Box allows additional operating systems to be installed on it, as a Guest OS and run in a virtual environment.

Virtual Machine Programs

There are several different virtual machine programs you can choose from:

Virtual Box (Windows, Linux, Mac OSX)

Virtual Box is very popular because it is open source and completely free. There is no paid version of

Virtual Box, so you don't have to deal with the usual "upgrade to get more features" up sells and nags. Virtual Box works very well, particularly on Windows and Linux where there is less competition.

VMware Player (Windows, Linux)

VMware has its own line of virtual machine programs. You can use VMware Player on Windows or

Linux as a free, basic virtual machine tool. More advanced features many of which are found in Virtual Box for free require upgrading to the paid VMware Work station program. We recommend starting out with Virtual Box, but if it does not work properly you may want to try VMware Player.

VMware Fusion (Mac OSX)

Mac users will need to buy VMware Fusion to use a VMware product, as the free VMware Player is not available on a Mac. However, VMware Fusion is more polished.

Parallels Desktop (Mac OSX)

Macs also have Parallel Desktop available. Both Parallel Desktop and VMware Fusion for Mac are more polished than the virtual machine programs on other platforms. Steps Required To Run Ubuntu Linux On Windows 10

- Download Oracle Virtual box

- Download Ubuntu

- Download Virtual box Guest Additions

- Install Virtual box

- Create a Ubuntu virtual machine

- Install Ubuntu

- Install Virtual box Guest Additions Install Virtual Box

There is absolutely no reason not to choose the defaults unless you want to choose a different installation location in which case click on "Browse" and navigate to where you want to install Virtual box. Click "Next" to continue. The next screen lets you choose the Virtual box installation options. Creating Virtual box Desktop Icons. You now have the option to create shortcuts, either on the desktop and / or the quick launch bar and whether to register file associations such as VDI files to Virtual box. It is up to you whether you want to create

shortcuts. Windows 10 is really easy to navigate with the powerful search button so you might decide not to bother creating either of the shortcuts. Click "Next" to continue.
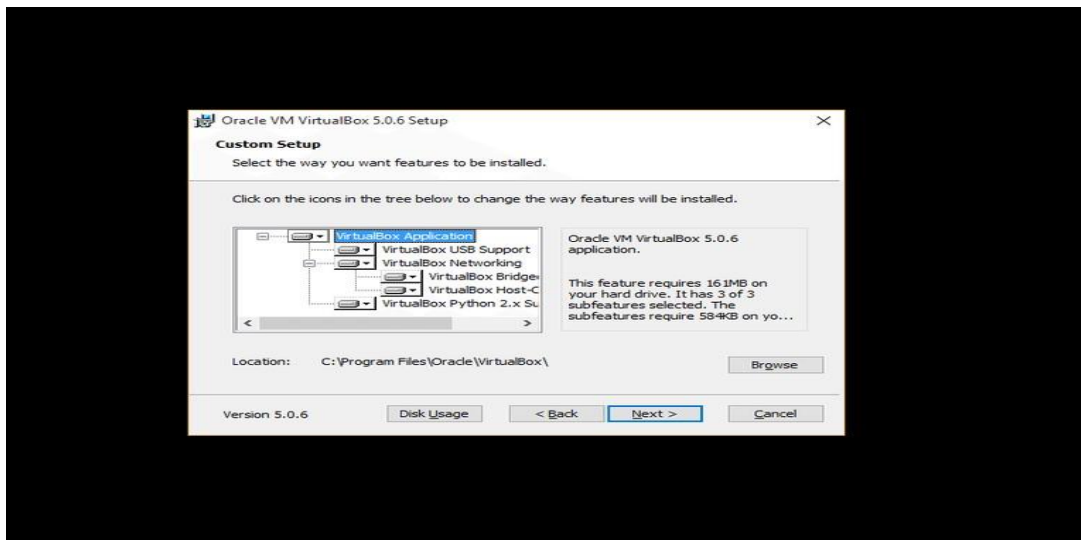

Figure 0-1 Step 1

Figure 0-2 Step 2

Create a Ubuntu Virtual Machine. You can start Virtual box simply by leaving the "Start Oracle VM Virtual box after installation "checked and clicking" Finish " or for future reference click the start button and search for virtual box. Click on the "New" icon on the taskbar. Choose The Type of Virtual Machine.

A list of pre-requisites will be displayed. Basically, you need to make sure your machine has enough power (i.e. plug it in if you are using a laptop), has over 6.6 GB of disk space and is connected to the internet.

You also have the option of downloading updates whilst installing and to install third party software. If you have a good internet connection check the download updates option otherwise remove tick it and leave the updates to install at a later point post installation. I recommend checking the install third party software option as it will allow you to play MP3 audio and watch Flash videos. Click "Continue" and Choose The Installation Type from the menu. Do not worry. This will not erase your physical hard drive. It will just install Ubuntu in the virtual hard drive created earlier on.
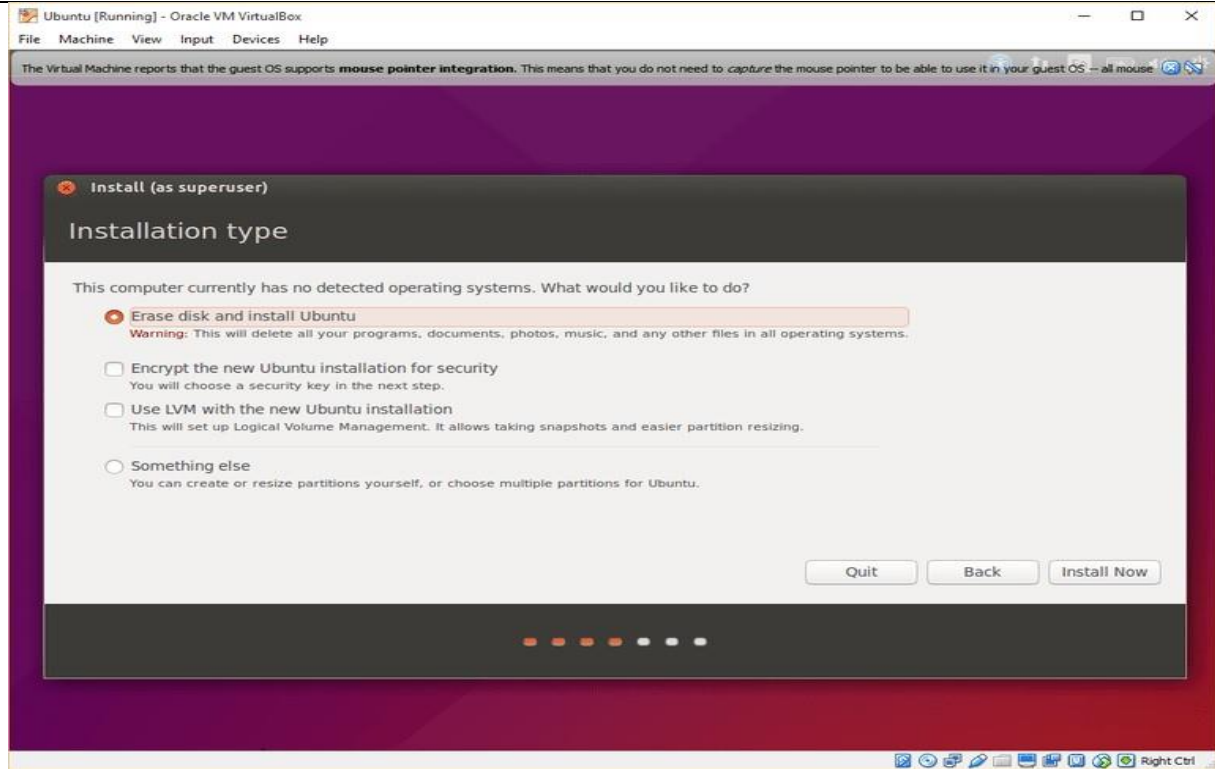
Figure 0-3 Step 3

Click "Install Now".

A message will appear showing you the changes that will be made to your disk. Again this is only your virtual hard drive and so it is safe to click "Continue".
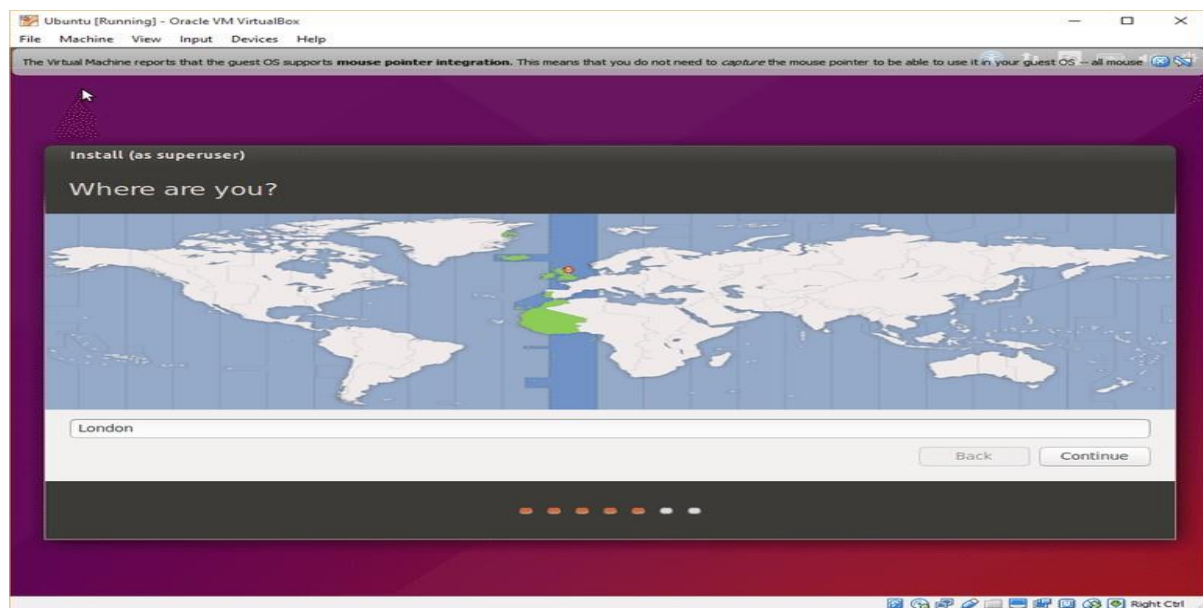

Figure 0-4 Step 4

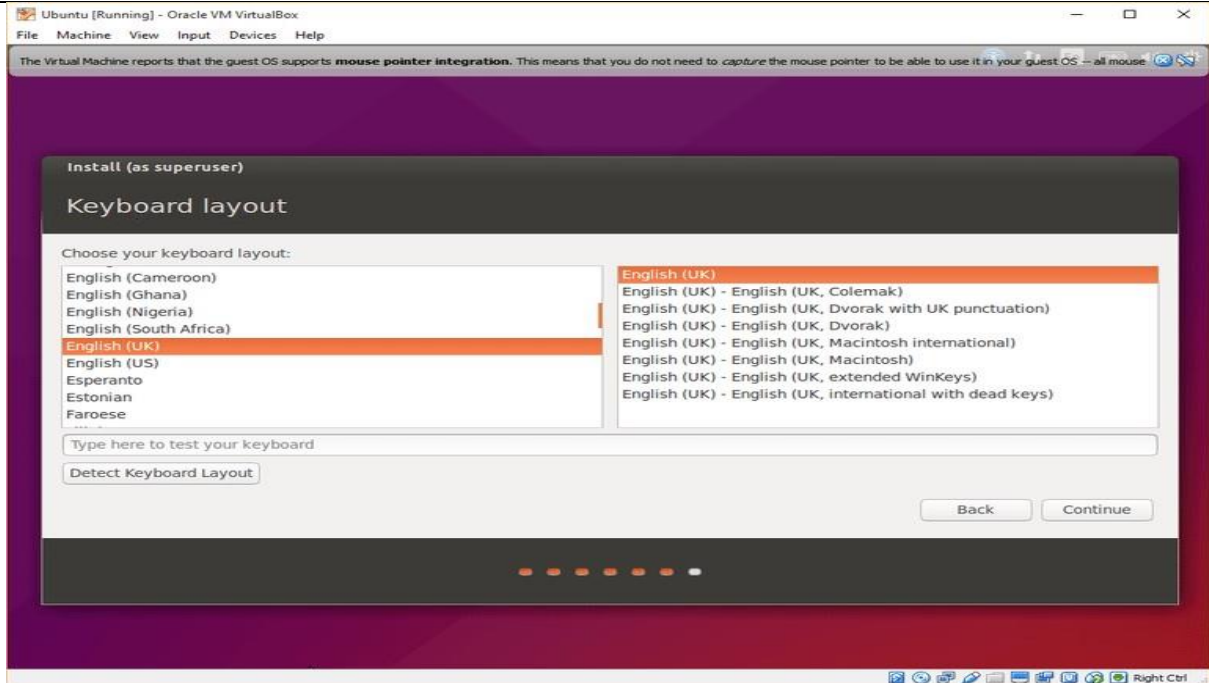**Lab Experiment 1: Installation of Linux Operating System**

Figure 0-5 Step 5

Ubuntu keyboard layout selection and the pen ultimate step is to choose your keyboard layout. You may find that the correct layout has already been chosen but it is not try clicking on the "Detect Keyboard Layout" option. If that doesn't work, click on the language for your keyboard in the left panel and then choose the physical layout in the right pane. Click "Continue".
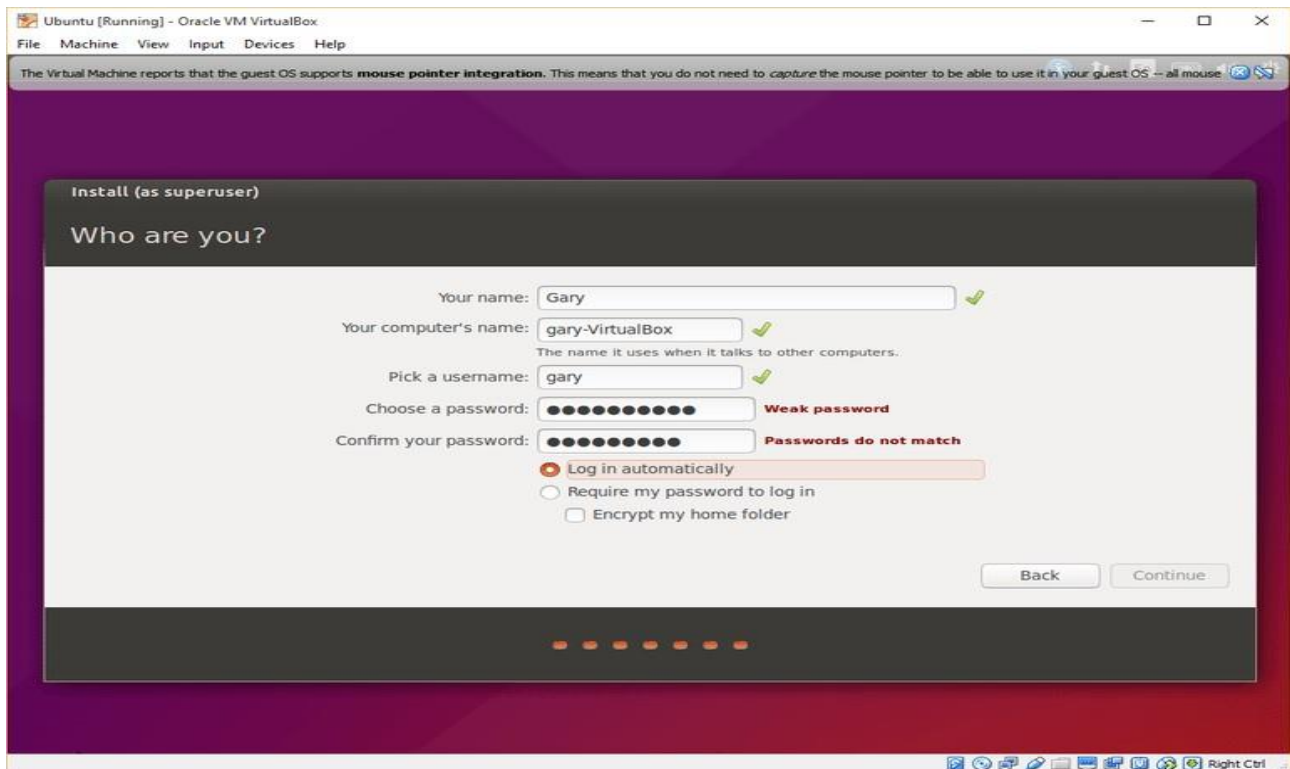

Figure 0-6 Step 6

The final step is to create a user. Enter your name into the box provided and give your virtual machine a name. Now choose a user name and enter a password to associate with that user. (repeat the password as required). The other options are to login automatically or require a password to login. You can also choose to encrypt your home folder. As it is a virtual machine you may as well go for the "Login automatically" option but I usually recommend always selecting the "Require my password to login". Click "Continue". Ubuntu will now be installed.

Lab Experiment 1: Installation of Linux Operating System

When the installation has finished click the File menu and choose close. You have the option to save the machine state, send the shutdown signal or power off the machine. Choose power off the machine and click OK.
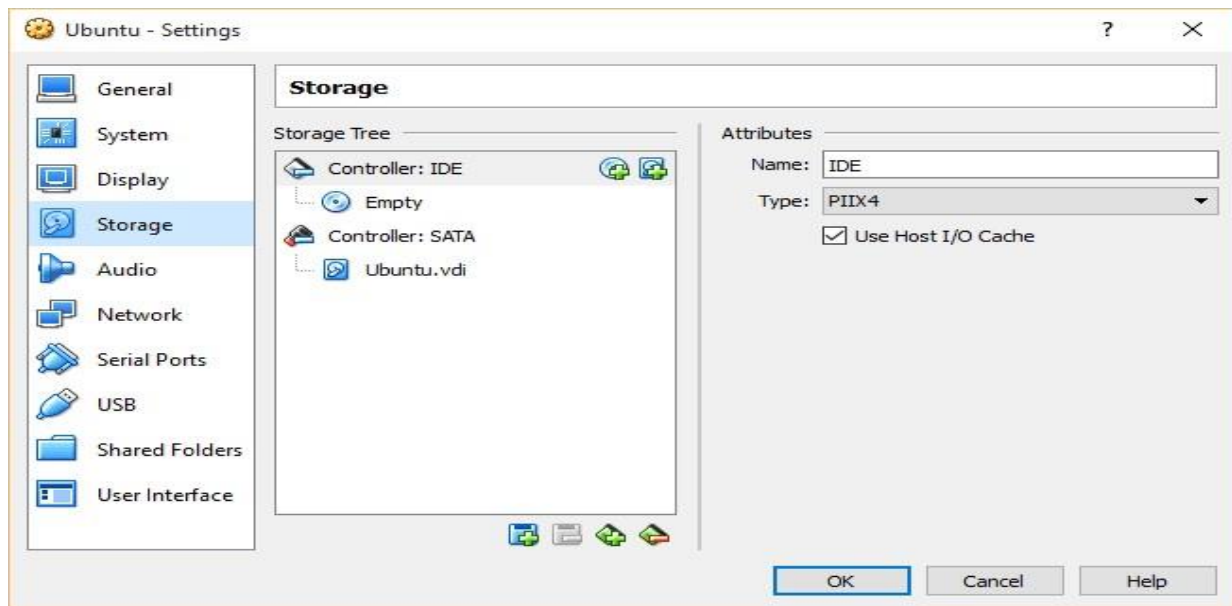

Figure 0-7 VM options

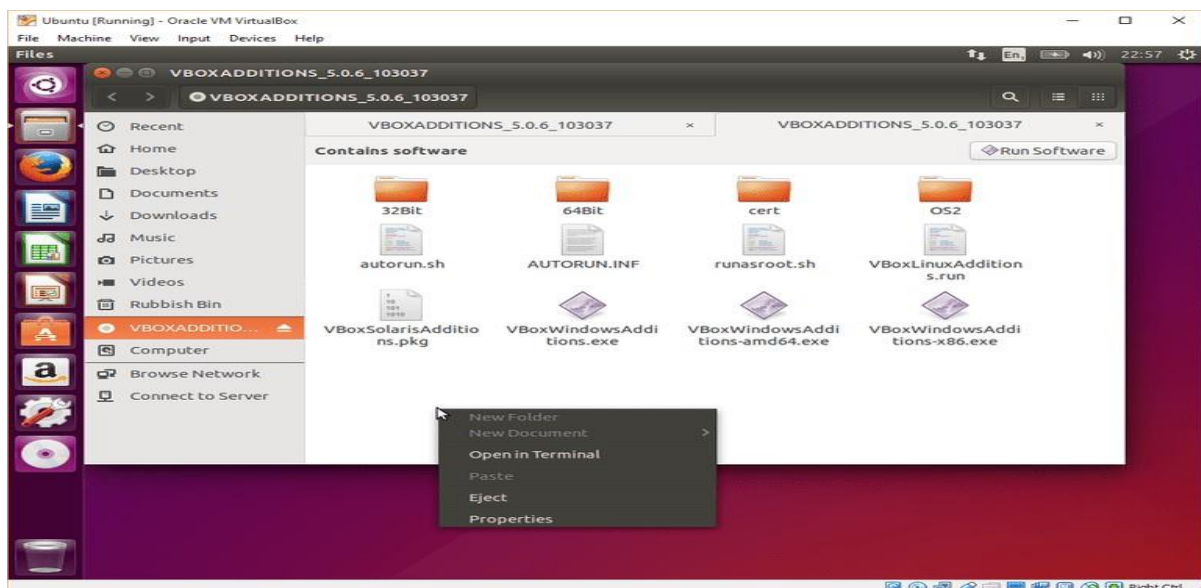Open The Virtual Box Guest Additions CD In Ubuntu and follow the next steps.

Figure 0-8 Drives

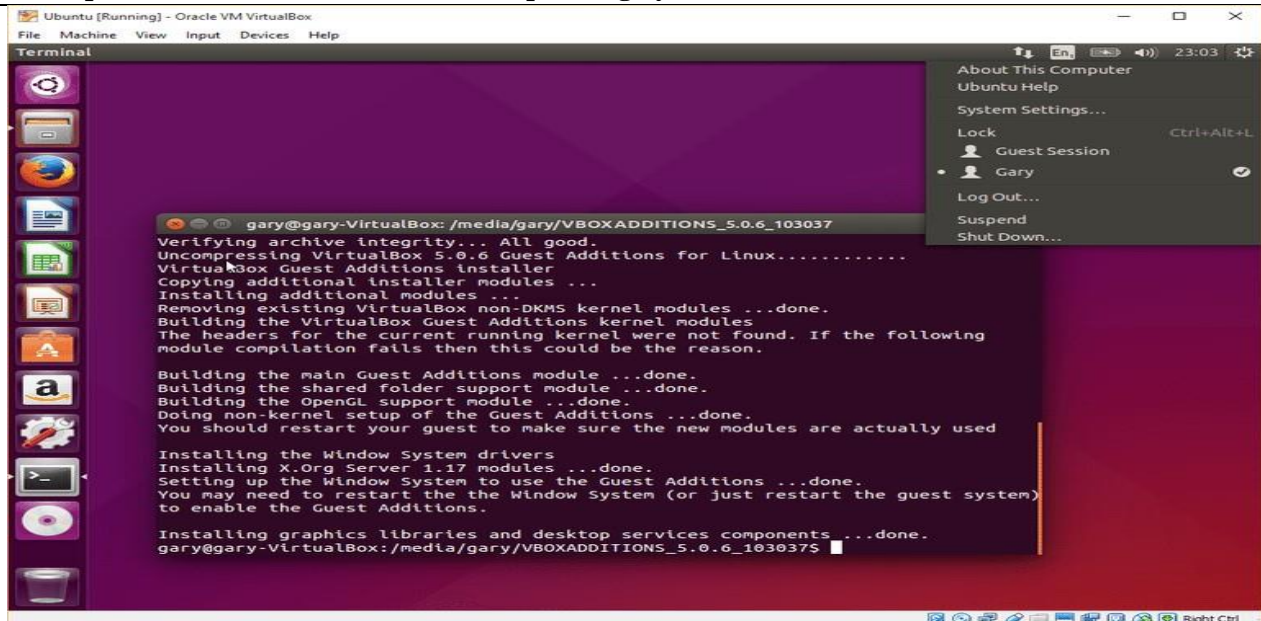**Lab Experiment 1: Installation of Linux Operating System**

Figure 0-9 Terminal

Basic Commands

A Linux command is any executable file. This means that any executable file added to the system becomes a new command on the system. A Linux command is a type of file that is designed to be run, as opposed to files containing data or configuration information.

Input and Output Redirection

Linux allows to "pipe" the output from a command so that it becomes another command's input. This is done by typing two or more commands separated by the | character. The | character means "Use the output from the previous command as the input for the next command." Therefore, typing command_1|command_2 does both commands, one after the other, before giving you the results. Another thing you can do in Linux is to send output to a file instead of the screen. To send output to a file, use the ">" symbol. There are many different reasons why you might want to do this. You might want to save a "snapshot" of a command's output as it was at a certain time, or you might want to save a command's output for further examination. You might also want to save the output from a command that takes a very long time to run, and so on.

Command Options & other parameters

You can use command options to fine-tune the actions of a Linux command. Instead of making you learn a second command, Linux lets you modify the basic, or default, actions of the command by using options. Linux commands often use parameters that are not actual command options. These parameters, such as filenames or directories, are not preceded by a dash.

Linux (Operating System Types)

1.   Command Line Interface (CLI): Command is a name of a program which can run/execute

2.   Graphical User Interface (GUI): It is heavy as CLI

Types of Commands

There are three types of commands in Linux

1. Simple: Execute only by name, e.g, date command in Windows

2. Complex: With which we have some arguments (actions on commands) or options (can change the behavior of command) e.g, dir, format, dir/w, copy

3. Compound: A mixture of two or more commands

Windows	Linux
dir	dir
	ls
	date
	clear

[root@localhost root]#   Note: Every user has assign a directory

[User Name @ Machine Name Directory name]

Executing a Linux Command

From the command prompt simply type the name of the command: $ command

Where $ is the prompt character for the Bourne shell.

Or if the command is not on your path type the complete path and name of the command such as: $ /usr/bin/command.

Some of the frequently used Linux commands are: su, pwd, cd, ls, more and less, find and grep, man. (Refer to Appendix B for detail of few commands)

   1. su

Description: "su" stands for "super user". Runs a new shell under different user and group IDs. If no user is specified, the new shell will run as the root user.

Syntax: su [-flmp] [-c command] [-s shell] [--login] [--fast] [--preserve-environment] [- command=command] [--shell=shell] [-] [user]

(See the man pages for the description of the flags and options by using man su)

$ su <username>   (to become another user) or $ su   (to become the root user).

Adding a new user

#useradd user1

#passwd user1

<password>

```
labit@labit:~$ sudo adduser user2
Adding user `user2' ...
Adding new group `user2' (1002) ...
Adding new user `user2' (1002) with group `user2' ...
Creating home directory `/home/user2' ...
Copying files from `/etc/skel' ...
New password:
BAD PASSWORD: The password fails the dictionary check - it is too simplistic/systematic
Retype new password:
passwd: password updated successfully
Changing the user information for user2
Enter the new value, or press ENTER for the default
        Full Name []: user2
        Room Number []: 2
        Work Phone []: 293848
        Home Phone []: 4939332
        Other []: -
Is the information correct? [Y/n] y
labit@labit:~$ ls
```

```
labit@labit:/home$ ls
labit   user1   user2
labit@labit:/home$ █
```

Deleting users:

```
labit@labit:/home$ sudo deluser user1
Removing user `user1' ...
Warning: group `user1' has no more members.
Done.
labit@labit:/home$
```

2. pwd

Description: Displays the name of the current directory.pwd stands for present working directory. By typing this command you are informed of which directory you are currently in.

Syntax: pwd

```
labit@labit:~$ pwd
/home/labit
labit@labit:~$
```

3. cd

Description: Changes the current directory to any accessible directory on the system.

Syntax: For instance to change from **/(root)** to a subdirectory of **home** then to change subdirectory **labit** use the following:

```
labit@labit:/home$ cd ..
labit@labit:/$ cd home
labit@labit:/home$ ls
labit   user1   user2
labit@labit:/home$ cd labit
labit@labit:~$ ls
```

4. ls

Description: Displays the listing of files and directories. If no file or directory is specified, then the current directory's contents are displayed. By default the contents are sorted alphabetically.

Syntax: To view the contents of user1 home directory use this:

$ ls

To list the contents of any directory on the system use:

$ ls /usr

```
labit@labit:~$ ls
arr.py          Downloads   file1.txt   functions.sh  lab1.sh    loops.sh   prac.py          Rr.py      snap        Videos
binarysem.py    fcfs1.py    file3.sh    gui.sh        lab4.sh    Music      practicefcfs.py  script6.py  Templates   zombie.py
case.sh         fcfs.py     first.py    ifelse.sh     lab4.txt   newfolder  pro.cpp          script.py   thread2.py   zom.py
cpufcfs.py      fc.py       first.sh    iter.sh       lab5.py    orphan.py  pr.py            sem.py      threadM.py
Desktop         fi.c        func.py     l8.py         labp.py    orp.sh     Public           shmem.py    threadp.py
Documents       fil2        func.sh     lab11         loop.sh    Pictures   rr.py            sm.py       threads.py
```

```
labit@labit:/$ ls
bin   cdrom  etc    lib    lib64   lost+found  mnt   proc  run   snap  swapfile  tmp  var
boot  dev    home   lib32  libx32  media       opt   root  sbin  srv   sys       usr
labit@labit:/$
```

5. find

Description: Looks for files below the specified paths that match all the criteria indicated by the command-line options and takes any action indicated by those options. If no paths are specified, the search takes place below the current directory.

Syntax: find [path…..] [Options] (See the man pages for the description of the flags and options by using man find)

This command can be used to search for files by name, date of creation, size and even file type. To search for files by name use the following Syntax:

$ find starting-directory parameters actions

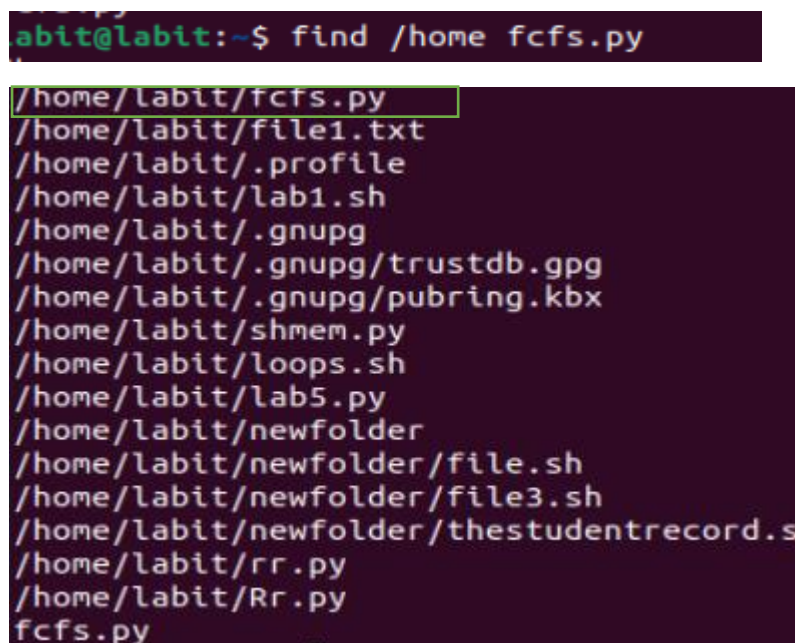The starting-directory specifies where to begin searching.

The parameters are where you specify the criteria by which to search. Here use – name filename to specify the file to search for.

The actions section indicates what actions to take on found files. If –print is used then the full name and path of file is displayed.

To search for all files named myfile on your system, use this:

$ find / -name myfile –print

(Notice that the previous command attempted to search for the entire system. To do this effectively user must be logged in as the root user).

```
abit@labit:~$ find /home fcfs.py
/home/labit/fcfs.py
/home/labit/file1.txt
/home/labit/.profile
/home/labit/lab1.sh
/home/labit/.gnupg
/home/labit/.gnupg/trustdb.gpg
/home/labit/.gnupg/pubring.kbx
/home/labit/shmem.py
/home/labit/loops.sh
/home/labit/lab5.py
/home/labit/newfolder
/home/labit/newfolder/file.sh
/home/labit/newfolder/file3.sh
/home/labit/newfolder/thestudentrecord.s
/home/labit/rr.py
/home/labit/Rr.py
fcfs.py
```

6. **grep**

Description: Searches files for lines matching a specific pattern and displays the lines

Grep stands for Global Regular Expression Parser. What grep does, essentially, is find and display lines that contain a pattern that you specify. There are two basic ways to use grep.

The first use of grep is to filter the output of other commands. The general Syntax is <command> | grep <pattern>. For instance, if we wanted to see every actively running process on the system, we would type ps - a | grep R. In this application, grep passes on only those lines that contain the pattern (in this case, the single letter) R. Note that if someone were running a program called Resting, it would show up even if its status were S for sleeping, because grep would match the R in Resting. An easy way around this problem is to type grep " R ", which explicitly tells grep to search for an R with a space on each side. You must use quotes whenever you search for a pattern that contains one or more blank spaces. The second use of grep is to search for lines that contain a specified pattern in a specified file. The Syntax here is grep <pattern> <filename>. Be careful. It's easy to specify the filename first and the pattern second by mistake! Again, you should be as specific as you can with the pattern to be matched, in order to avoid "false" matches.

Syntax: Assuming that you are in your home directory, then the following command searches for the

word "hello" in each file in your home directory and produces the results as follows:

$ grep hello*

This command then returns one line for each occurrence of the word. The name of the file is also shown. In general, the pattern for the grep command is:

$ grep text-pattern file-list

The text-pattern can be a simple word or phrase or a more complicated regular expression. The use of regular expressions can be found in the man pages.
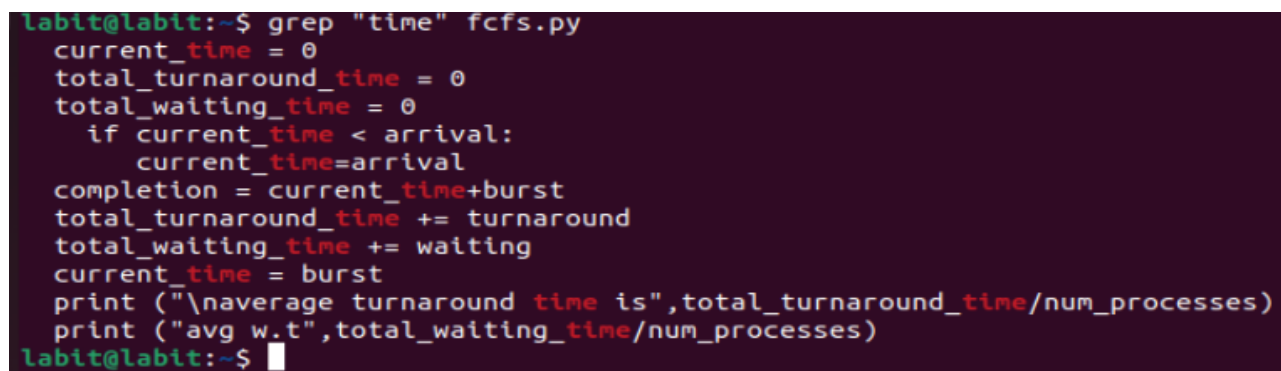
The file-list can take any form allowed by the shell.

To check for the contents of all files in a directory use the following: $ grep text-pattern * where * indicates that all files in the current directory should be searched.

In its simplest form, the text pattern is a single word or part of a word containing no spaces. To search for a phrase such as "is a test", enclose the pattern in quotation marks as follows:

$ grep "is a test" *

(See the man pages for the description of the flags and options by using man grep)

```
labit@labit:~$ grep "time" fcfs.py
  current_time = 0
  total_turnaround_time = 0
  total_waiting_time = 0
    if current_time < arrival:
      current_time=arrival
  completion = current_time+burst
  total_turnaround_time += turnaround
  total_waiting_time += waiting
  current_time = burst
  print ("\naverage turnaround time is",total_turnaround_time/num_processes)
  print ("avg w.t",total_waiting_time/num_processes)
labit@labit:~$ 
```

7. USING THE man PAGES

The man pages are manual pages provided in a standard format with most Linux software. Almost all the commands that ship with Red Hat Linux distribution include man pages. Using the man command in its most basic form, any existing man page can be read:

$ man command-name

The above displays the man page for the specified command and allows scrolling through it and searching it the same way as when using the less command to display text. If the specified man page cannot be found an error is displayed.

Some more useful Commands: root]# clear root]# ls –r (show files in reverse order) root]#
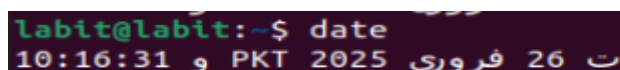   cd /   /]

/]# cd root

root]# ls –l (show root info) root]# dir; ls (compound command) root]# dir;date

root]# cd.. (move to previous directory)

8. Date:

The date command in Linux allows the user to display the current date and time in a variety of formats and set the system date and time**.** Being a part of the **core utilities** in Linux and Unix-like operating systems, making it an essential command for system configuration and scripting**.**

```
labit@labit:~$ date
10:16:31 و PKT 2025 فروری 26 ت
```

9. **wc**

wc command in Linux indicates the number of words, characters, lines, etc using a set of options.

- **wc -w** shows the number of words
- **wc -l** shows the number of lines
- **wc -m** shows the number of characters present in a file

```
labit@labit:~$ wc -l fcfs.py
25 fcfs.py
```

10. **whoami**

The **whoami command** provides basic information that is extremely useful when working on multiple systems. In general, if you are working with a single computer, you will not require it as frequently as a network administrator.

```
labit@labit:/home$ whoami
labit
```

11. **ps command in Linux**

ps command in Linux is used to check the active processes in the terminal.

```
labit@labit:/home$ ps
    PID TTY          TIME CMD
   2817 pts/0    00:00:00 bash
   5270 pts/0    00:00:00 grep
   6428 pts/0    00:00:00 man
   6443 pts/0    00:00:00 pager
  36941 pts/0    00:00:00 ps
```

12. **clear**

The **clear command** is a standard command to clear the terminal screen.

13. **uname**

The **uname command** is used to check the complete OS information of the system. Check out the command and the output below

```
labit@labit:~$ uname
Linux
```

 EXERCISES

Write a command to:

- List all files (and subdirectories) in the home directory.

- Count all files in the current directory.

- Create a Text file "test_lastDigitOfyourRollNumber". Store some data in it then use "grep" to find some text_pattern in it.

- Display the content of "test_lastDigitOfyourRollNumber" file with as many lines at a time as the last digit of your roll number.